Toward Efficient Blockchain for Internet of Vehicles with Multi-Channel Scheme and Hierarchical Resource Scheduling

Liming Gao

A Thesis Submitted for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF ELECTRO-COMMUNICATIONS

2021

Toward Efficient Blockchain for Internet of Vehicles with Multi-channel Scheme and Hierarchical Resource Scheduling

by

Liming Gao

Supervisory Committee:

Prof. Tsutomu Yoshinaga

Prof. Mitsugu Iwamoto

Assoc. Prof. Hideki Yagi

Assoc. Prof. Zhi Liu

Assoc. Prof. Celimuge Wu

Graduate School of Informatics and Engineering

The University of Electro-Communications

マルチチャネルスキームと階層的リソーススケジューリングを備えた車両のインターネットのための効率的なブロックチェーンに向けて

和文概要

高度道路交通システム（ITS）を実現するために Internet of Vehicles（IoV）技術が利用されているが，従来の集中型サービスモデルだけでは，IoV における分散型データ交換需要の増加に対応できなくなりつつある．従来の集中型サービスモデルは，信頼できる第三者機関（Trusted Third Party：TTP）に依存しているため，単一障害点に対して脆弱であり，柔軟性が欠ける．上記の集中型サービスモデルの問題を解決するために，ブロックチェーン技術が注目を浴びている．しかし，IoV における効率的なブロックチェーンシステムの構築には，下記の課題が存在する．

まず，車両の密度は道路，時間帯により大きく変化する場合がある．異なる密度環境において，ブロックチェーン取引（トランザクション）のスループット，遅延要求を保証するには，車両の密度の変化に応じて，ブロックチェーンシステムのパラメータを調整する必要がある．また，ブロックチェーンシステムにおいてコンセンサス（合意）を形成するためにコンピューティング資源の効率的な使用が必要となる．

本論文では，上記の問題を解決するために，下記2つの手法を提案している．1つ目は，車両密度に応じて最適なブロックチェーンパラメータを使用できるマルチチャネルブロックチェーン手法を提案している．提案手法では，事前に複数のブロックチェーンチャネルを定義し，各チャネルは特定の車両密度レベルに最適化される．ブロックチェーンシステムは，車両密度，およびトランザクションのスループットと遅延に関するアプリケーション要求に応じて，最適なチャネルを選択することでシステムパフォーマンスの最適化を目指している．

２つ目は，計算リソースを効率的に配分することで，ブロックチェーンシステムのパフォーマンスを向上させる多階層リソーススケジューリング手法を提案している．ブロックチェーン内部サービスレベル，インフラレベル，ネットワークレベルといった三つの階層におけるコンピューティングリソースの効率的な配分を行うことで，リソース利用効率の最適化を図っている．

ブロックチェーンベンチマークツールである Hyperledger Caliper を使用して，シミュレーションデータを生成し，さまざまなシナリオにおいて提案手法の性能を評価している．既存のベースライン手法と比較することで提案方式の優位性を十分示している．

上記のように，本論文は，IoV における効率的なブロックチェーンのためのマルチチャンネル手法と階層型リソーススケジューリングを提案し，現実的なシミュレーションを用いて既存手法と比較しながら，提案手法の有効性を確認している．

Table of Contents

List of Figures

List of Tables

Abstract

With the development of wireless communication and data processing technology, the internet of vehicles (IoV) is designed by researchers to associate with implementing the intelligent transportation system (ITS). However, due to the exploding demand for data communication and rapidly increasing service requirements, the traditional centralized cloud architecture alone can not satisfy the IoV system completely. The traditional centralized architecture of the vehicular network has the potential risk of a single point of failure and lacks autonomy since the system highly relies on a trusted third party (TTP) to provide identity management. Fortunately, the emergence of blockchain technology brings a new way to settle the problems in the traditional centralized architecture.

Nevertheless, there are still some problems existing in the integration of blockchain technology in IoV systems. First, the density of vehicles may vary greatly according to the road condition and traffic flow period. To ensure the throughput and delay requirements of blockchain transactions in different vehicle density environments, it is necessary to adjust the parameters of the blockchain system according to changes in vehicle density. In addition, since the blockchain system is usually a computing-intensive system, efficient allocation of computing resources is required to improve productivity and lower the construction cost of the blockchain systems.

In this thesis, the following two methods are proposed to address the problem mentioned above. The first one proposes a multi-channel scheme for the blockchain-enabled IoV system that can utilize the optimum configuration according to traffic conditions. In this method, multiple blockchain channels are defined in advance to provide adaptive services for different traffic conditions. The blockchain system aims to optimize the system performance by selecting the optimum channel for the next transaction according to the traffic conditions and the demands of the applications.

The second method proposes a hierarchical resource scheduling method that improves the performance of the blockchain-enabled IoV system by efficiently allocating computational resources from three levels: the blockchain service level, infrastructure level, and network level.

Hyperledger Caliper, a blockchain benchmark tool, is used to generate simulation data and evaluate the performance of the proposed methods under different scenarios. As mentioned above, this thesis proposes a multi-channel scheme and hierarchical resource scheduling for efficient blockchain in IoV. The superiority of the proposed methods is fully demonstrated by comparing them with the existing baseline method using realistic simulation.

Acknowledgments

It is a genuine pleasure to express my deep sense of thankfulness to everyone who has supported and advised me during the preparation of my thesis.

First of all, I would like to express my deepest gratitude to my supervisor, Associate Professor Wu Celimuge, who has given me great encouragement and continuous guidance throughout the process of writing this thesis. He gave me a lot of enlightening ideas and inspired me to a large extent. Without his professional and patient guidance, the completion of this study could not have been possible.

Secondly, I would like to thank Professor Tsutomu Yoshinaga, Professor Mitsugu Iwamoto, Associate Professor Hideki Yagi, and Associate Professor Zhi Liu for the precious guidance and enlightenment in completing this thesis.

Thirdly, I would like to appreciate all the members of Tsutomu Yoshinaga Lab for their selfless help and kind companionship, which gave me the courage to settle the difficulties I encountered during my research.

Finally, there is no doubt that I would like to thank my parents and my wife for supporting me and encouraging me to overcome the difficulties in my life and research during these years of study at UEC.

The deepest gratitude for you all.

Chapter 1

1 Introduction

This chapter presents an overview of the internet of vehicles (IoV) and blockchain technology. The structure of this chapter is shown as follows. Section 1.1 presents the research background and the problems that traditional vehicular networks are facing. Section 1.2 introduces blockchain technology and the reason why it is useful to settle the problems in vehicular networks. Section 1.3 introduces problems to be solved in the combination of blockchain technology and IoV technology. Section 1.4 lists the contributions of the thesis. Section 1.5 presents the overall organization of this thesis.

1.1 Background introduction

At present, because of the explosively developing intelligent devices and technologies, we are experiencing an era of big data and our daily life is moving towards a data-driven society, which provides the technical foundation for the implementation of internet of things (IoT) technology [1][2].

At the same time, along with the increasing number of connected devices in the network, the increasing requirements of data processing and transmission in the network will bring a huge challenge for the traditional centralized architecture of the IoT system which will eventually make the system incompetent [3]. Meanwhile, the enterprises or administrative departments also need to invest more and more money to maintain the ordinary operations of the system. Therefore, to lower the cost and match the capability requirement of the IoT system, the design of IoT systems is trending increasingly decentralized as shown in Figure 1-1 [4].

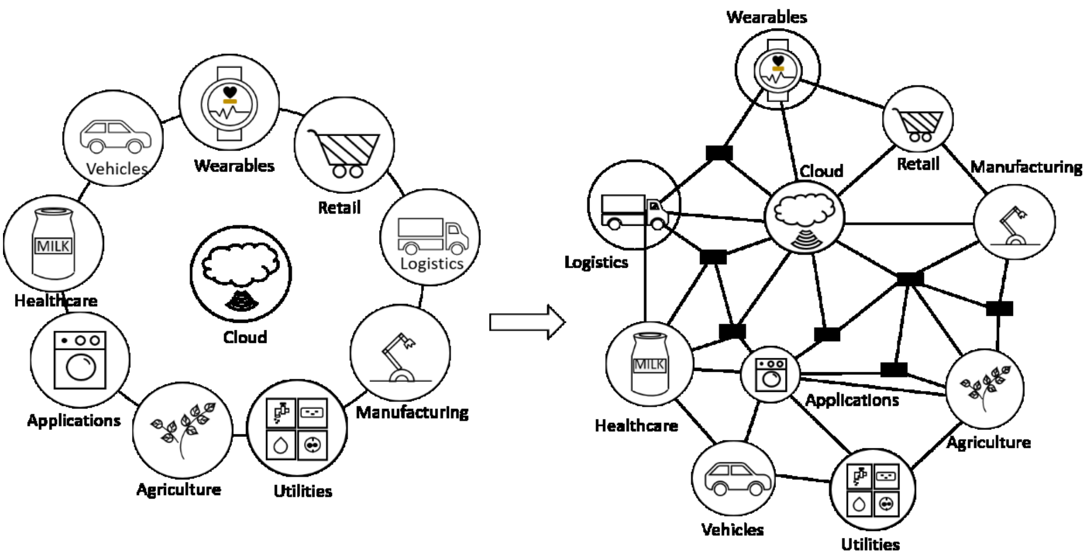


Figure 1-1 The development tendency of IoT systems.

As a typical application of the IoT system, the intelligent transportation system (ITS) is considered to be an effective solution for future transportation systems [5][6]. ITS is a synthesis of a variety of traffic management infrastructures and applications, and combines advanced technologies such as cloud/edge computing, data processing, and wireless communication to build an all-round transportation system to achieve efficient and real-time data collection, communication, and control [7][8].

In the traditional vehicular networks, traffic management is realized through a centralized cloud server which is suffering from the same drawback of the traditional centralized IoT architecture [9][10]. To break through the disadvantages of traditional centralized cloud servers as well as to adapt to the development trend of IoT systems, many decentralized designs have been proposed, such as wireless mesh networks (WMNs) and mobile ad hoc networks (MANETs) [11][12][13].

Vehicular Ad Hoc Network (VANET) is a typical IoT application that is designed for vehicular networks [14]. VANET is one of the key components of realizing the ITS that helps the vehicles to realize vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P), vehicle-to-sensor (V2S), and vehicle-to-cloud (V2C) communication to implement the IoV system as it is shown in Figure 1-2 [15][16]. IoV is derived from IoT and aims to connect all the vehicles, devices, and infrastructures in the vehicular network in a distributed way to break through the limitations of a centralized cloud server architecture

[17][18].

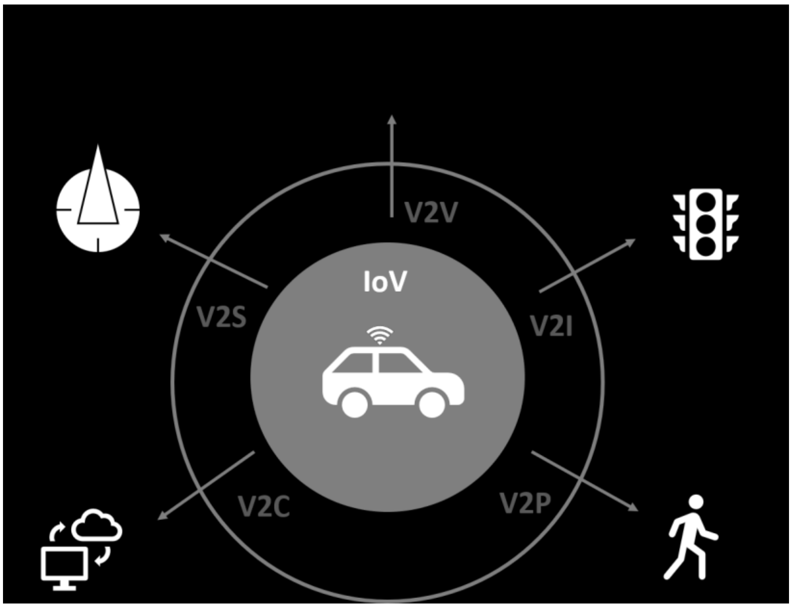


Figure 1-2 Communication patterns in IoV.

However, as a typical distributed system, IoV is at a higher potential risk for many kinds of cyber-attacks because of the inconsistent firmware versions, the difficulty for updating security patches in time, authentication and authorization violations via various network ports of the connected devices, and the casual data transmission without encryption [19][20]. It is easy for a malicious node to attack the system or deny the benefit they got from the system and refuse to fulfill its obligations if there is no centralized control center [21]. Moreover, it is hard to distinguish whether the data are authentic or have been tampered with by malicious nodes which increases the difficulty of the collaboration [22][23].

In addition, due to the heterogeneity of applications and equipment from different service providers and manufacturers, it is difficult to guarantee the compatibility and efficiency of the networks [24]. These devices are not always cooperating efficiently. Vehicles and smart devices have no desire to share data with others because it needs appropriate rewards or incentives to pay for the resource costs of sensing data or delivering messages [25].

Fortunately, the emergence of blockchain provides a new direction to solve the problems faced by traditional vehicular networks. The consensus algorithms, encryption technology, smart contract, distributed data storage, and other technologies adopted by the blockchain system will effectively lower the risk of cyber-attacks and increase the enthusiasm of vehicles to participate in data collection and sharing [26].

1.2 Combination of the blockchain and IoV

These complicated and transdisciplinary problems in the vehicular network mentioned above are hindering the further development of IoV technology [27]. In order to solve these problems, lots of researchers have been trying to use blockchain technology in IoV to reduce the risk of cyber-attacks and motivate the vehicles to participate in the data sharing between heterogeneous devices in the vehicular networks.

Blockchain technology is a new type of distributed ledger technology (DLT). As the fundamental technology of Bitcoin (a well-known cryptocurrency), it was first introduced by a researcher or group called Satoshi Nakamoto in 2009 [28]. Blockchain is a distributed digital ledger that stores the data in the chronologically growing list of blocks. The main contribution of blockchain technology is to enable the nodes in a distributed system to transfer digital value without a trusted third party (TTP). Meanwhile, it can also introduce effective incentive mechanisms to a distributed system to encourage the distributed nodes in the system to participate in the data sharing and as well as making correct decisions for the rewards.

Based on the above characteristics of blockchain technology, many obstacles of traditional IoV networks can be overcome through the combination of blockchain technology and IoV technology [29]. Firstly, blockchain can help the vehicular networks establish a distributed trust system to reduce the risk of being attacked by malicious nodes without the need for TTP. Secondly, blockchain can help provide an appropriate incentive mechanism for promoting data exchange and resource sharing between heterogeneous devices [30]. Thirdly, blockchain can also help solve the scaling problem of the IoV system caused by centralized cloud servers and allow more devices to connect to the network at a lower cost [31] [32].

However, the dynamic features of the network topology and uneven resource distribution of the IoV have brought major challenges to the application of blockchain technology in IoV [33]. To improve the efficiency of blockchain technology in IoV environments, it is necessary to design a particularly customized blockchain architecture and communication mechanism for vehicular networks [34][35].

1.3 Customize blockchain for the internet of vehicles

Since blockchain technology was proposed as the underlying technical framework of Bitcoin, its original design purpose was only to serve digital currency and finance. For high-frequency and dynamic networks like the IoT system, the current blockchain technology is quite inefficient to adapt to the dynamic system. Especially when it is used in combination with the IoV system to solve some traditional distributed problems in vehicular networks, its limitations will be particularly obvious.

The performance of a blockchain system is highly related to the parameters of the system such as block size (the number of the transactions in one block) and batch timeout (the enforcement block formation intervals) [36]. For example, when the block size is big, there will be more transactions in a block and therefore if the transaction arriving rate is low, it will take a longer time to pack a block and the transaction confirmation latency will be higher than a configuration with smaller block size. In the conventional blockchain system, most of the parameters are determined in advance. Modifying the parameters requires a lot of procedures and the agreement of most nodes. This will not only take unnecessary time but will also generate a lot of additional network overhead. So, it is not suitable to directly deploy an existing blockchain architecture in an IoV environment.

Moreover, different IoV applications have different requirements for blockchain services [37]. For example, emergency notification systems have very strict delay requirements, while some vehicle data collection systems need higher throughput. Thereby, it is essential for the blockchain system to be flexible enough to adapt the performance requirements under different conditions [38].

In addition, blockchain systems are usually computing-intensive architecture. The more resources allocated to the system, the better it will perform [39]. However, in a blockchain-enabled IoV system, the number of vehicles is always changing, and the data flow generated by the vehicles will also change with the density of vehicles and road conditions. If we allocate the resources for the system based on the data flow during the peak period, there will be a huge waste of resources in idle time. Correspondingly, if we allocate resources based on the average value, we can reduce waste, but it will greatly affect the performance of the system during peak periods.

Thus, it is necessary to develop an efficient and flexible blockchain architecture according to traffic conditions and specific service requirements. It should be flexible enough to cope with the mobility challenges of the dynamic topology, and efficient enough to make the best use of limited resources in the vehicular networks.

1.4 Contributions

This thesis discusses the advantages and necessity of combining blockchain technology with the IoV and explores the specific difficulties in combining the two technologies. It aims to design a customized blockchain architecture that increases the performance of blockchain-enabled IoV systems and improves resource utilization efficiency of the system from the blockchain configuration perspective and resource management perspective.

For the first proposal which aims to improve the performance of blockchain-enabled IoV systems from the blockchain configuration perspective, the contributions are listed as follows:

- A new three-layered blockchain-enabled IoV scheme with multi-channel management is proposed. All the nodes in the infrastructure layer join multiple channels which are preconfigured in the network to provide adaptive channel services for the vehicles under different vehicle density situations.

- An adaptive channel selection algorithm is proposed to be associated with the proposed blockchain-enabled IoV scheme. The RSUs in the network will collect the safety beacon messages (SBM) from the network and inform the vehicles about the current traffic condition. The vehicles in the network will select the most suitable channel according to the requirement of the application that initiates the message (throughput-sensitive or latency-sensitive) to achieve better network resource utilization and network performance.

- Extensive experiments based on Hyperledger Fabric and Caliper are designed and conducted to evaluate the performance of the proposed scheme. Different from the existing research, we are the first to design and simulate a blockchain-enabled IoV scheme on a benchmark with a standardized workload to evaluate the performance of the proposed blockchain-enabled IoV system.

For the second proposal which aims to improve the performance of blockchain-enabled IoV systems from the resource management perspective, the contributions are listed as follows:

- A hierarchical resource scheduling scheme for blockchain-enabled IoV systems is proposed. The scheme improves the performance of blockchain-enabled IoV systems by efficiently allocating the computing resources of the system. A resource monitoring system is developed to cooperate with the proposed scheme to implement the resource scheduling of the system. It helps collect the operating status of the system and calculates the current latency, transaction success ratio, and throughput of the system.

- A resource control algorithm and a scaling control algorithm are proposed to help improve the resource scheduling of the system. The resource control algorithm increases the CPU share of the peer container to improve resource priority of the peer node of the blockchain system according to the allocated resources and the number of vehicles which can effectively increase the efficiency of the system resource utilization and improve the performance of the system. The scaling control algorithm helps the system to scale up locally or remotely according to the resource utilization of the system. The scaling control algorithm will also update the peer with the most available resources as anchor peer. The proposed scaling control algorithm provides scalability for the system and further improves the system resource utilization. The superiority of the proposal is fully demonstrated by comparing it with the baselines.

1.5 Organization of the thesis

The proposed hierarchical blockchain resource scheduling scheme in Chapter 4 inherits and extends the architecture of the blockchain scheme from Chapter 3. It configures the components of the system with different roles according to their capability and status to make better use of the heterogeneity of the IoV system and make blockchain technology more suitable for the IoV systems. Both the proposals improve the performance of the blockchain-enabled IoV system under different traffic conditions. However, Chapter 3 improves the performance by adjusting the blockchain parameters and providing adaptive services for different traffic conditions and applications requirements. On the other hand, Chapter 4 improves the performance from the network resource scheduling perspective. It helps to adjust the resource scheduling and allocate the resources across the network for different traffic conditions.

The rest of this thesis is organized as follows:

- In Chapter 2, an overview of blockchain technology and related work are introduced including the classification of blockchain technology, applications of blockchain technology, the related research about the blockchain application in different research areas, and some platforms and tools used in this thesis.

- In Chapter 3, the proposed multi-channel scheme for blockchain-enabled IoV is presented. The performance of the deployed blockchain system with different parameters under different vehicular densities is investigated first to find the best configuration under different circumstances. The channel selection algorithm proposed in this chapter will help to select the most suitable channel to send the messages according to the application requirements and the traffic conditions. The simulation design including the simulation assumptions and the setting of the environment are also introduced.

- In Chapter 4, the proposed hierarchical resource scheduling scheme for blockchain-enabled IoV systems is presented. The proposed scheme improves the performance of blockchain-enabled IoV systems by efficiently allocating computing resources with the proposed resource control algorithm and scaling control algorithm. The proposed resource monitoring system is introduced to cooperate with the above algorithms to implement the resource scheduling of the system.

- Chapter 5 concludes the thesis and discusses future work for the research.

CHAPTER 2

2 Blockchain for the internet of vehicles and related works

In section 2.1, the introduction of blockchain technology is presented. Section 2.2 presents the classification of blockchain technology, and Section 2.3 introduces various applications of blockchain technology. Section 2.4 listed the related research about the blockchain application in vehicular networks. Some platforms and tools used in this thesis are introduced in section 2.5. Finally, section 2.6 concludes this chapter.

2.1 Introduction of blockchain technology

2.1.1 Blockchain overview

Like most of the newly invented computer technologies such as edge computing, IoT technology, and big data technology, blockchain technology is not a single information technology. It is a subtle combination of existing technologies (e.g., encryption technology, distributed storage technology, and consensus algorithm) to implement the transfer of a digital asset in a decentralized manner [40].

Broadly speaking, blockchain technology is a newly invented distributed system framework. It verifies and stores data by the use of a chained structure linked with the hash value, maintains the database by the use of distributed consensus mechanism, protects the security of data transfer by the use of cryptography, processes data by the use of self-executing smart contract [41]. In a narrow sense, blockchain is an encrypted digital ledger with a linked data structure that concatenates data packages in chronological order [42].

There are five distinctive features of blockchain which are tamper-resistant, decentralization, anonymity, transparency, and autonomy respectively [43].

- **Tamper-resistant:** Since the consistency of blockchain networks is based on distributed consensus algorithms, nodes can only modify their own copies of blockchain data, and this does not affect the copies held by other nodes. Moreover, because each block contains the hash value of the previous block, and these blocks are connected in chronological order, once one of the pieces of data is modified, all the subsequent pieces of data must be modified to ensure that they pass the verification. This requires simultaneous modification of the copies of data stored by most nodes according to the consensus protocol to make the modified data valid.

- **Decentralization:** Blockchain does not have any centralized infrastructures or management organizations, thus all of the blockchain functions such as data storage and transfer, validation, and consensus are implemented in a decentralized manner among the distributed nodes in the network. Decentralization also helps the system to avoid the single point of failure in the traditional centralized system.

- **Anonymity**: Since the transactions in the system use the transaction address or account address, and the transactions are based on decentralized consensus and encryption algorithm without mutual trust, both parties of the transaction do not need to provide private information to gain the trust of the other party in the transaction process [44].

- **Transparency:** All the transactions or smart contracts in a blockchain system are visible to all the related nodes. All the nodes in the system can verify the validity of a certain transaction or data package with the consensus rules or open-source validation codes embedded in the system. It makes the blockchain system easy to be verified and possesses a particularly high degree of freedom.

- **Autonomy:** All the nodes in the blockchain network can maintain a copy of the blockchain ledger. The consistency of the decentralized nodes in the blockchain network is based on the consensus mechanism predefined in the system. It does not rely on any other nodes or organizations to confirm the validation of the data.

The biggest contribution of blockchain technology is the realization of digital value transfer in a decentralized manner where network nodes do not need mutual trust. It is implemented through peer-to-peer transactions supported by the use of encryption technology, consensus algorithm, game theory, and other techniques and methods [45].

2.2 Classification of blockchain

With the development of blockchain technology, researchers have proposed a variety of blockchain models with different characteristics and functions. These blockchain system models can be divided into three different categories according to their authentication method, ledger data structure, and transaction model as shown in Figure 2-1.

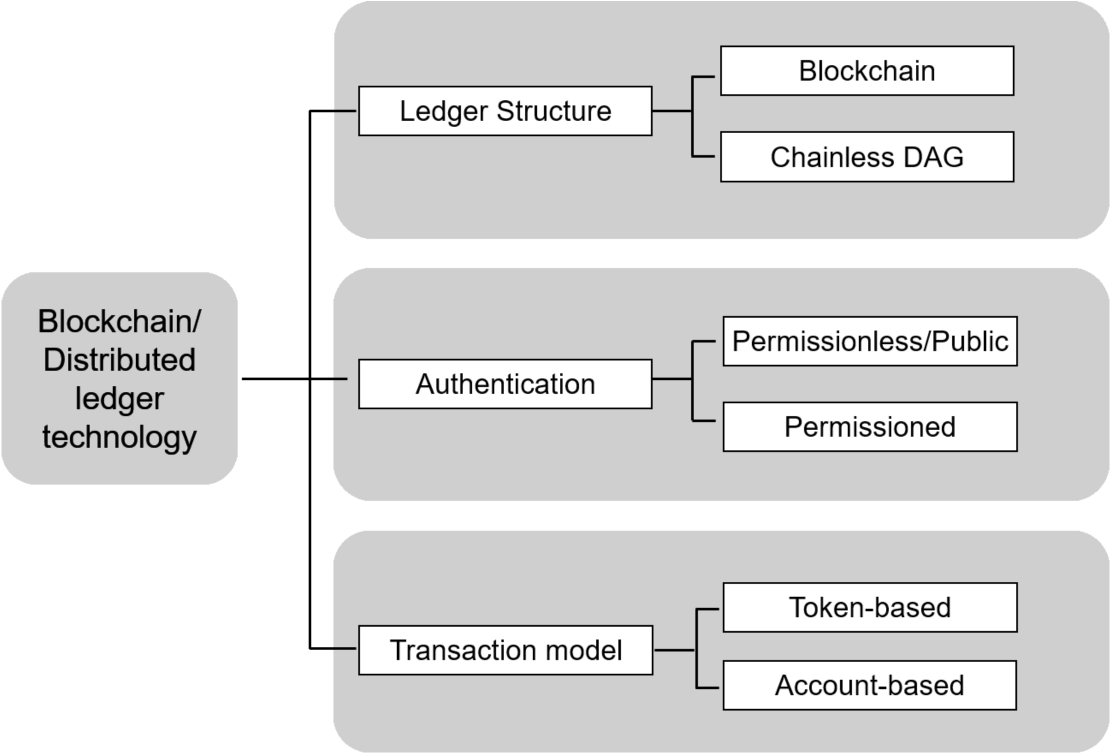


Figure 2-1 Classification of blockchain technology.

2.2.1 Authentication

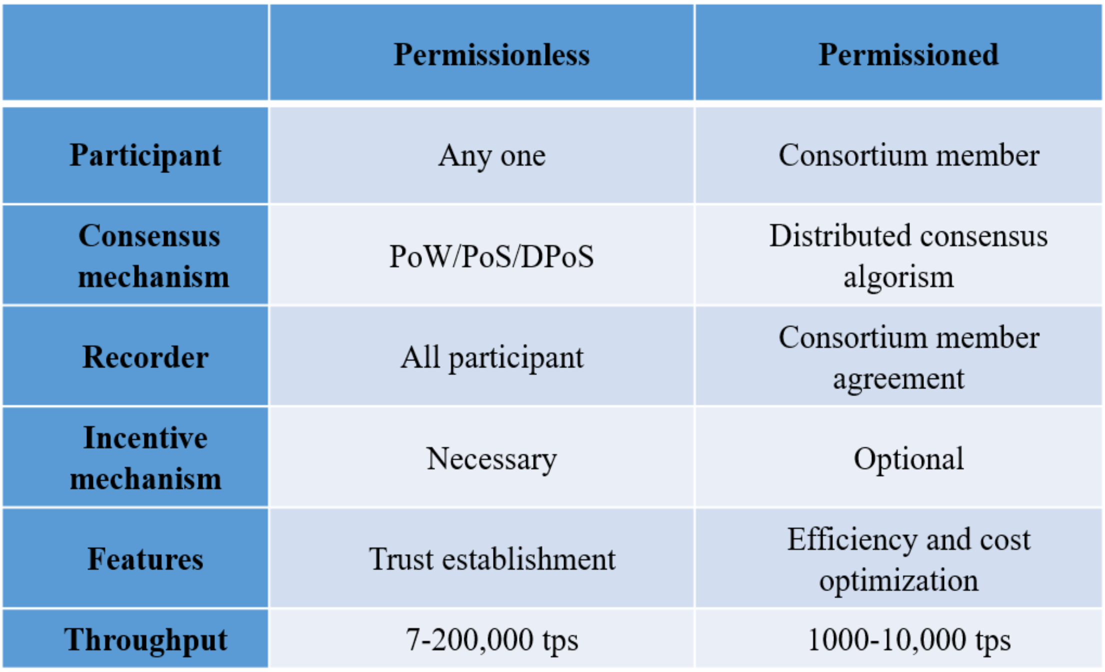
2.2.1.1 Permissionless chain (Public chain)

The comparison diagram of permissionless and permissioned blockchain is shown in Table 2-1. In a permissionless blockchain, there is no access control mechanism. Any individual or group can join the blockchain network and participate in the consensus process. These kinds of blockchain systems, such as Ethereum and Bitcoin [46], are the earliest and most popular frameworks nowadays. Usually, there will be computational competition among the nodes during the consensus process to avoid malicious interference or attack [47].

2.2.1.2 Permissioned chain (Consortium chain)

In permissioned blockchain, all nodes must have the proper certification to participate in the blockchain network. Therefore, it can be regarded as a private blockchain for one or more organizations. The consensus of the block data among these organizations is usually implemented by pre-configured leader selection mechanisms. Since the consensus mechanism in a permissioned blockchain system is defined and managed by the initiation organizations, it is easier to be supervised by authorities and accounting departments [48].

Table 2-1 Comparison of permissionless and permissioned blockchain.



	Permissionless	Permissioned
Participant	Any one	Consortium member
Consensus mechanism	PoW/PoS/DPoS	Distributed consensus algorism
Recorder	All participant	Consortium member agreement
Incentive mechanism	Necessary	Optional
Features	Trust establishment	Efficiency and cost optimization
Throughput	7-200,000 tps	1000-10,000 tps

2.2.2 Ledger data structure

The data structure of the ledger represents the way data is stored in the blockchain network. In the original blockchain system, data is stored in the form of chained data blocks. This is also the reason why this kind of new technology is called blockchain technology. However, with the deepening of blockchain technology research, different application requirements have also emerged, such as scaling and efficiency problems. To address these problems, some different data structures are applied in blockchain systems to settle these problems.

2.2.2.1 Chained structure

The chained structure of blockchain is the most classic type of blockchain data structure, which is employed by most of the popular blockchain platforms such as Bitcoin, Ethereum, and Hyperledger Fabric. The blockchain ledgers are maintained in a decentralized manner. Each node in the network stores a duplication of the blockchain ledger locally. It is easy to be verified but hard to falsify. The transaction data are packed into blocks that contain the hash value of the previous block, and all the blocks are chained in chronological order so that malicious nodes can not tamper with the data without changing all the data blocks afterward [49]. Additionally, there are also some extensional chain structures proposed to further expand the functionality of the blockchain systems, such as sidechain, childchain, and offchain.

2.2.2.2 Chainless structure

Throughput has always been the bottleneck of blockchain technology. In a traditional blockchain structure, the blockchain has only one chain, and blocks can not be packaged and executed concurrently. Thus, the scalability of the blockchain systems limits its wider application. None of the existing blockchain platforms can satisfy the growing throughput requirements of the blockchain system. For instance, the throughput performance of Bitcoin is only about 5 transactions per second. Thereby, to solve the scalability problems of blockchain technology, some different forms of data structure have been proposed by researchers. Directed Acyclic Graph (DAG) structure is one of the most popular data structures which is used to integrate with blockchain systems to provide different blockchain services and improve the scalability of the system. For example, Block Directed Acyclic Graph (BlockDAG) is one of the most popular newly invented data structures. The block in this structure refers to multiple previous blocks instead of a single block to enable parallel processing capability. Another popular structure type is tree DAG which abandons the traditional block data structure and connects the transactions directly in a tree-like DAG form.

2.2.3 Transaction model

Based on transaction models, blockchain technology can also be divided into two categories, token-based model and account-based model. These two transaction modes also represent two different stages in the development of blockchain technology.

2.2.3.1 Token-based model

The most notable token-based blockchain system is Bitcoin. The transactions in this type of blockchain system are usually executed based on the unspent transaction output (UTXO). UTXO is an abstraction of digital concurrency. Each UTXO is analogous to a certain amount of digital concurrency with a chain of ownership signatures [50]. There are no accounts or balances in this type of blockchain system. Ownership of the digital concurrency is recorded on the UTXO.

2.2.3.2 Account-based model

The token is not necessary for the account-based models. The blockchain system with the account-based model is not only used for the digital concurrency systems but also used for implementing various logic functions with smart contracts according to the rules or protocols pre-defined in the contracts. The ownership of the digital concurrency or assets can be transferred directly between accounts [51].

2.3 General blockchain applications and research directions

The main contribution of blockchain technology is the realization of digital value transfer in a decentralized system. With features like distributed data storage, privacy protection, and tamper-resistant, blockchain technology can not only be used to implement cryptocurrency systems [52] but also help implement digital assets or property ownership transfer in a wide range of decentralized systems, especially in data exchanging scenarios under IoT environments as shown in Figure 2-2. Here, we take supply chain and Healthcare as examples.

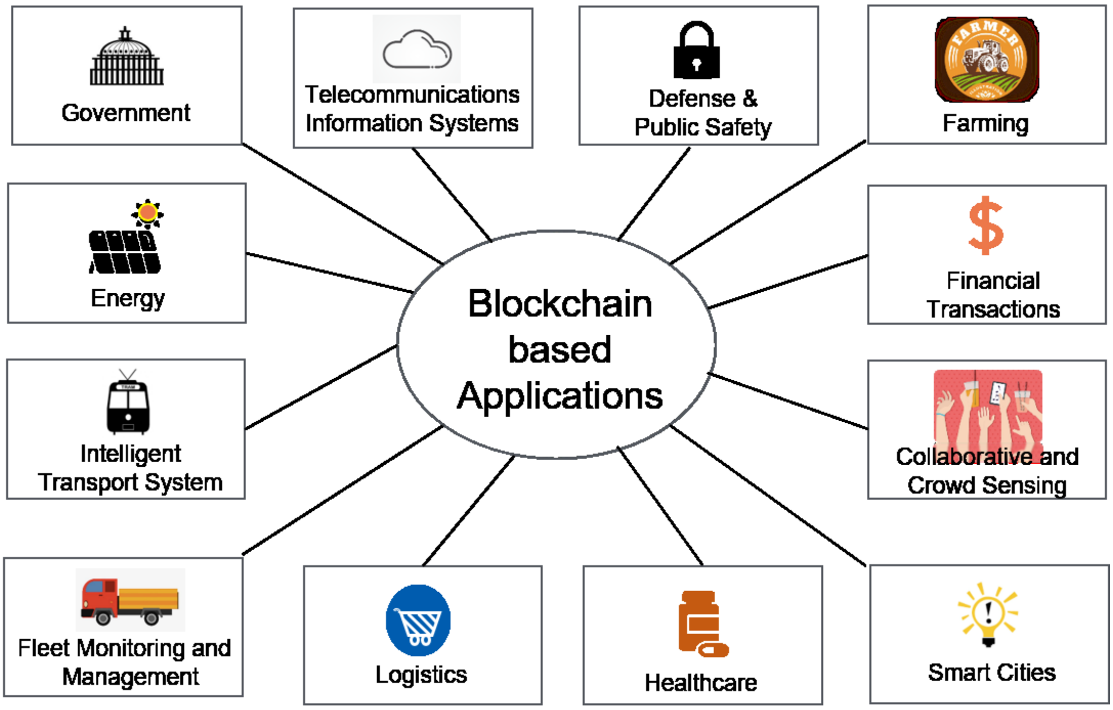


Figure 2-2 Blockchain-based applications.

2.3.1 Applicable directions of blockchain

The blockchain system network is a typical distributed network with various powerful features, therefore, it can be applied to various distributed scenarios [53].

- **Product traceability:** Blockchain technology can provide functions such as tamper-resistant, self-executive smart contract, access control to solve the trust issues in the manufacturing industry, and provide transparency for the information exchange, logistical support, and turnover of capital from the platform level which will make the traceability more convenient and efficient [54].

- **Energy trading:** Blockchain technology can help enable the distributed transactions in energy markets at a lower cost locally at the user end, especially for renewable energy trading [55].

- **Identity management:** The application of blockchain technology in identity management can solve data ownership and trust issues in identity verification without disclosing personal privacy [56][57].

2.3.2 Supply chain management

The application of blockchain in the supply chain is one of the most popular directions of blockchain application, which may bring very significant progress to the supply chain system [58]. Due to asymmetric information and opaque processes, traditional supply chain systems are faced with problems such as low efficiency and scheduling overdue, which makes process tracking and overall planning difficult [59].

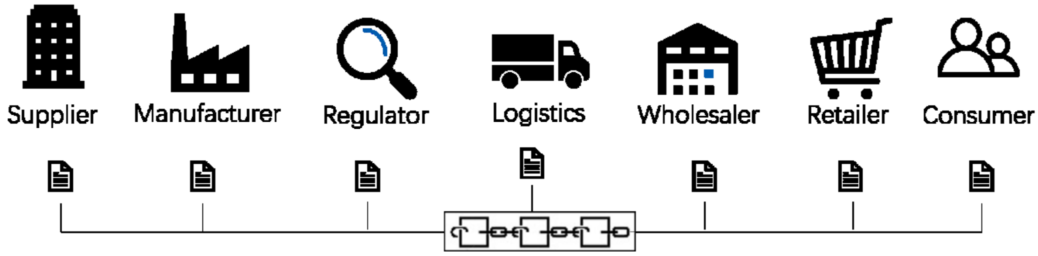


Figure 2-3 Blockchain in supply chain management.

Blockchain technology can provide information transparency as well as privacy protection for the supply chain system. Everyone in the system can obtain corresponding information according to their role without disclosure of the user's privacy [60]. Meanwhile, the data in the blockchain ledger is stored distributedly which makes it hard to tamper with the data. Thus, all the products can be tracked through the system and it is easy to trace the source of fake commodities [61].

2.3.3 Healthcare management

Another important blockchain application is in the healthcare industry. Privacy protection and data sharing among different organizations have always been the pain point of the healthcare industry which could be effectively solved by constructing a blockchain-enabled medical data sharing platform [62]. Figure 2-4 shows the blockchain applications in the healthcare industry.

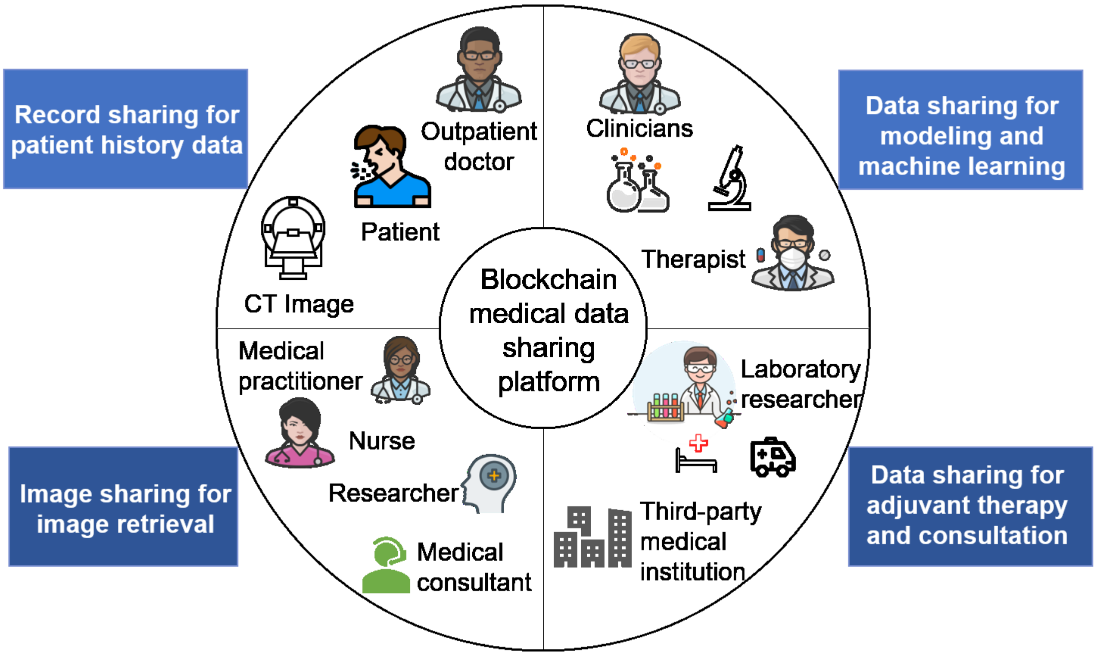


Figure 2-4 Blockchain applications in the healthcare industry.

Blockchain can help improve the healthcare industry from many aspects. From the privacy protection aspect, the medical data including patient information, therapeutic schedule, treatment outcome, and research reports, are stored in the blockchain in an anonymous and encrypted manner which can ensure the sharing of the medical data without leaking the privacy of the users [63]. In addition, since the data are maintained in a distributed way, no one can tamper with the data without the modification of most of the ledger duplicates in the system [64].

From the access control aspect, only those related users or organizations can access the data stored in the healthcare blockchain system. Thus, the medical data are transparent and reliable to those who have access to the system which can greatly improve the efficiency of many processes in the healthcare system, such as medical insurance, research of the medical data, and multilateral consultation [65].

2.4 Research of blockchain technology in IoV

Over the last few years, thanks to the great efforts of blockchain practitioners, it has made significant progress in many academic and industrial fields such as finance, smart grids, supply chains, digital identity, and IoT systems [66]. As one of the typical examples of IoT applications, IoV is one of the current hotspots to integrate blockchain technology. Research in related fields mainly focuses on the application of blockchain technology to help vehicle networks in trust management, incentive mechanisms, vehicle positioning, and system performance improvement [67].

2.4.1 Trust management

In terms of trust management, it has become one of the main directions in applying blockchain technology in many different research areas [68]. It is usually used to help eliminate the dependence of a distributed system on a trusted third party (TTP), and thus enables information sharing among different entities in a trustless environment. Lei et al. [69] proposed a blockchain-based key management scheme to reduce the security key exchange time when a handover happens in vehicular networks. Xia et al. [70] proposed a Bayesian-game-based electricity trading scheme in blockchain-enabled internet of vehicles (BIoV). Blockchain is used to enable trustworthiness trading in a V2V manner. Luo et al. [71] proposed a bidirectional auction mechanism based on the Bayesian game and designed a new price adjustment strategy to improve social welfare and cost performance. The authors employed the blockchain to record energy transactions and protect the privacy of the clients. Gao et al. [72] pointed out the advantages of the VANET system combining blockchain with software-defined networking (SDN) and developed an SDN-based blockchain trust management model for the VANET system to prevent malicious activities.

2.4.2 Incentive mechanisms

Another main direction of combining blockchain and IoV is designing an incentive mechanism for the collaboration of heterogeneous devices and applications in the IoV system [73]. Yin et al. [74] developed a new blockchain-based incentive model to coordinate multi-vehicle collaboration situations for both general and emergent tasks. A bidding mechanism and a novel time-window-based method are developed to further encourage the vehicles to participate. Zhou et al. [75], the authors proposed a permissioned blockchain framework for energy trading on the internet of electric vehicles (IoEV). A contract theory-based incentive mechanism with various contract items is proposed to promote more electric vehicles (EVs) to participate in the scheduling and trading process. In [76], the authors developed a quality-driven incentive mechanism for information sharing in vehicular networks. It also considered both the on-chain and off-chain scenarios to maximize social welfare and minimize the cost. Several other studies also proposed specific blockchain-based incentive mechanisms for vehicular resource management and data storage [77][78][79][80].

2.4.3 Vehicle positioning

Vehicle positioning is also one of the main directions of applying blockchain technology in IoV systems. Song et al. [81] developed a blockchain-enabled cooperative vehicular positioning framework. A deep neural network (DNN) algorithm based self-positioning correction model and a data-sharing scheme for vehicle positioning error are designed to improve the positioning accuracy and reduce positioning error. This system is built on a blockchain-enabled architecture to implement information selection and sharing among vehicles and to ensure security. In [82], the authors proposed a trusted cloaking area construction in positioning services. Blockchain technology and identity pseudonyms are used to improve privacy protection. The construction also employed edge computing to lower the computing latency caused when evaluating the trust value. Li et al. [83] developed a blockchain-based information-sharing scheme for positioning error evolution in IoV. The authors consider malicious nodes and deployed blockchain technology for data storage and sharing as well as protecting the user privacy and ensuring credibility.

2.4.4 System performance improvement

The performance of blockchain in different environments and resource configurations has also attracted the attention of many researchers. [84] studies the impact of mobility on the performance of the blockchain-enabled vehicular ad hoc network (VANET) from three key aspects: block confirmation probability in a rendezvous, rendezvous stability, and exchanged block amount in a rendezvous period. The numerical results indicated that the blockchain system performance was greatly affected by the mobility of the vehicles. Nguyen et al. [85] presented a comprehensive evaluation of the latency impact on Hyperledger Fabric, a well-known consortium blockchain platform. The results demonstrated that Hyperledger Fabric does not provide efficient consistency to deploy in a network environment with large delays. The authors in [86] conducted a comprehensive performance evaluation of the Hyperledger Fabric platform. A pipelined execution method is proposed to improve the efficiency of validation/commit phases and a new type of peer called sparse peer is designed to reduce redundant work. Sharma et al. [87] proposed an optimized structure for the Hyperledger Fabric called Fabric++. It brings a significant improvement in throughput and latency performance against the original Fabric structure.

2.5 Preliminary studies

In this section, the blockchain platform and simulation-related tools employed in this thesis are introduced as preliminary studies which will help better understand the mechanism of the proposals in the thesis.

2.5.1 Hyperledger Fabric

Hyperledger is an open-source project aimed at promoting the interdisciplinary application of blockchain technology. The project was initiated by the Linux Foundation in December 2015. It is designed for enterprise applications including finance, healthcare, IoT, supply chain, and manufacturing industry.

Hyperledger Fabric is a subproject of the Hyperledger that is originally contributed by IBM to the Linux Foundation. It is an open-source permissioned blockchain designed for enterprise-level applications. Hyperledger Fabric has a highly modular architecture that allows most of its components like consensus mechanism and membership services to be pluggable. This characteristic of Fabric makes it more flexible to be customized in the target system.

Hyperledger Fabric is a mature project that has developed a complex structure. Due to the space limitations, this section will only introduce a small part of the Fabric project based on the importance of the contents and the relevance to this thesis.

2.5.1.1 Glossary

Anchor Peer: Peer that is used to hold distribution information of the peers in the system and to guarantee the peers from different organizations can contact each other. In other words, an anchor peer is like a lighthouse in the network. To ensure that the Gossip protocol works properly, there must be at least one anchor peer in the network.

Chaincode: Also known as the smart contract in other blockchain platforms. A chaincode is a piece of code that is designed to help users in a blockchain network implement logical functions and modify database values in the world state through transactions. Chaincode is deployed on peers and can be used by multiple channels.

Organization: An organization is a member of the network invited by the blockchain network provider. Every organization has a membership service provider (MSP) and it needs to add the MSP to the network when it wants to join one. The MSP defines the validity of the signatures issued by the organization.

Peer: A peer is a basic component of a blockchain network. It maintains the ledger of the channel it has joined. They are operated by organizations in the network. A peer in the network can be an endorsement peer (which verifies the read/write set in the transactions proposed by the users) and a commitment peer (which verifies the validity of the transactions in the block received from the ordering service) at the same time.

Ordering Service: A defined collective of nodes that pack the transactions into a block in chronological order with predefined parameters and disseminate the blocks to commitment peers for validation.

Channel: A channel is a blockchain overlay in the network. It is shared only by the peers that have joined the channel to implement data isolation and ensure confidentiality. Every channel maintains its own ledger. There is no public ledger in consortium blockchains. All the transactions are recorded on the respective ledger of the channel that the chaincode is installed on.

Ledger: A ledger is a chain of blocks recording all the transactions in the blocks received from the ordering services. All the peers in a channel maintain their own copies of the ledger.

World State: The world state is a database that contains the current state of the key-value pairs. These key-value pairs can be created, modified, or deleted by the transactions in the ledgers which are verified and committed by the peers.

2.5.1.2 Network example

Figure 2-5 shows an example network from the official website of Hyperledger Fabric.

R1, R2, R3, and R4 are 4 organizations that have decided to set up a Fabric network together. The network configuration (NC4) is initiated by R4 which also provides ordering services for the network. R4 has no plans to conduct commercial transactions in the system, so it does not join any channel. R1 and R2 have a shared channel C1 with predefined channel configuration CC1. R2 and R3 have a shared channel C2 with predefined channel configuration CC2. Each of the organizations R1, R2, and R3 has an application which are A1, A2, and A3. A1 and A2 can generate transactions on channel C1. A2 and A3 can generate transactions on channel C2.

Peer P1 belongs to the organization R1and does not join any channel, but it is maintaining a copy of the ledger L1 which records all the transactions performed on channel C1. P2 belongs to organization R2 and has joined both C1 and C2. P2 is maintaining both the ledger of L1 and L2. P3 belongs to organization R3 and is maintaining L2. The network is managed by R1 and R4 with the predefined network configuration (NC4). C1 is managed by R1 and R2 according to the channel policy predefined in CC1 and C2 is controlled by R2 and R3 together according to the channel policy predefined in CC2.

Ordering peer O4 provides ordering services for both C1 and C2. Each organization in the network has a Certificate Authority (CA) to provide membership certifications for its application users.

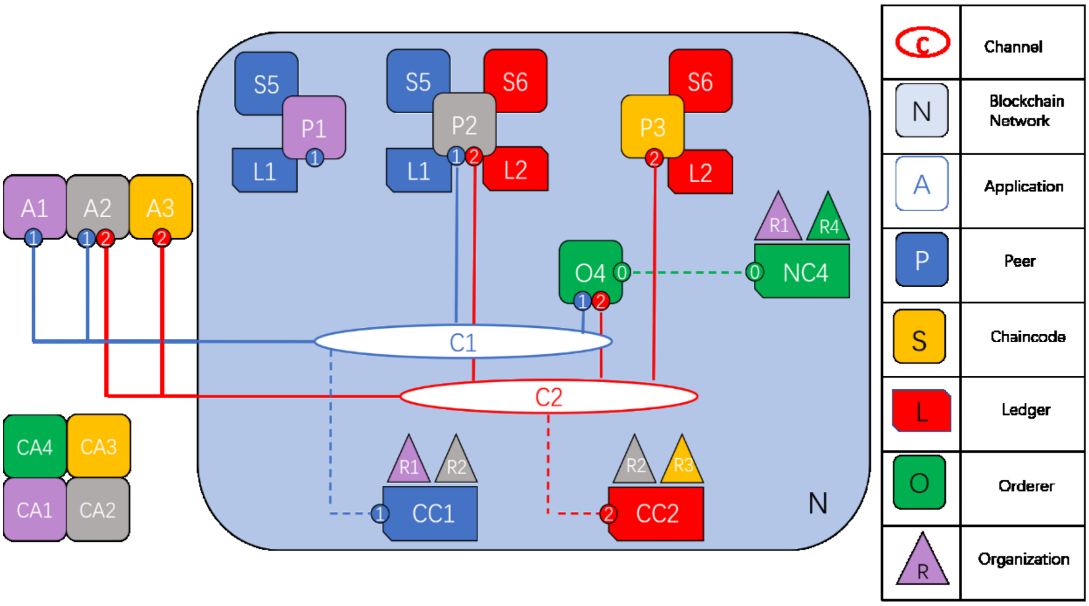


Figure 2-5 The example network of Fabric.

2.5.2 Hyperledger Caliper

Hyperledger Caliper is a benchmarking tool for measuring the performance of a blockchain system. It is contributed to the Hyperledger project by HUAWEI. Hyperledger Caliper is compatible with various blockchain platforms such as all the Hyperledger projects, FISCO BCOS, and Ethereum.

Figure 2-6 shows the working procedure of the Hyperledger Caliper. Before the test, two configuration files and a workload module are needed to define the actions of the Caliper CLI. One of the configuration files is the benchmark configuration file. It defines how the engine works and the other one is the blockchain network configuration file which defines the network topology under test. It can also be connected to an existing network through a connection file. The workload module invokes the smart contracts installed on the target channel in the blockchain network.

[redacted]

Figure 2-6 Working procedure of Hyperledger Caliper.

2.5.3 Stress-ng

Stress-ng is a well-known CPU stress test tool. It is designed to conduct extremely heavy loads for CPU-related hardware of a computer to test the various performance of a system such as thermal overruns and operating system bugs.

It can also be used to produce the requested type of pressures on a specified CPU to test a certain performance of the system under test. It can produce various complex pressures with different stress mechanisms. For example, it can produce a Fast Fourier Transform (FFT) task with 2 workers consuming 50% of the resources of two CPU cores for 10 mins and output the result into a YAML file with the command “stress-ng --yaml -c 2 --cpu-method fft --cpu 2 --cpu-load 50”.

2.6 Conclusion

This chapter first introduces an overview of blockchain technology and related work. Then the classification of blockchain technology is presented. Various applications of blockchain technology and related research about the blockchain application in vehicular networks are also introduced. Some platforms and tools used in this thesis are introduced in the last section of this chapter.

In summary, although blockchain technology has become one of the most popular research directions, it is still in the early stages of its development. In order to make the blockchain better serve the IoV systems, the customized design of the blockchain system is essential to make the blockchain better adapt to the vehicular networks and improve system performance and resource utilization efficiency.

CHAPTER 3

3 Multi-channel blockchain scheme for internet of vehicles

In this chapter, the proposed multi-channel blockchain scheme for IoV is presented. The performance of the deployed blockchain system with different parameters under different vehicular densities is investigated first to find the best configuration under different circumstances. The channel selection algorithm proposed in this chapter will help to select the most suitable channel to send the messages according to the requirements of the applications and the traffic conditions.

In section 3.1, the problems of the traditional ITS system and IoV networks from several different aspects and improvements that need to be made for blockchain technology to run better on the IoV system are discussed.

In section 3.2, the architecture of the proposed multi-channel blockchain scheme for IoV is presented. Section 3.3 discusses the set-up procedures including the definition of the multi-channels and the configuration of the blockchain parameters during the initialization phase of the blockchain for the IoV system. The proposed channel selection algorithm is introduced in section 3.4 to assist the vehicles in selecting the appropriate channel according to current requirements for the next transaction.

In section 3.5, the generality of the proposed scheme is discussed and the simulation design including the simulation assumptions and the environmental settings is introduced in section 3.6. The configuration of the blockchain workload tool is also presented in this section. The results of the simulations are presented in section 3.7 to evaluate the performance of the proposed multi-channel blockchain scheme and section 3.8 concludes this chapter.

3.1 Problem discussion

For the last decades, the concept of ITS has been generally applied to various parts of our regular routines [88]. In contrast to the conventional transportation architecture, these ITS applications implement the all-around management for traffic services through the use of cloud computing and advanced communication technology. In the vehicular network, there are an enormous number of components and applications connected to each other to collect data and share information [89]. Due to the characteristics of the transportation system, most of these applications and services have high requirements for system throughput and delay. This will be an enormous burden for cloud servers to fulfill all these requirements [90][91]. Thus, most of the traditional cloud-based services are trending to deploy distributed architecture to make the best use of all the constituent parts of the system and process the data close to the data source. However, there are still some inescapable needs to be considered in applying the distributed architecture in vehicular networks [92].

In terms of compatibility perspective, the application of VANET in the vehicular network can help implement the distributed communication between vehicles in a small-scale network. However, with regards to a bigger size of scope of implementation, the interoperability of the vehicles and applications in the network is yet an issue that should be tackled. It is difficult to guarantee the exchange of information among vehicles and services from various manufacturers and applications in the ITS system [93].

In terms of security perspective, a distributed architecture is more defenseless against malicious nodes, anybody in the network can be spiteful. This will bring a privacy protection issue [94]. All the nodes in VANETs will periodically broadcast safety beacon messages (SBMs) which contain a lot of significant data like vehicle ID, velocity, and locating information. Also, by gathering and studying the SBMs, malevolent peers can acquire the private data of the target devices [95]. The conventional IoV framework confirms the identification of a vehicle by introducing a trusted third party (TTP). However, in some cases, these TTPs are not reliable. Moreover, TTPs are generally centralized architecture that is vulnerable to a single point of failure [96].

Additionally, in terms of motivation perspective, vehicles and devices from different manufacturers are unwilling to share the data they collected with others without appropriate incentives or reward mechanisms [97][98]. The computing resource or resource cost during the data collection or message delivery needs to be compensated. Moreover, an efficient incentive mechanism will also encourage the vehicles in the network to keep being honest [99][100][101].

These issues are interdisciplinary and require the joint endeavors of researchers in various fields to facilitate the advancement of the transportation system. With the maturity of the blockchain, researchers have found that combining blockchain technology with vehicular networks might be a promising direction. But before that, there are still several issues that need to be settled and some proper improvements and modifications need to be implemented to make blockchain technology more suitable for the IoV systems.

First of all, building a blockchain-enabled IoV system implies that the system needs to be able to process vast amounts of data under extremely unstable connectivity because of the highly dynamic topology of the vehicular network. This feature of the IoV network will slow block propagation of the system and bring higher delay to the transaction confirmation time [102].

Secondly, all the vehicles and devices in the vehicular network have different capabilities and connectivity to implement different functions and resource utilization strategies. This heterogeneity of the vehicles and devices in the vehicular network makes it hard for them to play a completely equivalent role as they were in a traditional blockchain system. Moreover, the vehicles and devices in the vehicular network have different connectivity and resource allocation strategy [103][104].

Thirdly, the computing capabilities and energy supply of the vehicle are not powerful enough to support a computing-based puzzle-solving consensus mechanism like the proof-of-work (PoW) mechanism in Bitcoin or Ethereum to accomplish consistency in the system. Moreover, since the vehicular networks are usually latency-sensitive and regionally activated, vehicles in one city or region do not need to record or verify transactions worldwide which can also bring high latency to the network [105][106].

To this end, a blockchain-enabled IoV system must specify the role of each node in the network according to its capability and operating status. And the configuration of the network should also be flexible enough to adapt to the high mobility of the vehicular network.

3.2 Architecture of the multi-channel blockchain scheme

Compared with other networks, the damage caused by malicious nodes in vehicular networks can be more serious, it is not appropriate to use a completely anonymous system in a vehicle network. It is better to employ a permissioned blockchain framework in an IoV system to ensure access control and improve the identifiability of the participants. Therefore, Hyperledger Fabric, one of the most popular permissioned blockchains, is employed as the blockchain platform in this proposal.

The Kafka ordering services are deployed to implement the consensus mechanism in the proposed scheme. Kafka is a crash fault tolerance (CFT) implementation that uses a “leader and follower” node configuration in which transactions are replicated from the leader node to the follower nodes. When the leader node goes down, one of the followers becomes the leader, so that the ordering process can continue, ensuring fault tolerance [107].

The orderer nodes are always in one of three states: follower, candidate, or leader. All the orderer nodes initially start as a follower. In this state, they can accept log entries from a leader, or cast votes for the leader. If no log entries or heartbeats are received for a set amount of time, nodes self-promote to the candidate state. In the candidate state, nodes request votes from other nodes. If a candidate receives a configurable amount of votes, then it is promoted to a leader.

The leader is responsible for ingesting new log entries, replicating them to follower ordering nodes, and managing when an entry is considered committed. Followers receive the logs from the leader and replicate them deterministically, ensuring that logs remain consistent. The followers also receive “heartbeat” messages from the leader. It is noticeable that every channel on the network can have a separate leader [108].

Usually, the architecture of the blockchain network is a two-layered structure like it is shown in Figure 2-5, the example network of Fabric. The peers and ordering services are running on the cloud layer to ensure computing resources and connectivity. And the clients are running on the user end devices which are usually far from the cloud servers [109]. But since the infrastructure nodes such as RSUs and base stations in the vehicular networks also possess the powerful capability and stable connectivity, they are capable of performing the endorsement and validation execution for the proposed transactions. Moreover, since the infrastructure nodes are usually distributed near the user end, they can respond to the transaction proposals faster than the cloud servers. Thus, the peer nodes of the Fabric network are more suitable to be configured on the infrastructure nodes in the vehicular networks as shown in Figure 3-1.

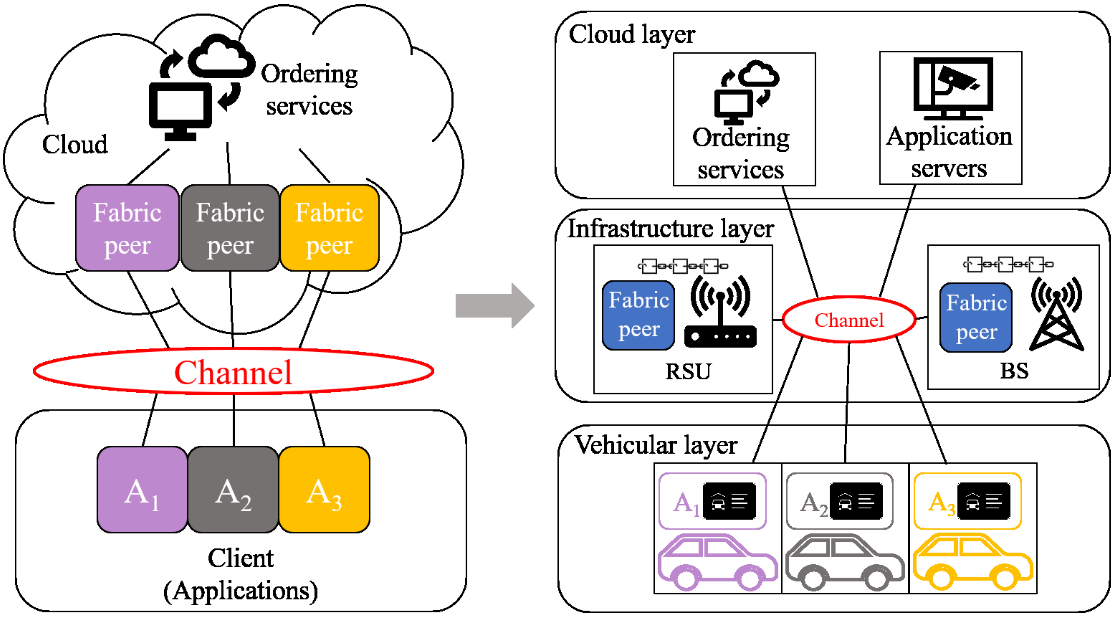


Figure 3-1 The integration of the blockchain technology and IoV system in the proposed scheme.

There are 3 layers in the proposed multi-channel scheme, namely, the cloud layer, infrastructure layer, and vehicular layer as shown in Figure 3-2.

Cloud layer: Some of the important blockchain components such as ordering servers and CA services are deployed in this layer. The ordering servers are used to provide ordering services for the blockchain systems. The orderer peers collect the transactions generated by the vehicular layer and pack them into data blocks in chronological order. All the organizations in the system have their own CA server which is linked to the root CA server and is used to generate signature materials for the members of each organization. The ordering peers in this layer maintain all the ledgers in the system while the application servers will only maintain ledgers generated in the channel they have joined.

Infrastructure layer: The infrastructure layer is composed of RSUs and base stations. Since the nodes in this layer have reliable connectivity and sufficient computing resources, this layer is responsible for the verification and endorsement of the proposed transactions. Since the transaction verification needs the complete version of the blockchain ledger, all the infrastructure layer nodes need to maintain a copy of the ledger locally. The RSUs are also responsible for the SBM message collection. The RSUs will complete the density calculation and inform the vehicles nearby.

Vehicular layer: The vehicular layer is composed of the client nodes. Nodes in this layer do not need to participate in the consensus or validation process, they do not need to have powerful computing resources or stable network connections. They are only responsible for collecting data and generating transaction proposals. Thus all the devices and vehicles can be part of this layer as soon as they can communicate with the others and send SBMs as needed. In the network initialization stage, all the vehicular layer nodes can join multiple channels which are pre-configured in the blockchain system to provide adaptive services for the vehicular layer nodes under various traffic conditions. The channels here serve as private blockchains for the specific participants who have joined the channels, and other nodes which have not joined the channels can not participate in the channel activities or send transactions to these channels [110].

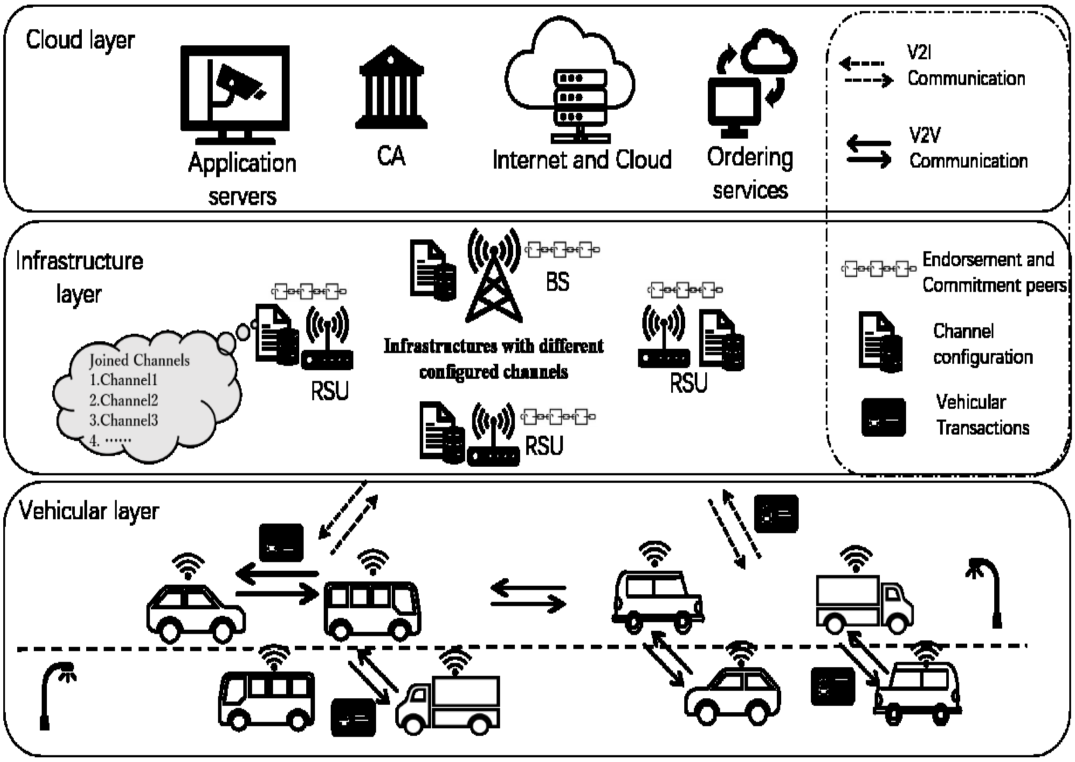


Figure 3-2 Layered architecture of the proposed blockchain scheme.

The infrastructure nodes will monitor the vehicle density continuously and inform the nodes in the vehicular layer. Then the vehicles and devices in the network will select a suitable channel to send the next transaction according to the traffic condition and the message type to achieve better performance [111].

Existing research mostly focuses on how to use the various features of blockchain in the vehicular network, such as trust management or incentive mechanisms. Different from these existing studies, this proposal focuses on the dynamic topology and regional characteristics of the IoV systems and develops a flexible blockchain scheme for the vehicular network to improve the performance of the system under various traffic situations.

3.3 Blockchain setup

Three main processes of the proposal during the blockchain setup are explained below which are network setup, vehicle registration, and transaction lifecycle respectively.

3.3.1 Network setup

During the system initialization, all the components and devices should deploy the corresponding software and play respective roles in the network according to their connectivity and resources.

As shown in Figure 3-3, the vehicles will only install client applications and propose transactions as users in the blockchain network since they do not have powerful computing resources and reliable connections. The infrastructure nodes will be configured as endorsing peers and committing peers. Endorsing peers are used to examining the proposed transactions from the vehicular layer and sign the transactions with their digital signature as endorsements. Committing peers are used to verify the transactions in the blocks received from the ordering services. No matter if the transactions are valid or not, they will all be stored in the blockchain ledger, but only those valid transactions can update the account values in the word state database.

All the infrastructure nodes will be configured to join multiple channels which are suitable for different traffic conditions to provide adaptive selections for the vehicles. All the vehicles can also register multiple channels with the following procedures as needed.

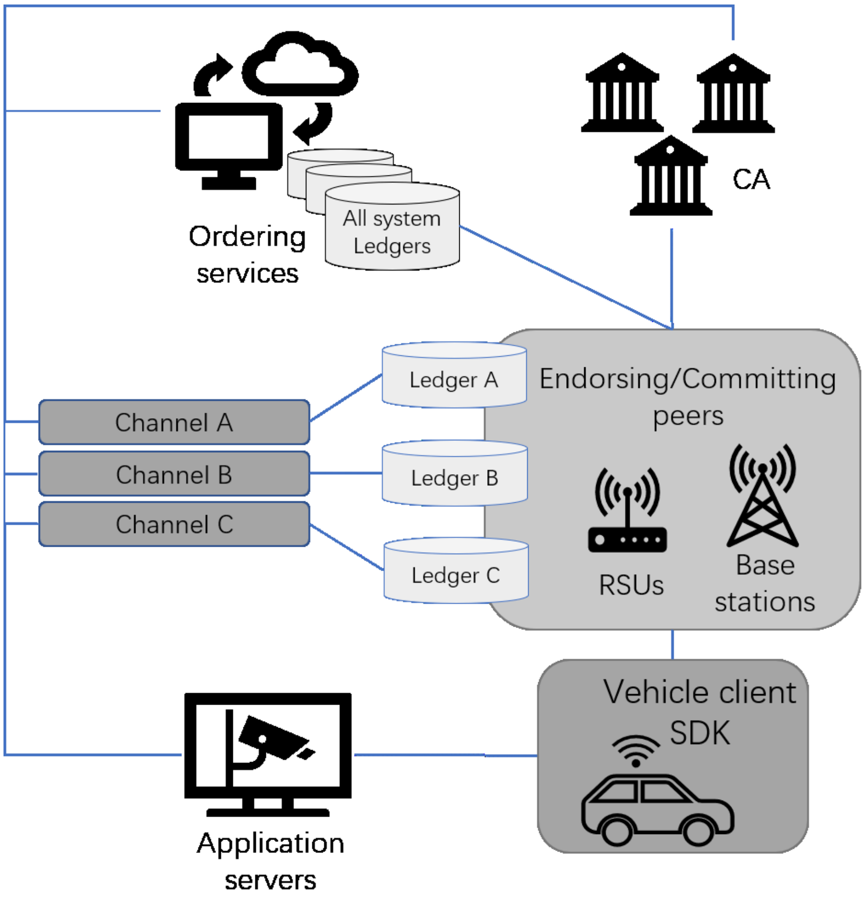


Figure 3-3 Blockchain network setup.

3.3.2 Registration of vehicles

A sequence diagram of the vehicle registration procedure is shown in Figure 3-4. In a Fabric network, every organization has a CA server and the CA server will first initiate and start the services. It is responsible for generating certifications for the members of its organization. Then the infrastructure nodes such as RSUs will enroll in the CA server as an administrator and the CA server will return the corresponding certification (ECert). When an RSU receives a register request from a vehicle, it will register the vehicle in the CA server and get a user-specific secret from the CA server. At last, the vehicle will enroll in the CA server with the registered vehicle information and user-specific secret. When the vehicle gets the transaction certification from the CA server, it will be able to propose transactions to the blockchain networks.

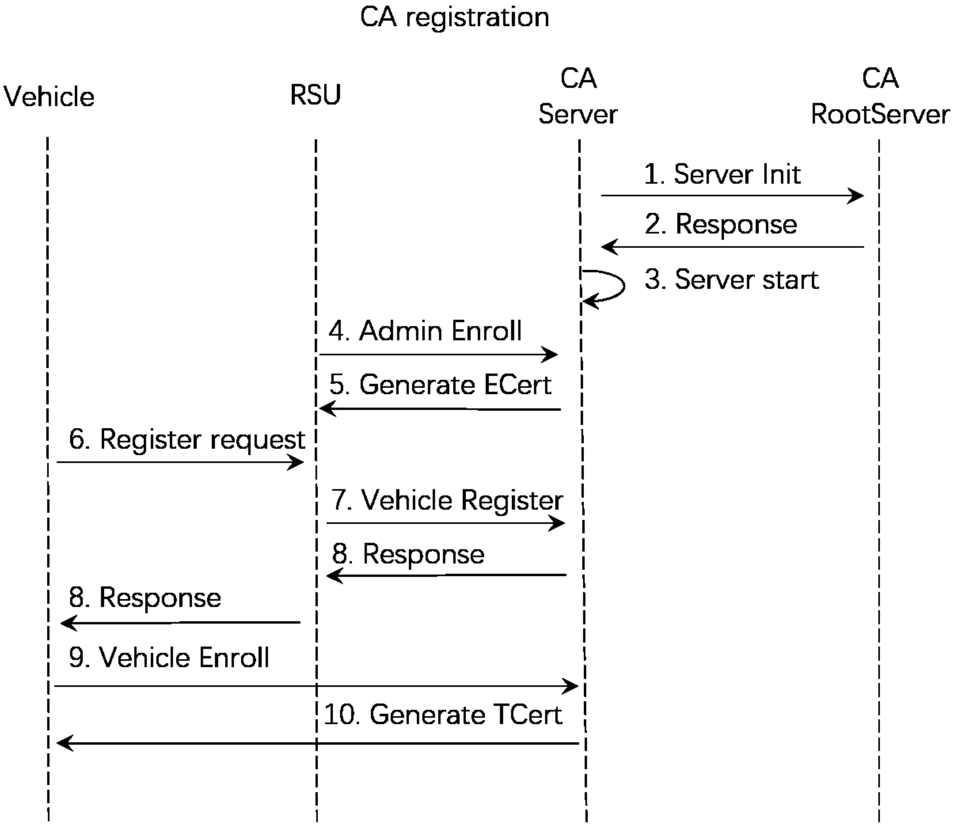


Figure 3-4 Sequence diagram of the vehicle registration procedure.

3.3.3 Transaction lifecycle

The transaction lifecycle is shown in Figure 3-5. There are four stages in the transaction lifecycle, namely, the endorsement stage, ordering stage, verification stage, and commitment stage.

Stage 1: Endorsement stage. When a vehicle wants to send a transaction request in the blockchain network, it needs to initiate a transaction proposal with its signature and send the proposal to the required amount of infrastructure nodes for endorsement. The infrastructure nodes will examine the contents of the proposal and simulate the result of the proposed transaction. If the proposal is appropriate, the RSUs will endorse the proposal with its signature and send the result back to the vehicle. If the vehicle has collected enough endorsements for the proposal, it will pack the original proposal and all the endorsements signed by the RSUs into a new transaction and send it to the ordering services.

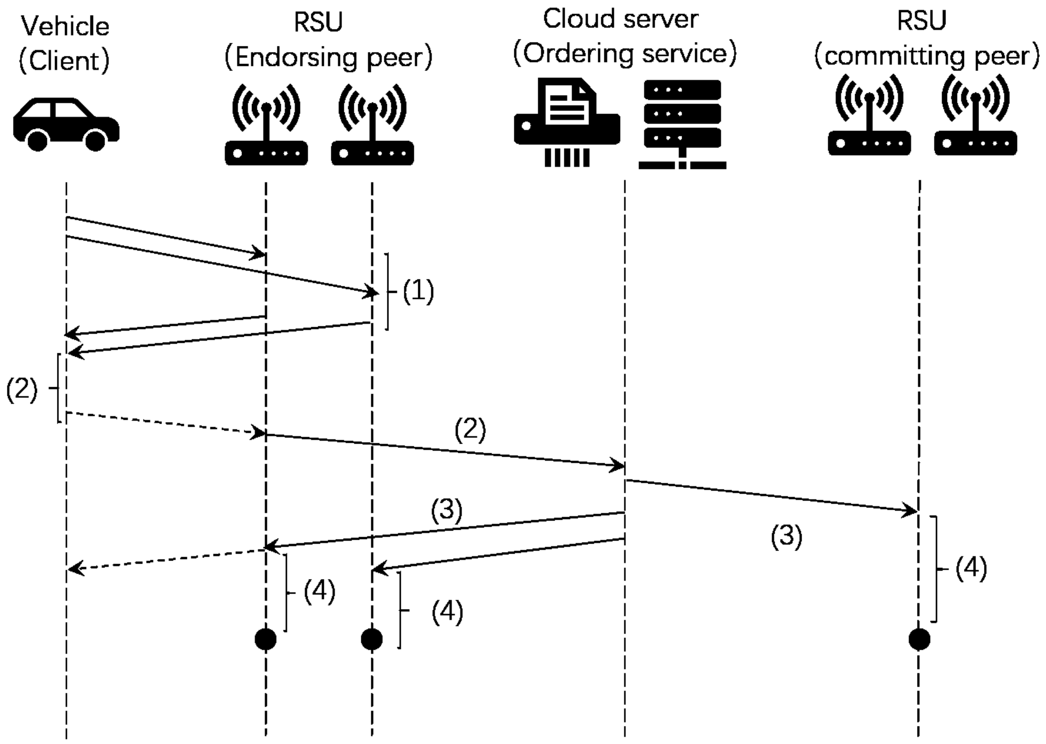


Figure 3-5 Transaction lifecycle.

Stage 2: Ordering stage. The ordering peers will sort the transactions by channels and pack them into blocks chronologically. In Fabric, each channel has an exclusive ledger that is only open to the related nodes to implement data isolation. The ordering services are only responsible for packing the transaction into blocks, they do not pay attention to the content or validity of the packed data. As soon as the number of transactions in the block reaches the configured block size, the ordering peers will disseminate the blocks to the commitment peers for further verification.

Stage 3: Verification stage. When a newly disseminated block arrives, the commitment peers will trigger the validation system chaincode (VSCC) to verify the validity of the transactions in the block. No matter if the transaction is valid or not, it will be stored in the local blockchain ledger.

Stage 4: Commitment stage. If a transaction is marked as valid by the validation process, the commitment peer will apply the read-write set of the transaction (e.g., update of the account balance or modification of the ownership of an asset) to the local world state database. Finally, the commitment peers will emit a notification event to inform the clients about the validity of the transactions.

3.4 The proposed channel selection algorithm

In order to assist the proposal to better service the vehicular networks, a multi-channel selection algorithm is developed and presented in this section. Algorithm 1 gives a summary pseudo-code to illustrate how to select the appropriate channel for the next transaction according to vehicle density and application requirements. The details of the proposed algorithm are explained below.

At the initialization stage, the infrastructure nodes will join multiple pre-configured channels to provide adaptive services for the users in the blockchain system. The RSUs will periodically inform the clients in the vehicular layer about the current traffic conditions.

It is assumed that the transaction generating rate of a vehicle is fixed. For example, the safety beacon message of a vehicle can be regarded as a transaction message, and when the interval of the SBM is 10 seconds, it means that the vehicle generates 1 transaction every 10 seconds [112][113]. Thus, if there are 1,000 vehicles in an area, the transaction generating rate will be 100 TXN/s. Thus, the number of vehicles can be converted into the transaction rate.

According to the assumption above, the performance of a set of different configured channels have been tested, and the most appropriate configuration under different vehicle numbers are recorded as shown in Table 3-1. Six channels with different block size are tested which are *Channel1* with 100 TNX/block, *Channel2* with 200 TNX/block, *Channel3* with 300 TNX/block, *Channel4* with 400 TNX/block, *Channel5* with 500 TNX/block, and *Channel6* with 600 TNX/block.

The reason that there are 6 different channels tested is that the maximum transaction generating capability of the simulation computer is about 200 TNX/s, so it is assumed that the vehicle capacity of the simulated area is 2,000 vehicles. Meanwhile, when the maximum transaction rate is 200 TNX/s, channels with more than 600 transactions per block decrease dramatically. Thereby, only channels with less than 600 transactions per block are tested in this proposal. The interval of the block size is 100 because if the interval is smaller, the differences between channels are not very obvious. If the interval is too big, there will be not enough channels to be selected by the proposed scheme for different situations. Thus, 6 channels are tested for the proposed multi-channel blockchain scheme. However, the number of the channel and the interval can be adjusted according to the actual situation as needed.

Table 3-1 Best channel mapping table

Message Type / Sending rate (T/s)	Throughput-sensitive	Latency-sensitive
100	Channel 5	Channel 5
110	Channel 2	Channel 2
120	Channel 2	Channel 2
130	Channel 3	Channel 3
140	Channel 3	Channel 4
150	Channel 3	Channel 3
160	Channel 2	Channel 2
170	Channel 3	Channel 1
180	Channel 3	Channel 4
190	Channel 4	Channel 1
200	Channel 4	Channel 1

As it is shown in Algorithm 1, the vehicle application will first convert the current number of the vehicles into transaction sending rate, and then select the most suitable channel to send the next transaction according to the sending rate and application requirements. Two kinds of application requirements are considered in this proposed channel selection algorithm which are throughput-sensitive applications and latency-sensitive applications.

Every transaction proposal has a channel ID in its payload which is used to specify the ID of the target channel this proposal is sending to. We can see from Table 3-1 that when the transaction sending rate is about 150 transactions per second, that is to say, the number of vehicles is between 1,500 and 1,600, Channel 3 is the best channel under both the throughput-sensitive situation and latency-sensitive situation. It should be noted that when the sending rate is lower than 100 transactions per second, which is equivalent to the number of vehicles is below 1,000, the performance of all the channels is similar, therefore, the best channel for the number of vehicles below 1,100 are uniformly set as channel 5 for convenience. Similarly, Channel 4 (throughput-sensitive situation) and *Channel1* (latency-sensitive situation) are set as the best channel when the number of vehicles is larger than 2,000.

Algorithm 1: Channel selection algorithm

```
Input: Current number of vehicles $D_0$ and application type $M_t$
Output: Selected ChannelID
Import the Channel Selection Table
Calculate the sending rate of the vehicle transactions $R$ as follows:
    $R = \begin{cases} 100, & if\ D_0 < 1{,}100 \\ 110, & if\ 1{,}100 \le D_0 < 1{,}200 \\ 120, & if\ 1{,}200 \le D_0 < 1{,}300 \\ 130, & if\ 1{,}300 \le D_0 < 1{,}400 \\ 140, & if\ 1{,}400 \le D_0 < 1{,}500 \\ 150, & if\ 1{,}500 \le D_0 < 1{,}600 \\ 160, & if\ 1{,}600 \le D_0 < 1{,}700 \\ 170, & if\ 1{,}700 \le D_0 < 1{,}800 \\ 180, & if\ 1{,}800 \le D_0 < 1{,}900 \\ 190, & if\ 1{,}900 \le D_0 < 1{,}200 \\ 200, & if\ D_0 \ge 2{,}000 \end{cases}$
//Set BestChannel according to the Channel Selection Table.
if Application type $M_t$ is "throughput-sensitive" then
    Switch (R)
    Case 100: BestChannel ← Channel5; Break;
    Case 110: BestChannel ← Channel2; Break;
    Case 120: BestChannel ← Channel2; Break;
    …
    Case 200: BestChannel ← Channel4; Break;
else if Application type $M_t$ is "latency-sensitive" then
    Switch (R)
    Case 100: BestChannel ← Channel5; Break;
    Case 110: BestChannel ← Channel2; Break;
    Case 120: BestChannel ← Channel2; Break;
    …
    Case 200: BestChannel ← Channel1; Break;
end if
ChannelID ← BestChannel
return
```

3.5 The generality of the proposed scheme

From the blockchain perspective, the proposed blockchain scheme and the channel management algorithm only provide a designing idea of adaptive block size for the application of the blockchain technology in the IoV system under different traffic conditions and does not involve any modifications of the consensus mechanism or data structure which make the blockchain platforms different from each other, thus, the proposed scheme is generally applicable in different kinds of blockchain platforms, including consortium blockchains, such as Hyperledger Fabric deployed in this thesis and public blockchains such as Ethereum.

From the application scenario perspective, since the proposed scheme is designed for the application of the blockchain technology in scenarios with highly dynamic topology and frequent data exchange, and does not require powerful capabilities from the terminal equipment, it is only suitable for the IoV system, but also suitable for other scenarios with the similar characteristics such as mobile networks.

3.6 Simulation design

3.6.1 Simulation tool

The performance of the proposed blockchain multi-channel scheme is evaluated through extensional simulations. The simulations are performed on a desktop with Intel i3-8100 CPU, 8GB RAM, and GeForce 1050 Ti graphics card. Hyperledger Fabric 1.4 with Kafka ordering service is deployed in the simulations and the world state database is set to LevelDB.

Hyperledger Caliper, a popular blockchain system benchmarking tool, is used to generate transaction workloads for the proposed scheme. Since the workloads produced by Caliper are actual blockchain transactions, the performance metrics will be more realistic than the metrics measured through a blockchain simulator. Figure 3-6 shows a simple working procedure of the Caliper benchmark.

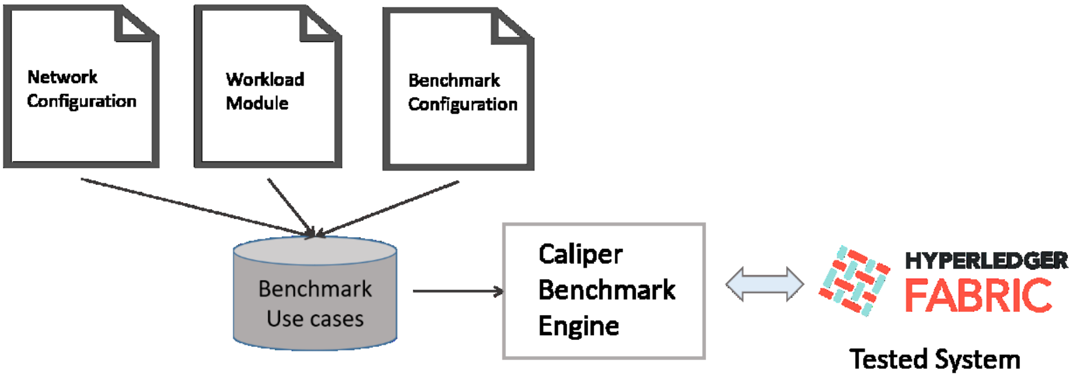


Figure 3-6 Working procedure of the Caliper benchmark.

3.6.2 Simulation set up

The simulated topological area is 1,000 * 1,000m, with three horizontal roads and three vertical roads which form 4 squares in it. There is at least one infrastructure node in each square so that any vehicle or device in this area can communicate with at least one of the infrastructure nodes. It is assumed that the vehicle capacity is 2,000 in this area.

Since the applications in the vehicular networks are configured with various functionality purposes, they also focus on different performance metrics[114]. For instance, accident and traffic condition applications are usually latency-sensitive, and data sharing applications are usually throughput-sensitive. Thus, two different application scenarios, namely, throughput-sensitive scenario and latency-sensitive scenario are simulated to evaluate the performance of the proposal for different application demands.

3.7 Simulation results

There are different types of transactions in the blockchain system which are designed for different purposes like registration transactions, query transactions, transfer transactions, system configuration transactions, and so on. Registration transactions are used to register the vehicle in the system and initialize the account balance of the vehicle. The query transactions are used to inquire account value or ledger status of the system. Transfer transactions help transfer the capital or ownership of an asset among the users. The system configuration transactions are used to update the system configurations.

Since the configuration transactions are not often used and the query transactions have no actual operation in the system, two types of transactions are tested in the extensional simulations, which are the registration transaction and the transfer transaction, to evaluate the impact of the proposal on different procedures of the blockchain system.

Many metrics need to be considered when evaluating a blockchain system, such as throughput, latency, success ratio, fault-tolerance, scalability, resource consumption, and construction cost [115]. Since the proposed scheme does not involve the improvement of consensus mechanism or resource management of the system, we will evaluate the proposed scheme from basic network metrics. Therefore, the proposed scheme is evaluated through three different metrics of the blockchain system which are transaction throughput, transaction commitment latency, and transaction success ratio.

The throughput of a blockchain system represents the number of transactions this blockchain system can handle per second. It is equal to the number of valid transactions divided by the elapsed time.

$$Throughput = \frac{Valid\ \ transactions}{Elapsed\ \ time} \tag{1.}$$

The transaction latency represents the amount of time to wait for a transaction to be validated after it is proposed to the system. It is measured through the difference between the time for a transaction being proposed and the time that the transaction has been committed.

$$Latency = \ Committed\ \ time\ \ -\ Proposed\ \ time \tag{2.}$$

The transaction success ratio is used to measure the transactions marked as valid by the committed peers and successfully applied on the user account value. It is calculated as the number of successful transactions divided by the total number of transactions.

$$Success\ \ ratio = \frac{Valid\ \ transactions}{All\ \ transactions} \tag{3.}$$

The change of vehicle numbers is simulated by varying the sending rate of the transaction workload. The varying range of all the simulations in Chapter 3 is from 100 to 200 transactions per second, which is equal to the number of vehicles changing from 1,000 to 2,000.

The performance of the original Fabric with a single channel and fixed block size, which are 100, 300, 600 TNX/block respectively, are given in the results as baselines to compare with the performance of the proposed blockchain scheme with multiple channels and adaptive block size. By comparing the performance metrics like throughput, latency, and success ratio of the proposed scheme to the original Hyperledger Fabric, we can find out the merits of the proposal and how much the proposal can improve the performance of the blockchain-enabled IoV system under different traffic conditions.

3.7.1 Throughput-sensitive scenarios

In a throughput-sensitive scenario, the throughput of the system performance is more important for the client application installed on the vehicles or devices. Thereby, the proposed channel selection algorithm will select the channel with the best throughput performance under the current traffic condition as the target channel for the next transaction.

3.7.1.1 Registration transactions

The result of the system performance with registration transactions is presented first in Figure 3-7 and Figure 3-8. The throughput of the system increases along with the increasing number of vehicles until 1,400 vehicles, namely, 140 transactions per second. This point is called saturation point which means it is the maximum transaction processing speed under the current system configuration. After that, the system throughput stops increasing with the number of vehicles and starts fluctuating around the saturation point. When the number of vehicles increases before the saturation point, the proposal uses a smaller block size to maximize the system resource utilization. Because the fewer transactions in a block, the faster a block will be formed. However, this will form more blocks for the same amount of transactions, thereby, more network overhead. After the number of vehicles reaches saturation point, the proposed scheme changes to a larger block size to reduce the system overhead. It is shown in the result that the throughput of the proposal performs the best compared to the other baselines under all different transaction rate simulations.

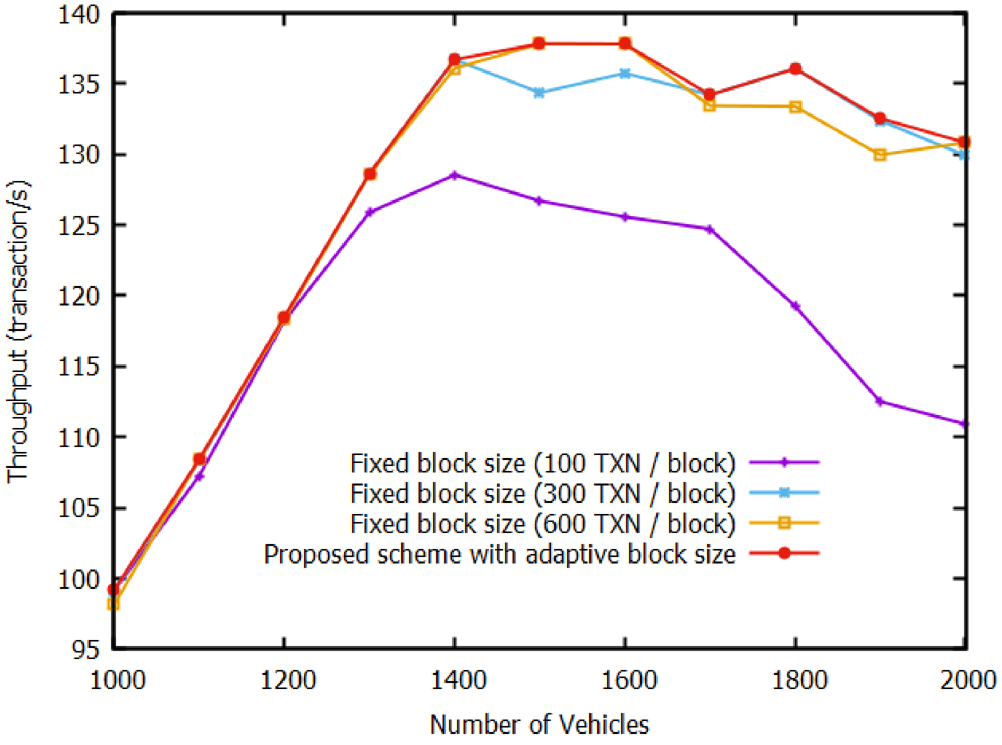


Figure 3-7 Throughput of registration transactions in throughput-sensitive scenarios.

In contrast, the difference in delay among the proposal and some of the baselines with different block sizes is not very obvious. It is shown in Figure 3-8 that the latency of the results is quite low when the number of vehicles is low. After the number of vehicles reaches the saturation point, the delay begins to increase sharply. This is because the system can only process 140 transactions per second, and the extra transactions can only be queued. As the sending rate increases, the waiting time becomes longer and the delay becomes higher and higher. Although the priority metric in this scenario is throughput, the delay of the proposal still performs slightly better than the other baselines.

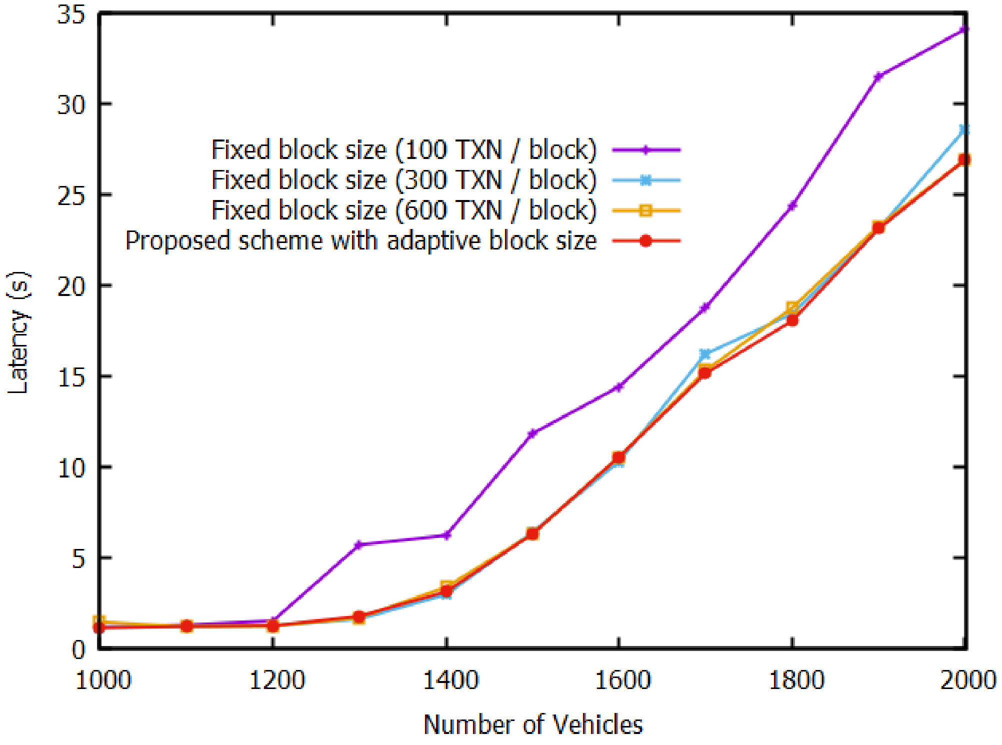


Figure 3-8 Latency of registration transactions in throughput-sensitive scenarios.

3.7.1.2 Transfer transactions

The performance of the system with transfer transactions is shown in Figure 3-9, Figure 3-10, and Figure 3-11. Only valid transactions are taken into account when calculating the throughput and success ratio of the system performance. The throughput is equal to the number of valid transactions per second, and the success ratio is equal to the number of valid transactions over the total number of transactions. Thus, these two metrics are proportional to each other at the same sending rate.

We can see from Figure 3-9 and Figure 3-10 that the throughput and the success ratio of the system drop sharply after the number of vehicles exceeds the saturation point. This is because when the waiting line grows after the saturation point, a transaction may have waited too long that the account value it is going to update has already been changed, and thereby, this transaction will be marked as invalid by the multi-version concurrency control (MVCC) verification process [116]. The MVCC mechanism is used to ensure the concurrency of the value of the world state database. If the value has changed during the transaction lifecycle, the transaction must be proposed again with the new version of the value to prevent double-spending. It is shown in the results that the proposal performs the best compared to the baselines in both throughput and success ratio metrics.

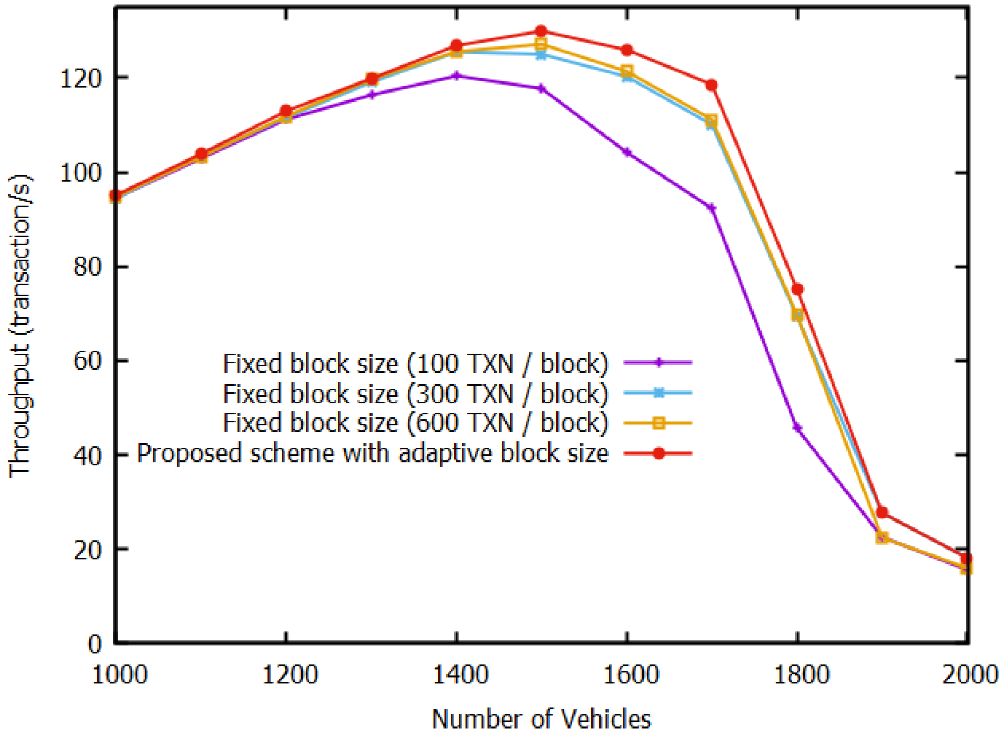


Figure 3-9 Throughput of transfer transactions in throughput-sensitive scenarios.

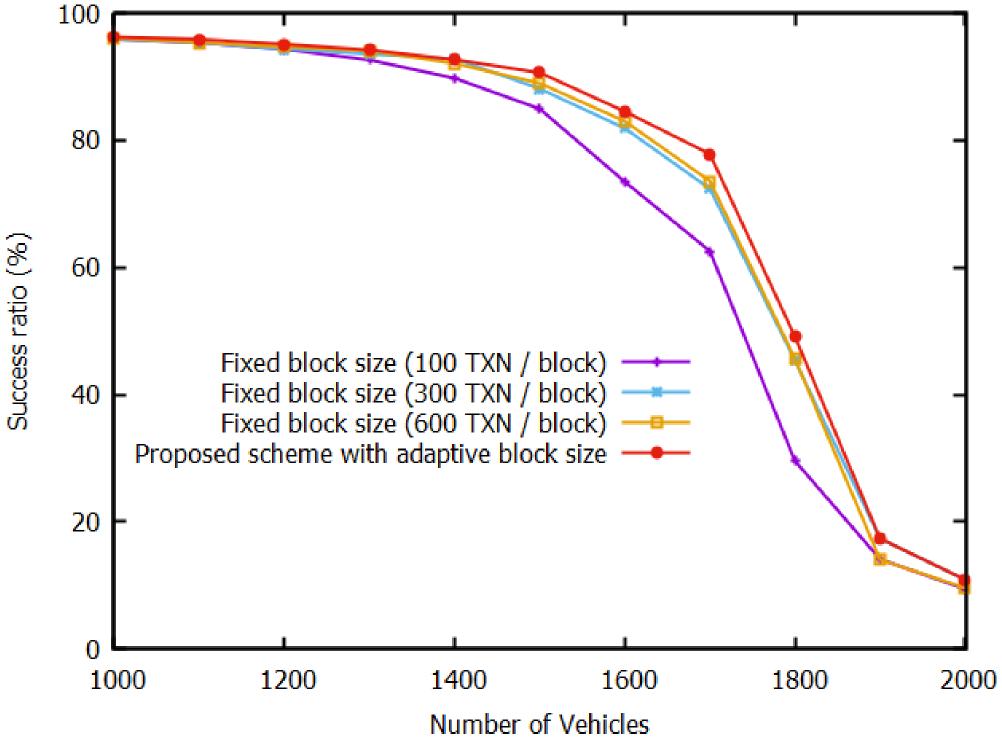


Figure 3-10 Success ratio of transfer transactions in throughput-sensitive scenarios.

The latency performance under the throughput scenario is shown in Figure 3-11. The latency of the system in all the simulations increases slowly before the number of vehicles reaches 1,400. After that, the latency begins to rise dramatically. This is because the number of transactions that are waiting to be processed is also increasing dramatically along with the number of vehicles. It is shown that the performance of the proposal keeps staying at a low level before the number of vehicles reaches 1,700. After that, the latency performance under all different situations does not keep increasing and begins to decrease sharply. This is because of the batch timeout mechanism of the system. Generally, the ordering services will pack the block when the number of transactions in the block reaches the configured block size. But when the transaction arriving rate is too low or network congestion happens, the ordering service will cut the line and pack the transactions into the block within a certain interval no matter how many transactions are in the block. So the latency of the transactions in this block will be lower than the former ones. This is also applicable in latency-sensitive scenarios.

Since the throughput is the priority in this scenario, it is hard to ensure latency performance at the same time. This is a tradeoff process. However, the latency of the proposal still performs lower than most of the baselines when the number of vehicles is not very high.

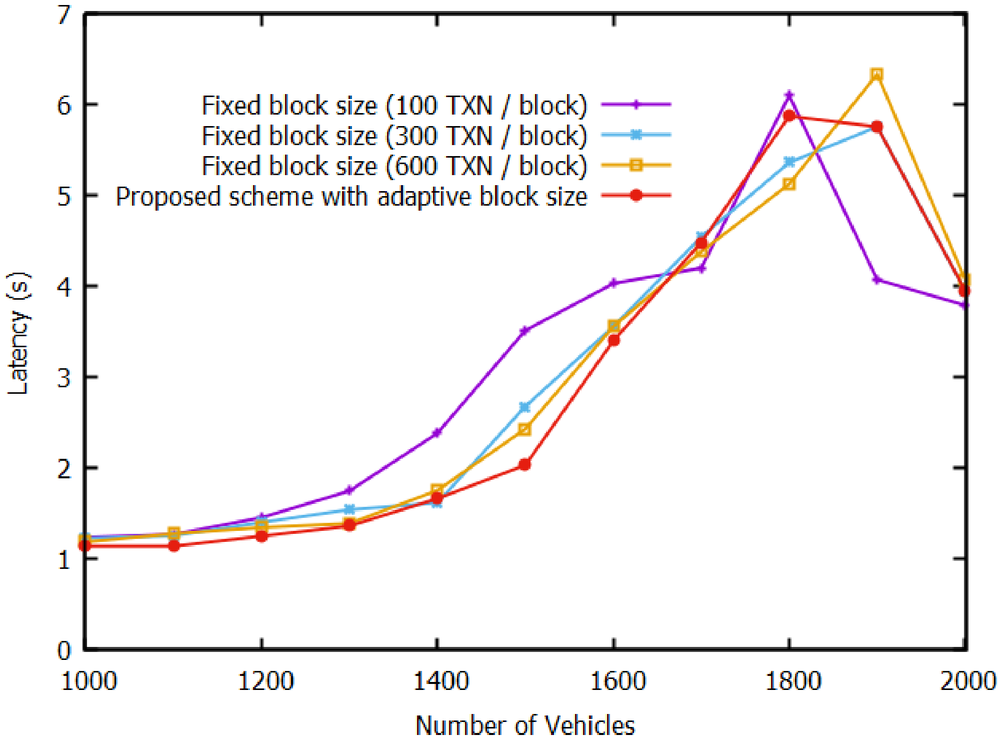


Figure 3-11 Latency of transfer transactions in throughput-sensitive scenarios.

3.7.2 Latency-sensitive scenarios

All the scenarios in this section consider the delay performance of the system as a priority and the proposal will help the vehicles select the channel with the shortest delay.

3.7.2.1 Registration transactions

The performance of the system with registration transactions is shown in Figure 3-12 and Figure 3-13. Similar to the former results, the latency of the system begins to rise dramatically after the number of vehicles surpasses 1,400. The performance of the proposal keeps staying at the lowest among all the simulations. Since the registration transaction does not involve the transfer of the account value among different accounts, there will be no double-spending problems. Therefore, the MVCC verification will not be triggered and no transactions will be marked invalid. Thereby, the throughput performance of the system will not decrease significantly as those in transfer transactions.

It is shown in the results that the latency performance of the proposal is better than all the other baselines. However, as shown in Figure 3-13, the throughput performance of the proposal is not always the best since the latency is the priority in the latency-sensitive scenario and we need to sacrifice the throughput to achieve a better latency performance.

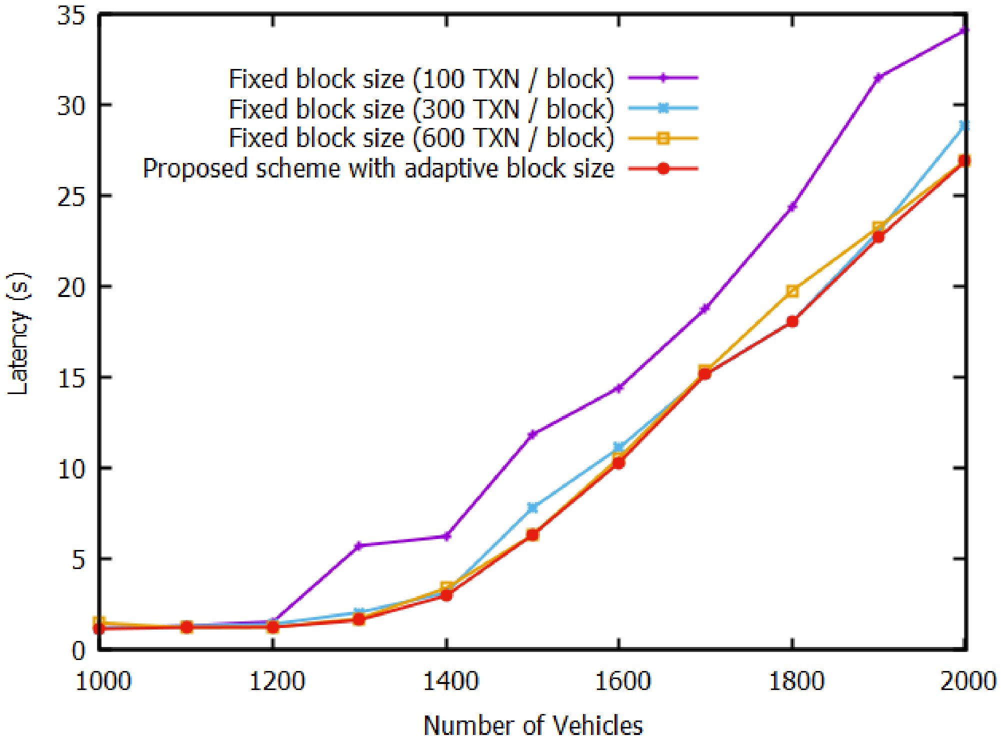


Figure 3-12 Latency of registration transactions in latency-sensitive scenarios.

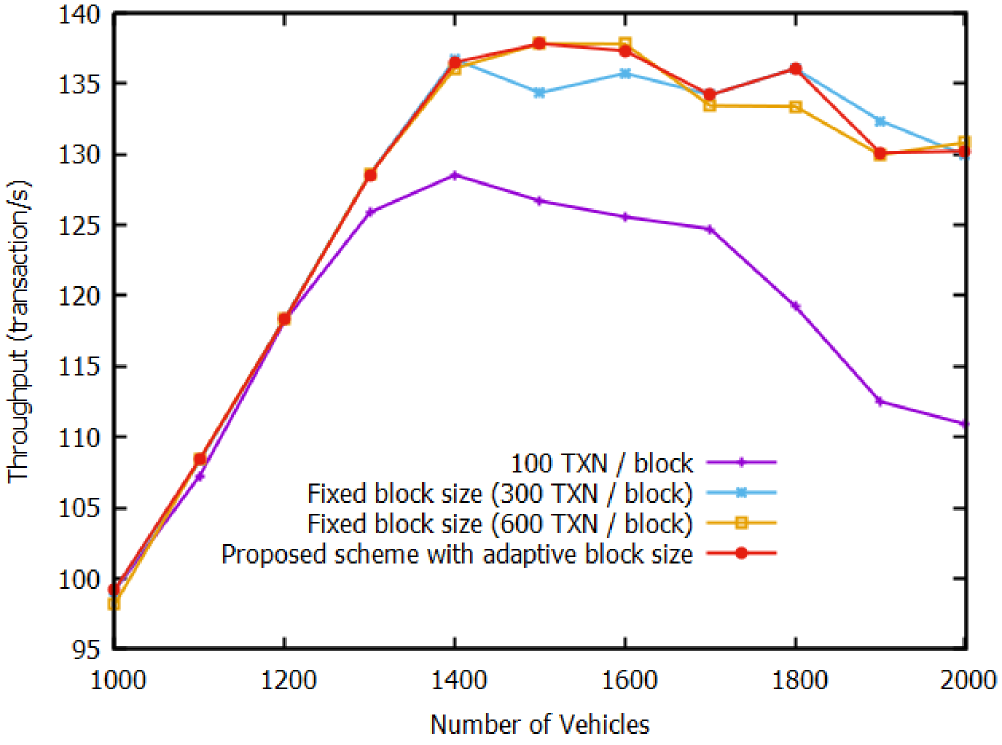


Figure 3-13 Throughput of registration transactions in latency-sensitive scenarios.

3.7.2.2 Transfer transactions

The latency performance of the transfer transaction under the latency-sensitive scenario is shown in Figure 3-14. The performance of the system increases along with the increasing number of vehicles until about 1,800 and drops quickly after that. The reason is similar to the one in the throughput-sensitive scenarios that the batch timeout has been triggered. We can see that the proposal can greatly reduce the latency of the system under transfer transactions and ensure that the next transaction is sent to the channel with the shortest latency.

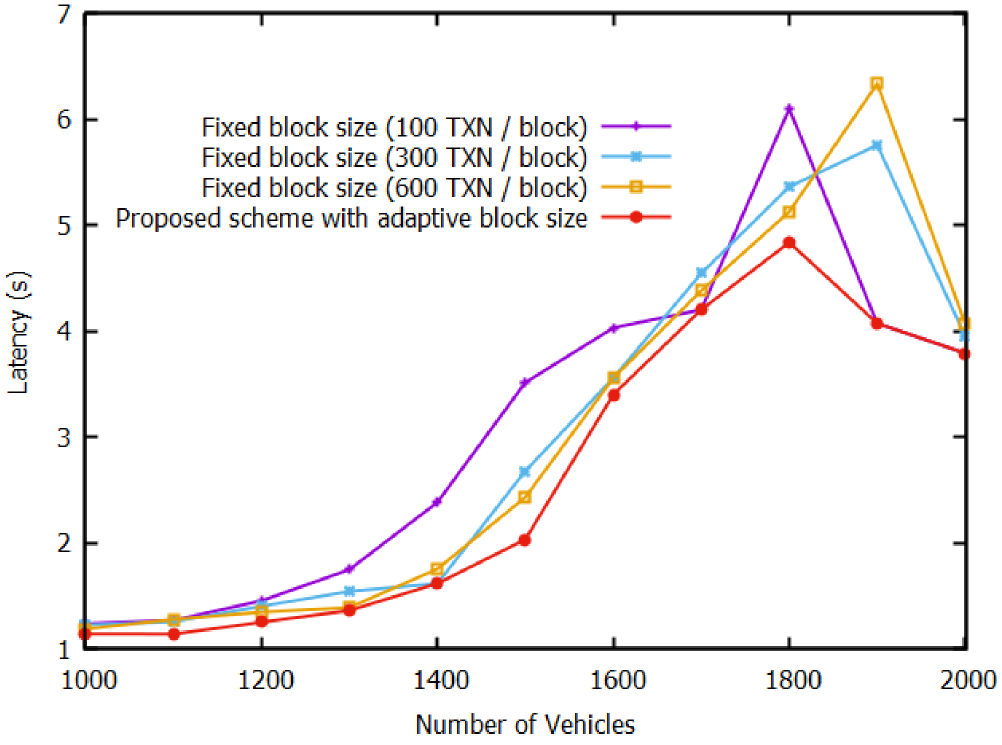


Figure 3-14 Latency of transfer transactions in latency-sensitive scenarios.

The throughput and success ratio of the transfer transaction under the different number of vehicles are shown in Figure 3-15 and Figure 3-16. Since latency is the priority in this scenario, the throughput and success ratio metrics are both sacrificed at some point to ensure the next transaction is sent to the channel with the lowest latency. However, the throughput and success ratio still perform better than the other baselines before the number of vehicles exceeds 1,600.

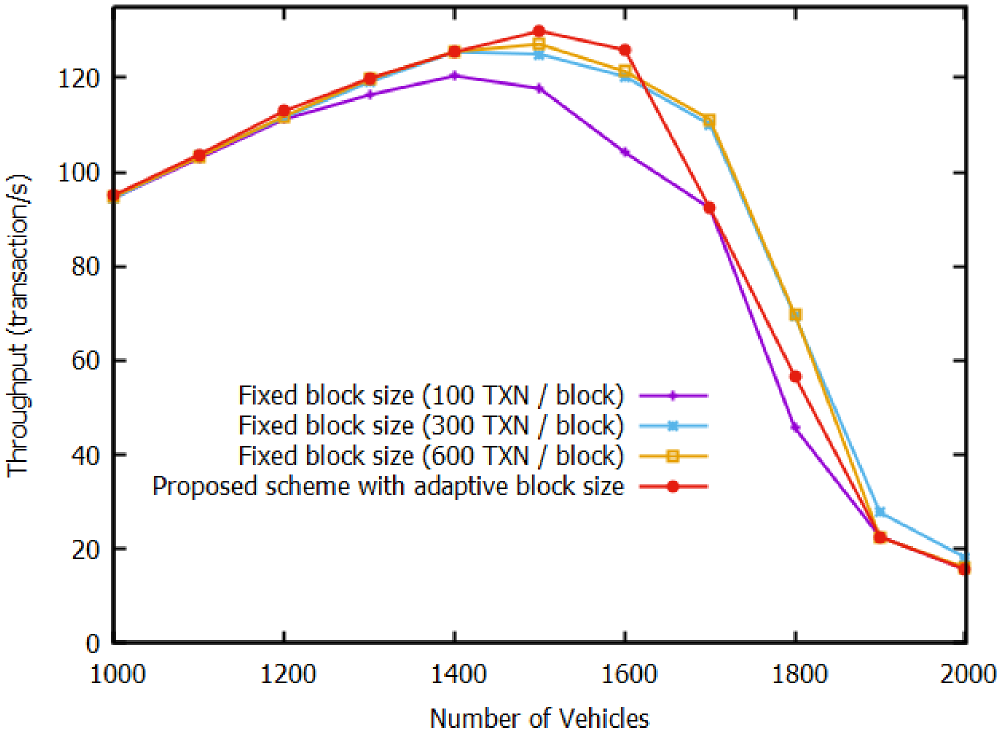


Figure 3-15 Throughput of transfer transactions in latency-sensitive scenarios.

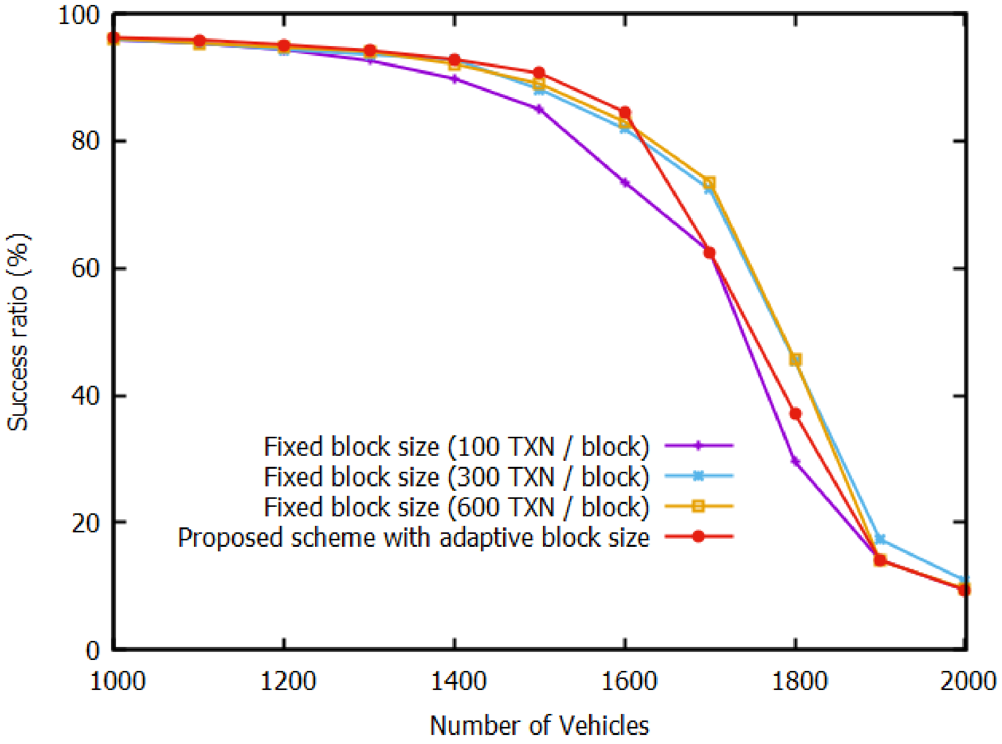


Figure 3-16 Success ratio of transfer transactions in latency-sensitive scenarios.

3.8 Conclusion

This chapter proposes a multi-channel blockchain scheme for IoV where each channel is optimized for a certain level of vehicle density and application requirements. It first discusses the problems of the traditional vehicular networks and why it needs blockchain technology and then introduces the architecture of the proposed multi-channel blockchain scheme for IoV followed by the set-up procedures and transaction flow of the system.

To find the best configuration under different circumstances, the performance of the deployed blockchain system with different parameters under different vehicular densities is investigated. Then multiple channels are configured in advance during the system setup stage to provide adaptive services for the vehicular networks under different traffic conditions. A channel selection algorithm is also proposed in this chapter to cooperate with the proposed blockchain scheme. This algorithm helps the vehicles in the network to select the most suitable channel for the transactions according to vehicle density and application requirements.

The generality is discussed in section 3.5. The simulation design including the simulation assumptions and the environmental settings is introduced in section 3.6. The configuration of the blockchain workload tool is also presented in this section. In the following section, the results of the simulations are presented to evaluate the performance of the proposed multi-channel blockchain scheme.

The simulation of the proposed scheme is conducted under the Hyperledger Fabric by varying values assigned to configurable parameters that simulate the change of the vehicle numbers. Extensional simulations show that the proposed scheme can significantly increase the performance of the blockchain system under the different number of vehicles in terms of the throughput, latency, and transaction success ratio.

CHAPTER 4

4 Hierarchical blockchain resource scheduling scheme for IoV

In this chapter, the proposed hierarchical resource scheduling scheme for blockchain-enabled IoV systems is presented. The proposed scheme improves the performance of blockchain-enabled IoV systems by efficiently allocating computing resources with the proposed resource control algorithm and scaling control algorithm. A resource monitoring system is developed to cooperate with the above algorithms to implement the resource scheduling of the system. The effectiveness of the proposed scheme is fully demonstrated by comparing it with the existing baseline structure. The rest of this chapter is organized as follows.

Section 4.1 discusses the problems of the traditional blockchain system and features of the IoV network that may affect the performance of blockchain systems from several different aspects.

Section 4.2 presents the architecture of the proposed hierarchical resource scheduling scheme. This section also introduces the layering principles in the proposed blockchain resource scheduling scheme.

Section 4.3 presents an overview of the proposed resource scheduling scheme. It gives the general structure and the working flow of the proposed resource scheduling scheme. The priority of resource utilization is also discussed.

In section 4.4, the mechanics of the proposed resource control algorithm and scaling control algorithm to implement the resource scheduling are elaborated. The design of the proposed resource monitoring system is also introduced in this section.

Section 4.5 discusses the generality of the proposed scheme and in section 4.6, the simulation design including the simulation assumptions and the environmental settings is introduced. The configuration of the blockchain workload tool is also presented in this section. The experimental results of the system under different scenarios are presented in section 4.7 to evaluate the effectiveness of the proposed hierarchical resource scheduling scheme and section 4.8 gives a conclusion of this chapter.

4.1 Problem discussion

The development of cryptocurrencies has taken off in recent years which led to more resources being devoted to cryptocurrency mining. The latest data from the Cambridge Bitcoin Electricity Consumption Index (CBECI) shows that Bitcoin mining is expected to consume 133.68 terawatt-hours of electricity per year (1 terawatt-hour is 1 billion kWh). This figure has surpassed Sweden's electricity consumption, ranking 27th in the world's electricity consumption.

Mining is the only way to create cryptocurrencies in most of the popular blockchain platforms. It is deliberately designed to consume so much power at the consensus process to ensure the security of the system by making the cost of the attacker higher than the benefit of the attack.

But as more and more policies on energy conservation, emission reduction, and mitigation of greenhouse gas effects are released, researchers have begun to abandon the traditional computing power competition way, and start looking for a more efficient mechanism for running blockchain systems [117].

Besides, in addition to the consensus mechanism, other processes in the blockchain system also need to consume a lot of computing resources, such as the verification and encryption process.

Moreover, none of the existing blockchain platforms can fully satisfy the current throughput requirements from any of the large-scale application scenarios like credit card or mobile payment platforms. In addition, with the innovation of related applications such as digital currency and decentralized markets, throughput requirements for blockchain systems will be further accelerated [118][119].

Finally, the change in vehicle distribution will also make the load in the system unbalanced, which will cause some areas to be overloaded, and some areas will run underload which results in lower performance and decreases resource utilization efficiency [120].

Therefore, it is necessary to improve the resource utilization efficiency of the blockchain system under different traffic conditions and provide a flexible network scaling mechanism to reduce system construction and operating costs and improve system performance.

4.2 Architecture of the proposed resource scheduling scheme

The proposed blockchain resource scheduling scheme will be discussed in detail in this section.

4.2.1 Architecture overview

Since blockchain systems are usually computing-intensive, the performance of the system can vary significantly according to the computing ability of the infrastructure and the computing load weight running on the infrastructure including blockchain service itself and coexisting services on the server. Thereby, we design a monitoring system to achieve the working condition of the system and develop a hierarchical resource scheduling scheme for the IoV system to make the best use of the system resources. The proposed scheme adjusts the resource allocation at 3 different layers, which are the blockchain service layer, infrastructure layer, and network layer to balance the workloads across the network.

Moreover, in vehicular networks, it is critical to guarantee the safety of every single node. In order to reduce the possibility of malicious nodes occurring, all the vehicles should be identifiable [121][113][122]. Therefore, permissioned blockchain architecture is more suitable for vehicular networks due to its access control mechanism.

There are two main blockchain transaction models in permissioned blockchain architecture: Execute-Order-Validate (EOV) and Order-Execute (OE). For example, Hyperledger Fabric, one of the most popular blockchain platforms, follows the EOV model, and Quorum, another popular permissioned blockchain platform funded by Ethereum, follows the OE model [123]. In this paper, since we are exploring the performance of the blockchain-based IoV system, and the EOV model has higher performance on a large number of peers scenarios, we focus on the EOV model, hence, in this proposal, Hyperledger Fabric is deployed as blockchain platform under test.

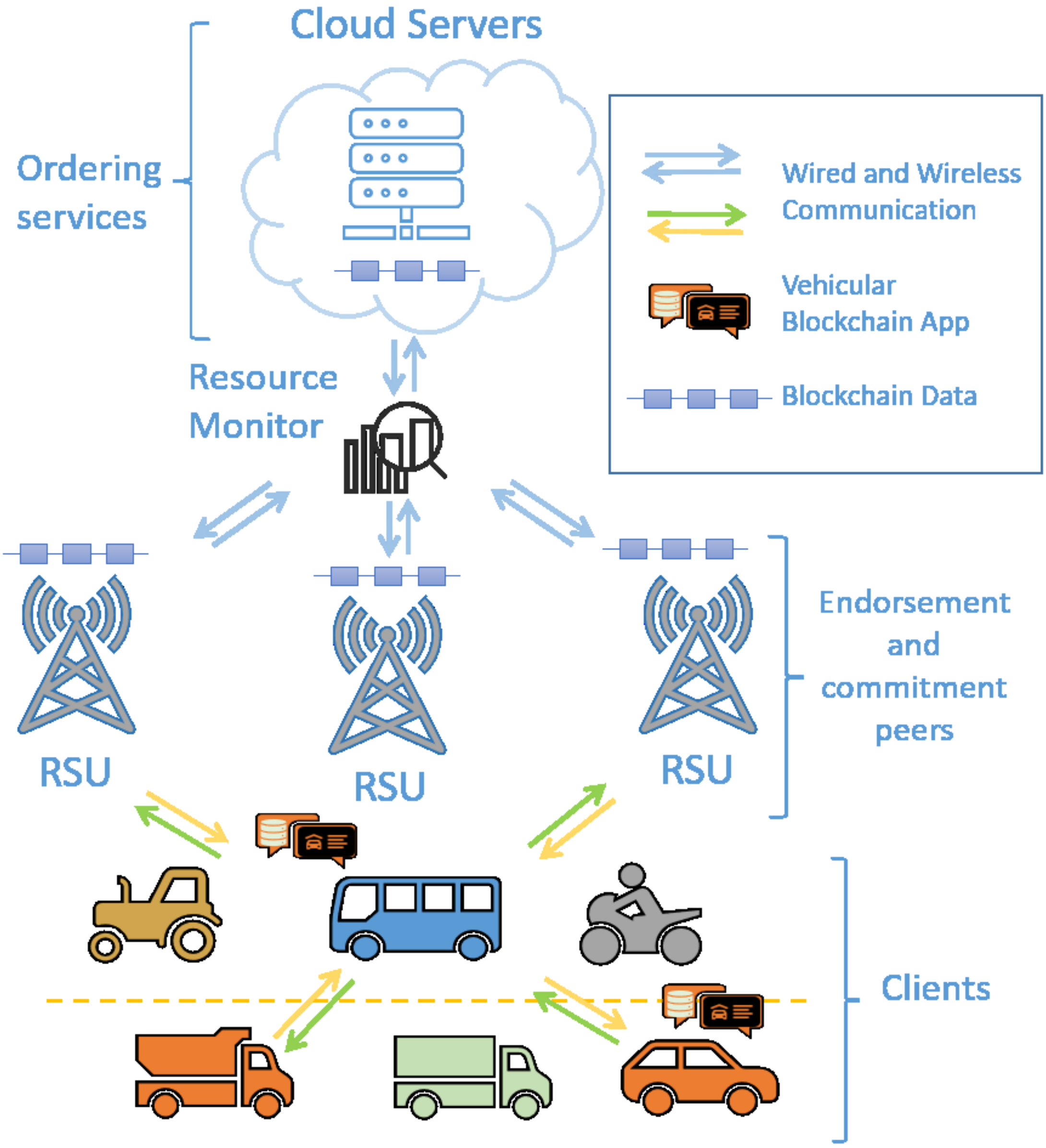


Figure 4-1 Architecture of the proposed blockchain resource scheduling scheme.

The architecture of the proposed scheme is an extension of the blockchain scheme from Chapter 3. The computation-intensive components of the blockchain system are configured to be running on powerful equipment and devices such as RSUs and base stations. The Raft ordering service provided by the Fabric project is selected as the consensus mechanism since the Kafka ordering service used in Chapter 3 is deprecated in Fabric v2.x.

The Raft ordering service is also a crash fault tolerance (CFT) implementation that uses a “leader and follower” node configuration, the same as the Kafka ordering services used in

Chapter 3. Raft ordering service is easier to set up and manage than Kafka-based ordering services, and its design allows different organizations to contribute nodes to a distributed ordering service. This means the ordering service in the IoV network can be provided by many different organizations or enterprises and provide a higher degree of decentralization and autonomy.

The proposed system architecture is shown in Figure 4-1. There are three main components in the system which are clients, endorsement and commitment peers, and ordering services.

Ordering services: The ordering services are made up of ordering peers provided by different organizations in the blockchain systems and can be distributed on different cloud servers. They are responsible for the ordering process of the EOV models like sorting and packing all transactions into contiguous blocks in chronological order and broadcasting the blocks to the commitment peers to be verified. The ordering service does not check whether the data in the block are valid or not. The ordering peers store complete blockchain data.

Peers: The endorsement and commitment peers are deployed on infrastructures like base stations or RSUs in the vehicular network. Endorsement peers are responsible for the execution process of the EOV models like verifying and endorsing the transactions proposed by the vehicular blockchain applications from the vehicles. Commitment peers are responsible for the validation process of the EOV models like checking the conflicts in the blocks received from ordering services and adding the new received block into the ledger. Each infrastructure node in the network will set up a pre-configured backup node for sharing its spare resources to the network and help balance the workloads in the system. The backup node can be activated periodically to synchronize with the system.

Clients: The vehicles and smart devices in the system only utilize the system as clients and are responsible for generating transactions with vehicular blockchain applications. The vehicles here do not participate in the consensus process of the blockchain system since the resources and connectivity of the vehicles are usually limited and unstable.

4.2.2 Resource distribution and layering principles

The system resource distribution of the system is shown in Figure 4-2. It is divided into different layers to better support the system and improve the efficiency of the resource utilization of the system. The layers are divided as follows:

Service layer: As we can see that every infrastructure node allocates a percentage of resources to the blockchain service in advance. These resources allocated to the blockchain services can be regarded as the blockchain service layer which is isolated from other services to prevent interference among services running on the infrastructure node.

Infrastructure layer: The rest of the resources on the infrastructure node constitute the infrastructure layer. When the resources of the blockchain service layer run out, the proposed scheme can apply for more resources from the local infrastructure node to improve the performance of the system.

Network layer: The network layer is composed of all the resources in the network. When the resource utilization of an infrastructure node is at a low level, and another infrastructure node is running out of resources, the pre-configured fabric backup peer can be activated to support the others in the network.

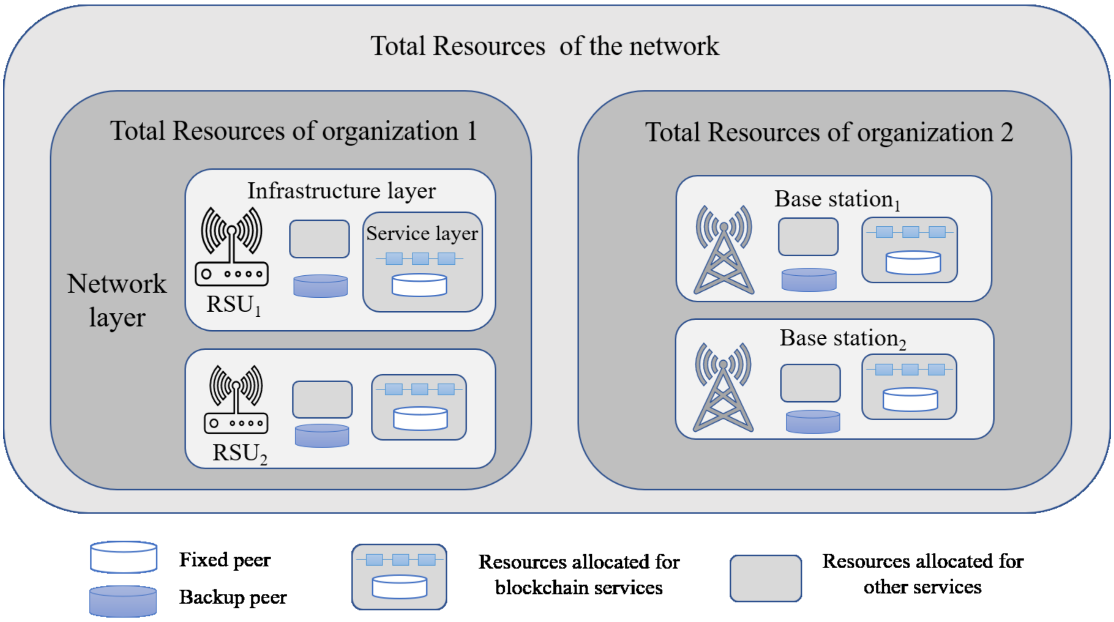


Figure 4-2 Resources distribution and layering of the system.

4.3 Proposed resource scheduling scheme overview

This section presents an overview of the proposed resource scheduling scheme. It introduces the general structure of the scheme and the priority of resource utilization. The working procedure of the proposed resource scheduling scheme is also presented.

4.3.1 Priority of the resource utilization

The priority of the resource utilization follows the service-infrastructure-network order as it is shown in Figure 4-3. When the system is running under pressure, it will first check the condition of its own allocated resources. If there are no spare resources left in the service layer, it will apply for more computing resources from the local infrastructure node. If the local infrastructure node is also running out of resources, the proposed scheme will turn to other infrastructure nodes in the network for help.

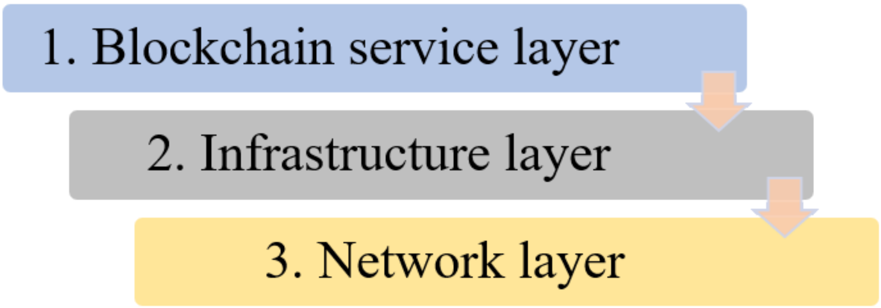


Figure 4-3 Resource scheduling priority.

4.3.2 Working procedure of the proposed scheme

Since the blockchain system is usually built on a Peer-2-Peer (p2p) communication pattern and uses epidemic protocols to propagate blocks, it needs to be isolated from the other services running on the same infrastructure node to prevent interference. During the initialization phase, the system will allocate an isolated portion of resources for the blockchain and its related services.

The working procedure of the proposed scheme is shown in Figure 4-4. After the system initialization, the proposed scheme will keep monitoring the operating status of the blockchain-enabled IoV system. When the current metrics of the system are performing lower than expected, which is a set of metrics measured from the system under sufficient resources, it will check the blockchain service layer that is isolated resources allocated for blockchain and its related services.

If the resources of the service layer do not run out, the proposed scheme will invoke the proposed resource control algorithm to increase the resource utilization of the Fabric peer container running on the service layer. If the resources of the service layer run out, the proposed scheme will check resource utilization of the infrastructure layer for reinforcement.

If the resources of the local infrastructure node do not run out, the proposed scheme will invoke the proposed scaling control algorithm to apply for more resources for the blockchain services. On the contrary, if the infrastructure node is busy, the proposed scheme will check for idle nodes in the network.

If there are idle nodes available in the network, the proposed scheme will invoke the proposed scaling control algorithm to activate the backup peer on the available node. Otherwise, the system will report overload.

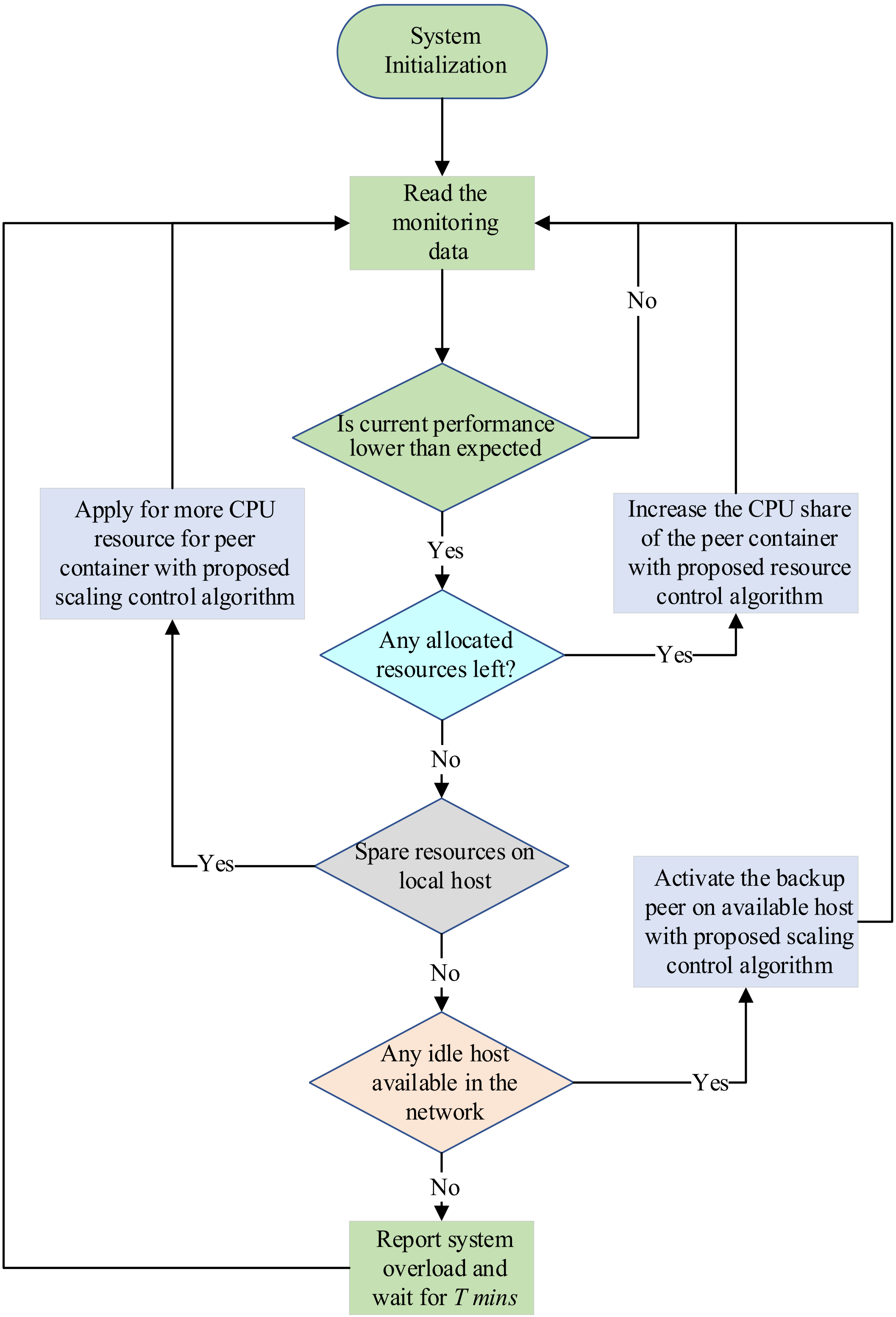


Figure 4-4 Flowchart of the proposed scheme.

4.4 Proposed monitoring system and control algorithms

The design of the proposed monitoring system and control algorithms is elaborated in detail in this section. The proposed scheme adjusts the system resource scheduling on 3 different layers according to the metrics collected by the monitoring system.

4.4.1 Proposed monitoring system

A monitoring system is designed to obtain the performance metrics like throughput, latency, transaction success ratio, and CPU utilization of the system by continuously monitoring and analyzing the running statistics of the peer containers. The monitor can either be running on an infrastructure node or cloud server. It is connected to all the peer containers through the Docker 2375 port on the target node to monitor all the statistics of the blockchain peer container. In this monitoring system, the throughput is equal to the number of transactions recorded on the blockchain ledger divided by the elapsed time.

$$Throughput = \frac{Recorded\ \ transactions}{Elapsed\ \ time} \tag{4.}$$

The latency is measured through the difference between the time for a transaction being proposed and the time that the transaction is committed.

$$Latency = \ Committed\ \ time\ \ -\ Proposed\ \ time \tag{5.}$$

The transaction success ratio is used to measure the transactions marked as valid by the committed peers and successfully applied on the user account value. This is necessary because invalid transactions are also included in throughput calculation and we need to know how many of the transactions are valid. It is calculated as the number of successful transactions divided by the total number of transactions.

$$Success\ \ ratio = \frac{Successful\ \ transactions}{All\ \ transactions} \tag{6.}$$

All these parameters can be easily found in the docker log files of the peer containers through the Docker 2375 port of the targeted node.

4.4.2 The proposed resource control algorithm

Since most blockchain systems are computing-intensive, the performance of the system highly depends on the allocated resources. However, it is impractical to equip all the nodes with sufficient resources in the blockchain system. There will be a huge cost and waste of resources during the idle period.

According to [86], the CPU resource of the Fabric peer container is the bottleneck of the system performance under limited resources. The monitoring data we got also evidence this conclusion. As the transaction sending rate increases, the peer container always runs out of resources first. So, we can improve the resource utilization efficiency of the peer nodes to get better performance.

The proposed resource control algorithm is shown in Algorithm 2. It can adjust the CPU utilization percentage by changing the CPU share of each component running in the blockchain service layer. The CPU share is the CPU utilization portion of a project allocated by the CPU scheduler which is 1024 by default. Increasing the CPU share of a container means increasing the priority and proportion of the resource utilization. The controller adjusts the resources allocated to the peers according to the metrics from the monitoring system and the current traffic conditions.

As shown in Table 4-1, a baseline parameter set that records the best performance of the system with sufficient resources under different transaction arriving rates is tested in advance as the reference CPU utilization. Note that the resource isolation of the system is in units of a single CPU, so the CPU utilization in the table is also made in units of the percentage of a single CPU core to make it easier to allocate resources, and the tested host has 8 CPU cores, so it can reach a maximum value of 800% theoretically.

Table 4-1 Reference CPU utilization table

Number of vehicles	Sending rate (TXN/s)	CPU utilization (%)
1000	100	75.03
1200	120	80.83
1400	140	86.25667

1600	160	92.65333
1800	180	102.3567
2000	200	113.2533
2200	220	118.7767
2400	240	122.5067
2600	260	137.9733
2800	280	141.3467
3000	300	152.84
3200	320	157.92
3400	340	163.8833
3600	360	171.27
3800	380	178.89

When the monitor detects the system is running under the reference performance, the proposed scheme will check if there are allocated resources left in the blockchain service layer. If there are spare computing resources, the resource controller will increase the CPU share of the peer container according to the equation shown below.

$$\begin{aligned} S_N &= 1024 + 1024 * \left(\frac{1 - U_P - U_C}{U_P}\right) \\ &= 1024 * \left(\frac{1 - U_C}{U_P}\right) \end{aligned} \qquad (7.)$$

$$\begin{aligned} S_N &= 1024 + 1024 * \left(\frac{1 - U_P}{U_P}\right) \\ &= \frac{1024}{U_P} \end{aligned} \qquad (8.)$$

Where S_N represents the new CPU share of the peer container, U_P represents the current CPU utilization of the Fabric peer container, and U_C represents the coexisting task CPU utilization. The coexisting task here refers to blockchain-related tasks running on the service layer with the blockchain system services, such as services for the vehicular blockchain applications.

The controller will check for coexisting tasks first. If there are coexisting tasks running in the service layer, it will adjust the CPU share of the peer container according to equation (7).

Otherwise, the controller will adjust the CPU share of the peer container according to equation (8). 1024 is the default value of the CPU share for all projects. The numerator on the right side of equation (7) and equation (8) represents the resources left in the service layer. The fraction on the right side of the equation represents the ratio of unused CPU resources to the CPU resources being used by the Fabric peer. The value of the new CPU share of the peer container S_N is inversely proportional to the current Fabric peer CPU utilization U_P. The more resources are left, the higher S_N will be. When the U_P is big enough, 100% for example, the S_N will remain near 1024 because if the allocated resources are all used by Fabric peer container, it is meaningless to increase its priority.

Algorithm 2: Resource control algorithm

```
Input: Current number of vehicles $N_V$, peer CPU utilization $U_P$, and coexisting
task CPU utilization $U_C$
Output: New CPU share of the peer container $S_N$
Import reference CPU utilization table
Get reference CPU utilization $U_R$ from Reference CPU utilization table
according to $N_V$
for each $U_P < U_R$ do
    if coexisting task = true, then
        $S_N = 1024 * \left(\frac{1 - U_C}{U_P}\right)$
    else
        $S_N = \frac{1024}{U_P}$
Return $S_N$
```

4.4.3 The proposed scaling control algorithm

The resource control algorithm only works on the blockchain service layer which is the resources allocated for the blockchain and its related services. When the service layer runs out of resources, the proposed scheme will apply for more resources from the local infrastructure node or activate backup peers from other available nodes in the network through the proposed scaling control algorithm as shown in Algorithm 3.

The CPU utilization of the system is used to determine whether the system is under high pressure. When the transaction sending rate is higher than 2000 where the CPU utilization of the system is 92.2% under limited resources, the performance of the system starts to drop fast.

So the threshold of the high pressure is set to 90% to prevent the system performance from a dramatic drop. The threshold can be adjusted flexibly according to the configuration of the system and the priority of the blockchain services. The lower the threshold, the earlier the blockchain system takes up more resources.

Algorithm 3: Scaling control algorithm

```
Input: Peer CPU utilization $U_P$, coexisting task CPU utilization $U_C$, and total
CPU resources allocated to the blockchain service layer $U_T$
Output: Selected scaling method
for situation under high pressure $U_P + U_C > 90\% U_T$ do
    if there are spare resources on the localhost then
        apply more resources for the blockchain service layer
    else if there are idle hosts available in the network
        activate the backup peer on the available host, for all new
        activated peer $U_P$ do
            Anchor node = min ($U_{P1}, U_{P2}, U_{P3}$)
            Update anchor node
        end for
    else
        report system overload
end for
Return
```

When the resource utilization of the service layer is running out, the proposed scheme will invoke the proposed scaling control algorithm to acquire reinforcement for the blockchain services as shown in Figure 4-5. If there are spare resources at the local infrastructure node, the algorithm will allocate more CPU resources for the blockchain-related containers with the docker command "*Docker run --cpus=<value>*". This command will adjust the resources allocated to the containers running on the local infrastructure node.

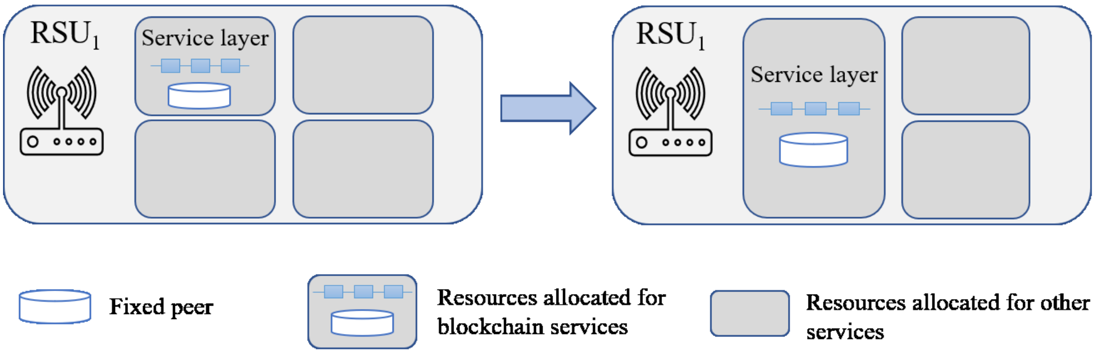


Figure 4-5 Scaling up locally with the proposed scaling control algorithm.

During system initialization, every infrastructure node will set up a backup peer configuration without activating the peer. If the local host is fully occupied, the proposal will check for idle hosts in the network layer and activate the backup peer on the available host as shown in Figure 4-6. After the backup peer is activated, the scaling control algorithm will compare the resources allocated to the peers and update the one with the most available resources as anchor node with Fabric command "*peer channel update*". The anchor node can be regarded as the communication junction in the network and is responsible for the dissemination of the blocks and ledger synchronization of the peers.

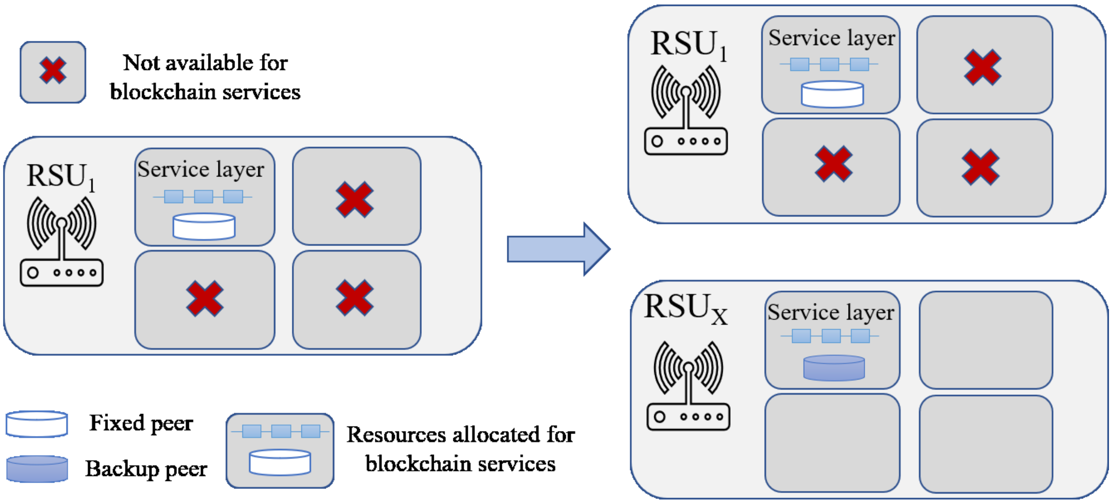


Figure 4-6 Scaling up with backup peers in the network with the proposed scaling control algorithm.

4.5 The generality of the proposed scheme

From the application scenario point of view, the proposed scheme is designed to improve the performance of the blockchain systems under dynamic workload with peak and idle periods such as IoV systems by efficiently allocating the computing resources of the devices or infrastructures, but it is also suitable for managing the resource allocation under different situations. Similar to the proposed scheme in Chapter 3, the proposed resource control algorithm is suitable for all kinds of different blockchain platforms since it only adjusts the resource allocation and does not involve any modification on consensus mechanism or data structure. However, the proposed scaling control algorithm is only suitable for consortium blockchain since it involves the scalability of the system. Unlike the public blockchain in which all the nodes are equal and verify the transaction on their own, the consortium blockchain relies on the powerful capabilities of the peer nodes to verify the validity of the transactions.

4.6 Simulation design

The simulation design of the system including environmental settings and simulation assumptions are presented in this section

4.6.1 Simulation design

The simulations are conducted on 3 computer hosts representing 3 different components of the blockchain-based IoV network. The first computer is a desktop consisting of an Intel i3-8100H CPU, 8G ram, and a GeForce GTX 1050Ti graphic card. It is configured as the ordering server (host1). 3 ordering nodes are configured on the ordering server. We use Fabric 2.2 as our simulation platform and LevelDB as our fabric database. There is one point that needs to be stated is that the resources of the ordering peer have not been exhausted in all the experiments.

The second and third computers are laptops with Intel i5-10300H CPU, 16G ram, and a GeForce GTX 1650 Ti graphic card. They are configured as the endorsement and committing peers (host2 and host3). Each laptop has 2 peers configured on it which are from the same organization and also deploy Fabric 2.2 and LevelDB in the simulation settings.

The block size is 100 transactions per block in all simulations. The batch timeout (blockchain formation interval) is set to 2 seconds. It means that no matter how many transactions are in the current packing block, the ordering peer will cut the line when 2 seconds elapsed from the first transaction in this block.

To generate proper workloads, Hyperledger Caliper is employed to produce the transaction traffic of the system network. It can be used to generate various kinds of blockchain transactions at different sending rates to the blockchain system under test. It is compatible with Ethereum and most of the Hyperledger blockchain solutions.

Similar to the simulations in Chapter 3, the variation of vehicle density is simulated by changing the transaction sending rate of the Caliper workload. It is assumed that all vehicles generate the same number of transactions within a certain period of time. Thus, the number of vehicles is proportional to the transaction sending rates of the Caliper workload. For example, a vehicle in VANET sends a safety beacon message every 10 seconds, so when there are 1,000 vehicles, the sending rate will be 100 transactions per second.

4.6.2 Simulation set up

Figure 4-7 shows the system setup of the simulations. Host1 represents the ordering peers which run on the desktop and provides ordering service for the system. Host2 and host3 are running on the laptops and represent endorsement and commitment nodes from different organizations. All the peers join the same channel to interact with each other. Sufficient resources scenario utilizes all the 8 cores of the host while limited resources scenario isolated the blockchain services on a single core to simulate a limited resource situation.

The simulation topology is similar to the topology in Chapter 3, an area of 1,000 * 1,000m with 3 horizontal roads and 3 vertical roads, forming 4 square blocks. Every block has an infrastructure node in it and all the vehicles on these roads can at least communicate with one of the infrastructure nodes directly.

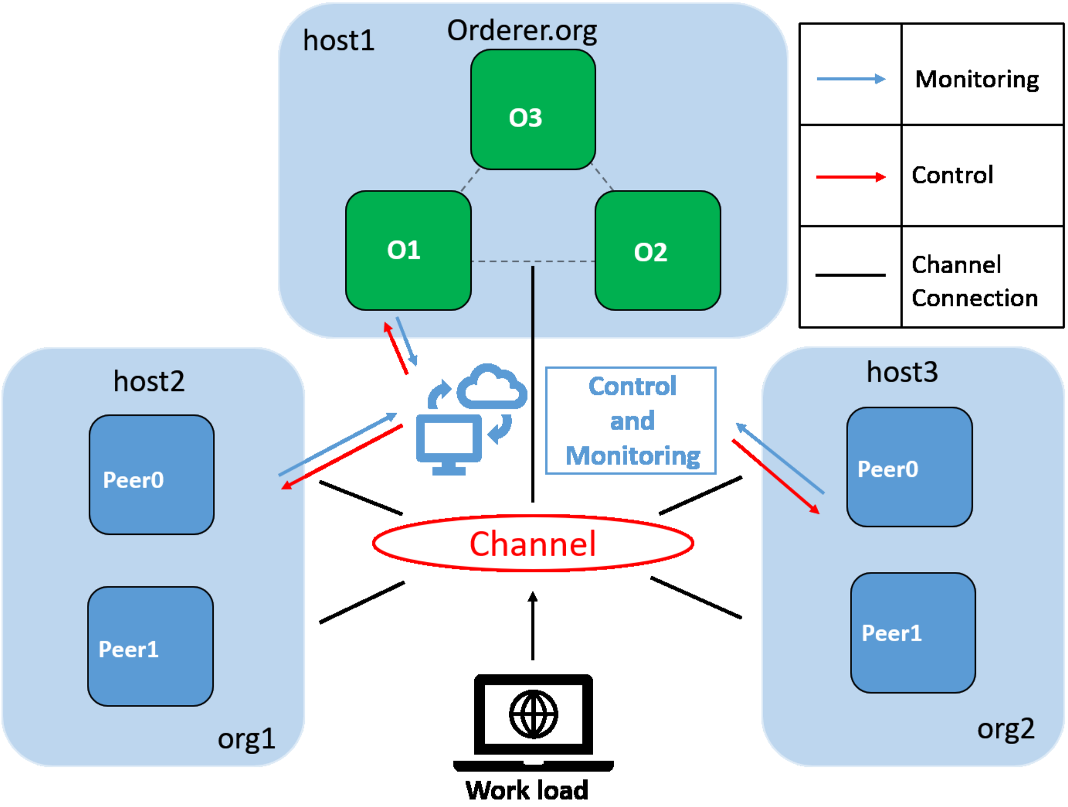


Figure 4-7 Simulation setup.

4.7 Simulation results

The performances of the system for different vehicle densities under sufficient resources (8 CPU cores), limited resources (1 CPU core) with even allocation, and limited resources (1 CPU core) with the proposal are tested in the simulations. The performance metrics of the system under limited resources with even allocation and sufficient resources are measured first as baselines in our simulation. Then we compare the system performance under limited resources with the proposal with the baselines to evaluate the impact of the proposed method.

Two different load scenarios are tested which are the situation with coexisting tasks and the situation without coexisting tasks to evaluate the performance of the proposed scheme. The coexisting tasks in simulations are simulated with the stress-ng, a well-known CPU stress test tool. The coexisting tasks here can be the vehicular blockchain app or other tasks coexisting with the blockchain services on the infrastructure nodes.

Similar to the simulation results in Chapter 3, the evaluation in this chapter also takes transaction throughput, latency, and success ratio as the evaluation metrics, and since the proposed scheme in this chapter involves resource management, resource consumption and scalability are also considered in the evaluation of the proposed scheme. The performance-cost ratio is shown in the third scenario to show the relationship between the performance and the construction cost of the system. The results of the proposed scheme under different scaling strategies are shown in the results to show the performance of the system when it is scaling up locally or remotely.

The simulations for evaluating the proposed scaling control algorithm are also tested with coexisting tasks since the algorithm is designed to improve the system performance under high pressure. And the last scenario is designed for the evaluation of the system performance with 2 backup peers activated at the same time.

4.7.1 Scenario without coexisting tasks

The latency performance of the system is shown in Figure 4-8. As we can see, the latency of the system under sufficient resources does not change much with the increasing number of vehicles, While the latency metrics of the system under limited resources increase dramatically after the number of vehicles surpasses 2,500. Since the ordering service has sufficient resources in all experiments, this indicates that the transactions can not be processed in time and begin queuing for verification at the peer nodes under the given computing resources. This is because, when the ordering peer broadcasts a newly packed block to a peer, it will trigger the validation process in a peer which invokes the validation system chaincode (VSCC) [124]. During this stage, all the transactions need to wait for validation. Therefore, if the peer does not have enough resources, the waiting time of the transactions will become longer which will result in higher latency. So, we raise the CPU share of the peer container with the proposed control method to increase the CPU utilization of the peer container. As we can see from Figure 4-8 that the latency of the proposal increases visibly slower than the one with even allocation. This shows that our method can effectively reduce the delay caused by the increase in the number of vehicles with limited resources.

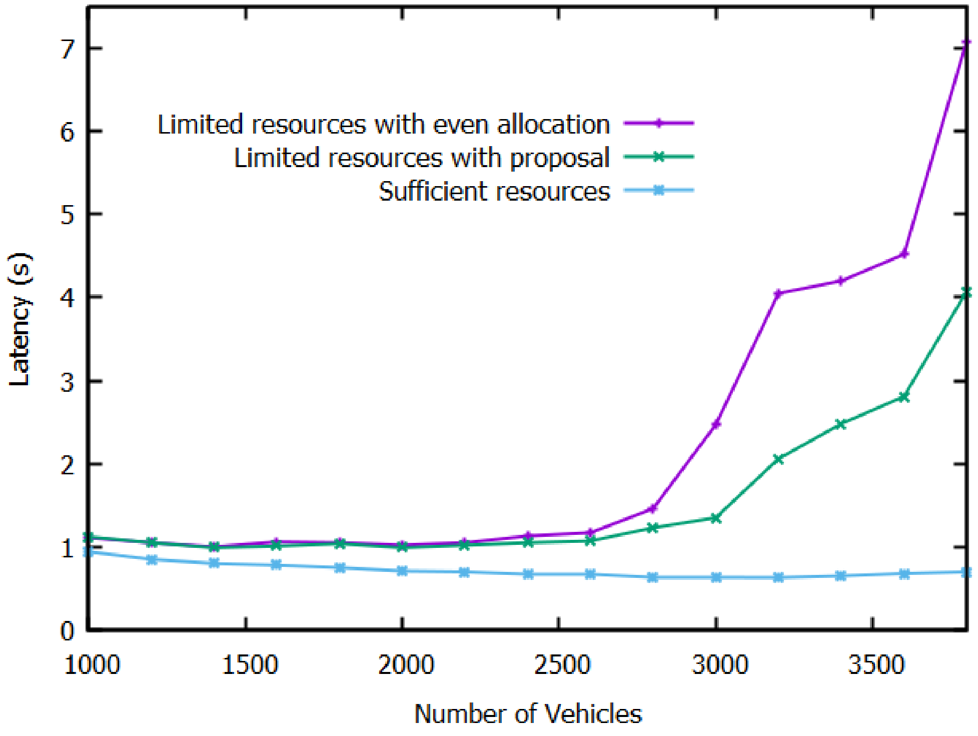


Figure 4-8 Latency performance of the system without coexisting tasks.

The performance of the system success ratio is shown in Figure 4-9. In contrast to the latency performance, we can see that the success ratio metrics of the limited resources drop significantly when the number of vehicles exceeds 2,500. This is because of the multi-version concurrency control (MVCC) which is employed by many concurrency systems to control the consistency of the key value. Since more transactions are waiting for validation at the peers, the value of the key that a transaction is trying to update may be different from the value of this key when this transaction is proposed which causes MVCC read conflict. As the waiting line of the transaction increases, the possibility of the MVCC read conflict will also increase, resulting in more transactions marked as invalid. As we can see from Figure 4-9 that the success ratio of the proposal is always higher than the one with even allocation which means our method can effectively lessen the probability of the transactions being marked as invalid along with the increasing number of vehicles under limited resources.

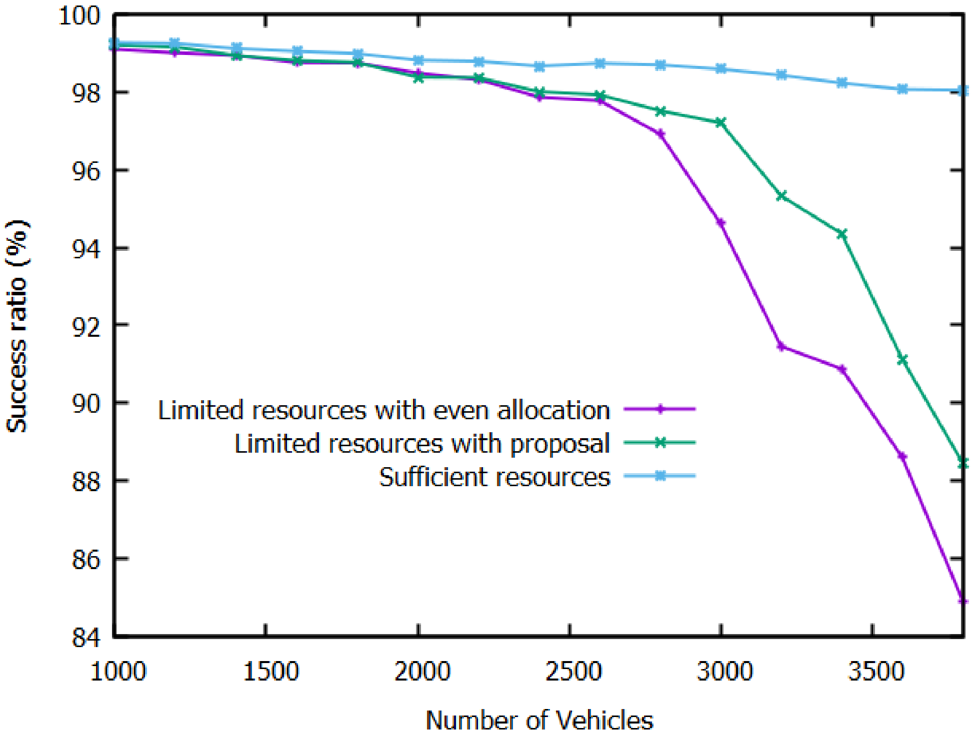


Figure 4-9 Success ratio performance of the system without coexisting tasks.

The performance of the system throughput under the different number of vehicles is shown in Figure 4-10. As we can see that the throughput performance does not change obviously as it is shown in the latency and success ratio result. This is because the throughput performance depends on the cooperation of all the components in the system including ordering services and total resources allocated for the blockchain services. Increasing the CPU utilization priority of the peer node will also influence the other components of the system, and the total resource of the system does not change. However, the proposal still keeps a higher throughput than the one with even allocation.

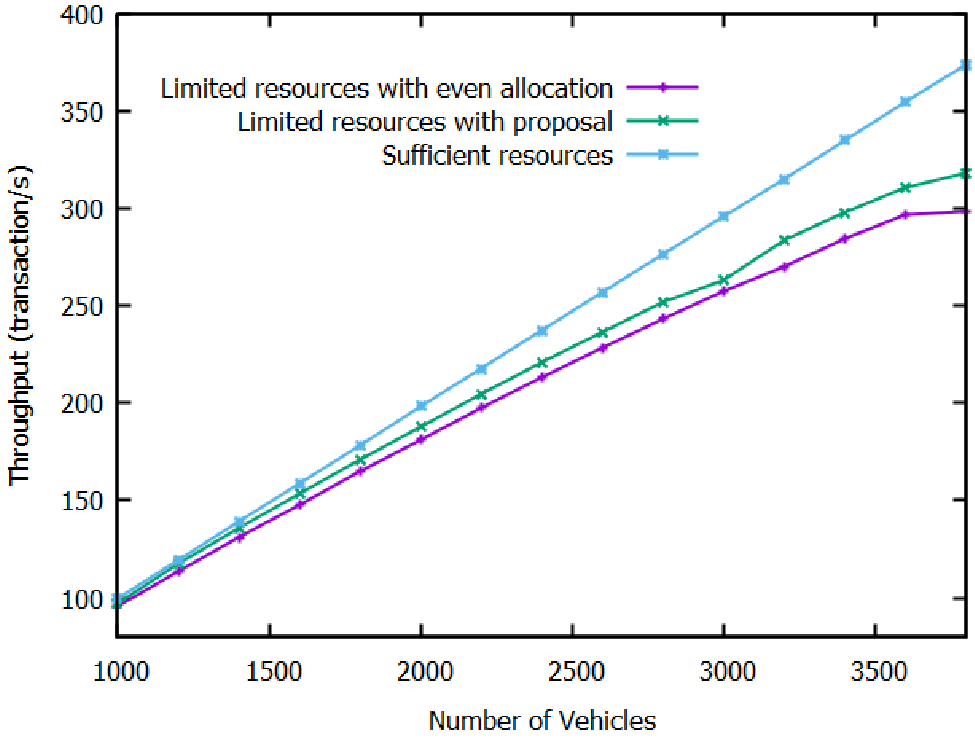


Figure 4-10 Throughput performance of the system without coexisting tasks.

4.7.2 Scenario with coexisting tasks

The coexisting task in this scenario is conducted with the stress-ng tool to produce pressure to the blockchain service layer. It is configured to produce a container in the service layer that consumes 50% of the CPU resources allocated to the isolated blockchain service.

The latency performance of the system with coexisting tasks is shown in Figure 4-11. We can see from the figure that the latency of the system under sufficient resources is not influenced badly by the increasing number of vehicles. On the contrary, the latency gets smaller as the number of vehicles increases. This is because the transaction sending rate also increases with the increasing number of vehicles. With sufficient resources, the transactions do not need to wait in line for validation. The faster the transactions are arriving at the Fabric peer node, the faster they will be sent to the ordering services, thereby the faster a block will be formed. Thus, the latency of the system will continuously decrease with the increasing transaction rate until there are not enough resources.

Unlike the latency performance without coexisting tasks, the latency metrics of the system under limited resources with coexisting tasks start increasing from the beginning of the simulation with the increasing number of vehicles. This is because with the coexisting task running together, the system needs to arrange the CPU schedule for the coexisting task anyhow which will result in the waiting of the blockchain services.

Since the system resources are not highly occupied in the early stage of the simulation, the difference between the performance with the proposal and with even allocation is not quite obvious. But when the number of vehicles surpasses 2,200, increasing the priority of the Fabric peer container with the proposed algorithm became more effective. But along with the increase of the vehicle number, it returns to a small difference between the two lines. This shows that the proposed resource control algorithm is not very reliable when the system resources are running out. It needs some other methods to further improve the performance of the blockchain-enabled IoV system.

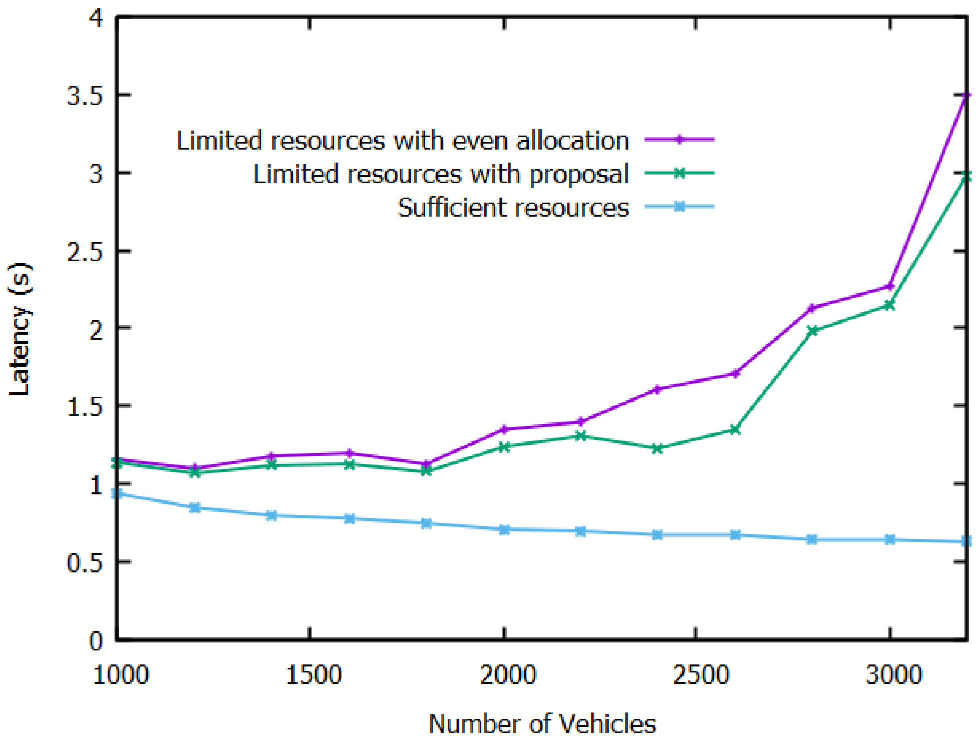


Figure 4-11 Latency performance of the system with coexisting tasks.

The performance of the system success ratio with coexisting tasks is shown in Figure 4-12. In contrast to the latency performance, we can see that the success ratio metrics of the system under sufficient resources are also influenced by the increasing number of vehicles. This is because although there are enough resources in the system, along with the increasing transaction rate, the probability of two transactions that are proposed to update the same value at the same time increases. So, when any one of them is verified and marked as valid first, the other one will be marked as invalid and need to be proposed again with the new version of the target key value. It will result in the failure of the proposed transaction thereby lowering the success ratio.

As we can see from Figure 4-12, the success ratios of the system under limited resources drop at nearly the same speed before the number of vehicles comes to 2,200. Similar to the latency performance, the proposal performs prominently from 2,200 and returns to the performance similar to that of the performance with even allocation at around 2,800.

Returning to the same point means that the system resources are running out, and not many spare resources left. However, we can see from the figure that the performance metrics of the proposal are better than those of the situation with even allocation. This is because when the system runs out of resources, the one with a higher CPU share will be preemptive and take over the resources from the others.

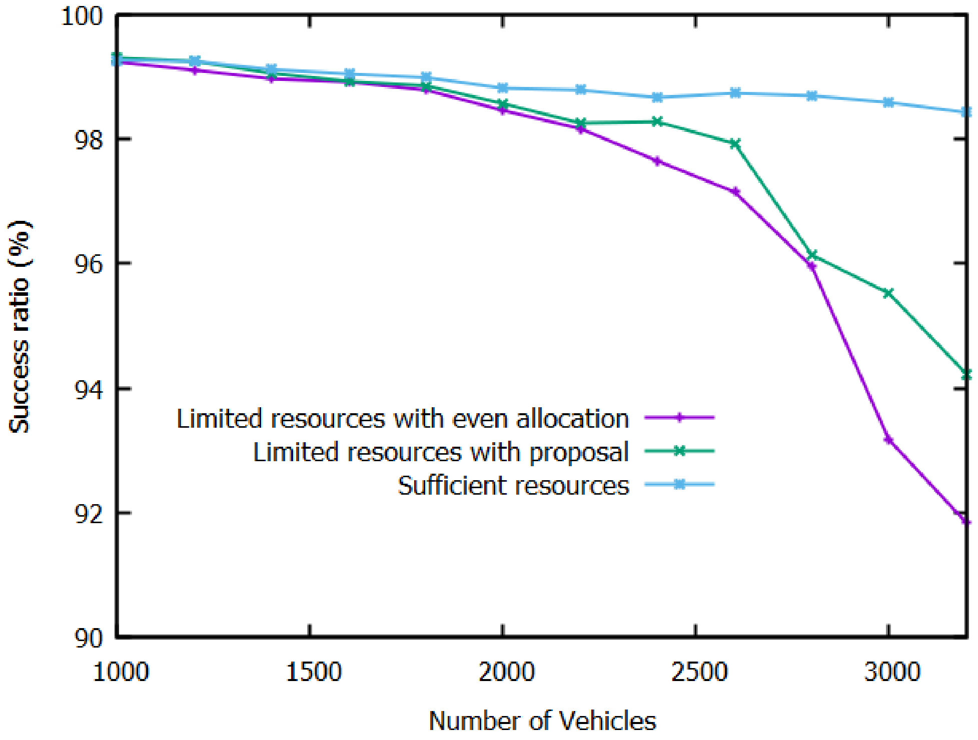


Figure 4-12 Success ratio performance of the system with coexisting tasks.

The performance of the system throughput with coexisting tasks is shown in Figure 4-13. As we can see that the throughput performance of the system under sufficient resources is nearly a straight line. The throughput is approximately equal to the transaction arriving rate which means the throughput is hardly influenced by the coexisting task.

The other two are both influenced by the coexisting task and the limited resources, but not changed as obvious as it is in the latency and success ratio result. This is because the proposed resource control algorithm does not change the total amount of the resources allocated to the blockchain services to which the throughput performance is highly related. Moreover, increasing the CPU utilization priority of the Fabric peer will also influence the other components of the system. Despite that, the throughput performance of the proposal is slightly higher than the one with even allocation.

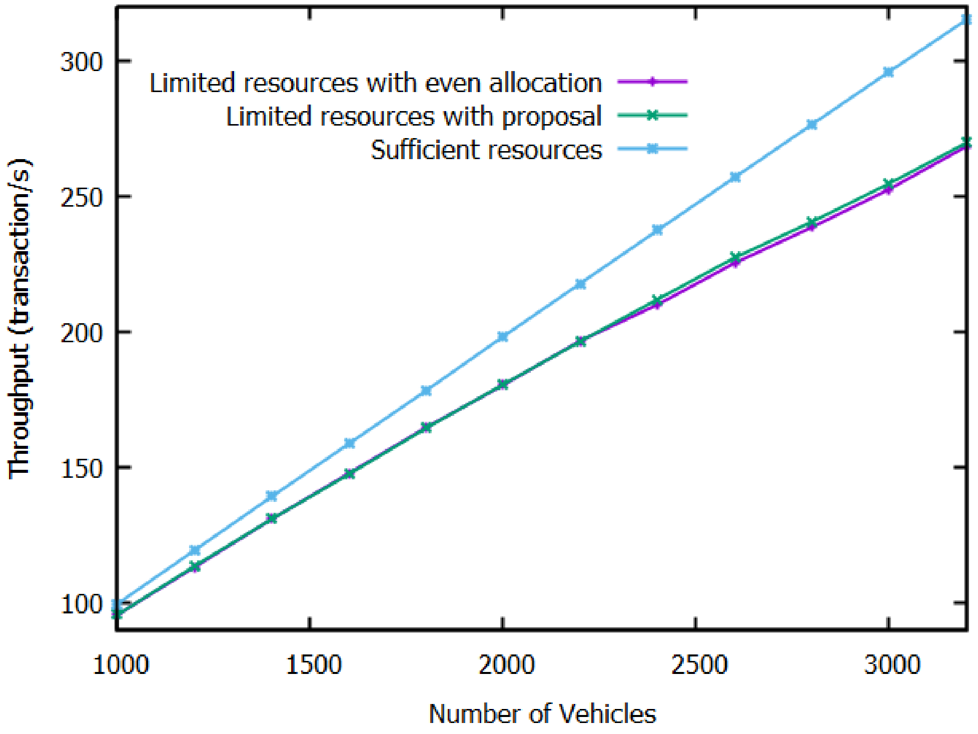


Figure 4-13 Throughput performance of the system with coexisting tasks.

4.7.3 Scenario with the backup peer activated

This scenario has tested the system performance under 4 different situations:

Full configuration with even allocation. As a baseline, all the resources are allocated to the system directly from the beginning with even allocation. All the tasks and processes will share the allocated resources evenly without any priority or hierarchical control.

One backup peer activated with even allocation. The system is first running on the limited resources with even allocation as a baseline. When the resources run out, the system will activate 1 backup peer with even allocation.

One backup peer activated with the proposal. The system is first running under control of the proposed resource control algorithm, and when the system runs out of resources, it activates 1 backup peer with the proposed scaling control algorithm.

Scaling locally with the proposal. The system is first running under the control of the proposed resource control algorithm, and when the system runs out of resources, it scales up locally with the proposed scaling control algorithm.

The first two simulations above are tested as the baselines to compare with the performance of the system controlled by the proposed resource control algorithm and scaling control algorithm to fully evaluate the impact of the proposal. The last three situations have the same allocated resources through the whole simulations: limited resources at the beginning (1 CPU core) and doubled resources (2 CPU cores) after the backup peer activated or scaled up locally. The first one with full configuration has doubled resources (2 CPU cores) from the beginning.

The latency performance of the system is shown in Figure 4-14. As we can see that the latency of the system with full configuration keeps staying at a low level and even reduces a little because of the reason similar to that explained earlier in section 4.7.2. The latency of the system with 1 backup peer activated with even allocation increases dramatically along with the increasing number of the vehicles although it gets the same resources as those with the proposal. This is because the blockchain running on the system is a typical distributed system, and in a distributed system, one more peer does not simply mean more resources for the system. It also means that the system has one more agent to communicate and discuss with which will generate extra overhead for the system. Adding one more peer without any control methods or adjustments to the system will further lower the performance of the system.

As we can see from the figure that there are two different stages in the increase of the latency performance of the system with 1 backup peer activated with even allocation. The first stage is a slow-growth stage which is from the beginning to the number of the vehicles comes to 2,200 where the original fixed peers run out of resources. After that, the latency of the system grows significantly fast because of the VSCC process mentioned above.

The latency of the system with 1 backup peer with the proposal and scale up locally both grow a little with the increasing number of vehicles before 1,800 and then the proposed monitoring system detects that the system resources are running out, so 1 backup peer is activated with the proposal in the third simulation and scales up locally with the proposal in the fourth simulation.

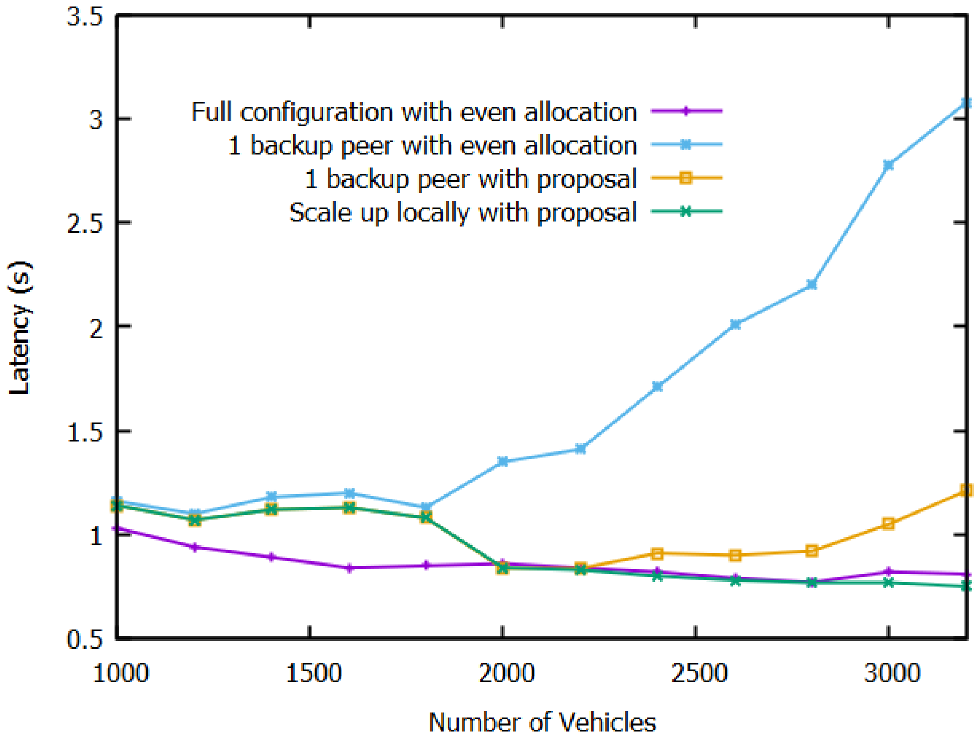


Figure 4-14 Latency performance of the system with a backup peer.

We can see that the latency of the system with 1 backup peer with proposal begins to grow after the number of the vehicles surpasses 2,200, but much slower than the one in the former one with even allocation which means the backup peer has shared the responsibility for the original peers and the workload is better balanced with the proposal. The proposal configures the peer with the most available resources as the anchor node. This measure helps to balance the system load and thereby improves the resource utilization of the system.

As we can see from Figure 4-14 that the latency performance of the system scaling up locally with the proposal is even lower than the one with full configuration when the number of vehicles is high. This shows that our methods can help the system scale up effectively with low latency. However, the one scaling locally is better than the one with an activated backup peer because although they have the same resources, 1 more peer will still bring unnecessary overhead to the system. So only when the local host is running out of resources should the system activate the backup peer for reinforcement.

To further illustrate the superiority of the proposal, the performance-cost ratio of the system latency is shown in Figure 4-15. Typically, the performance-cost ratio is the performance of the system divided by the cost, but since the performance to be compared here is latency, we divide the inverse of the latency by the cost of resources.

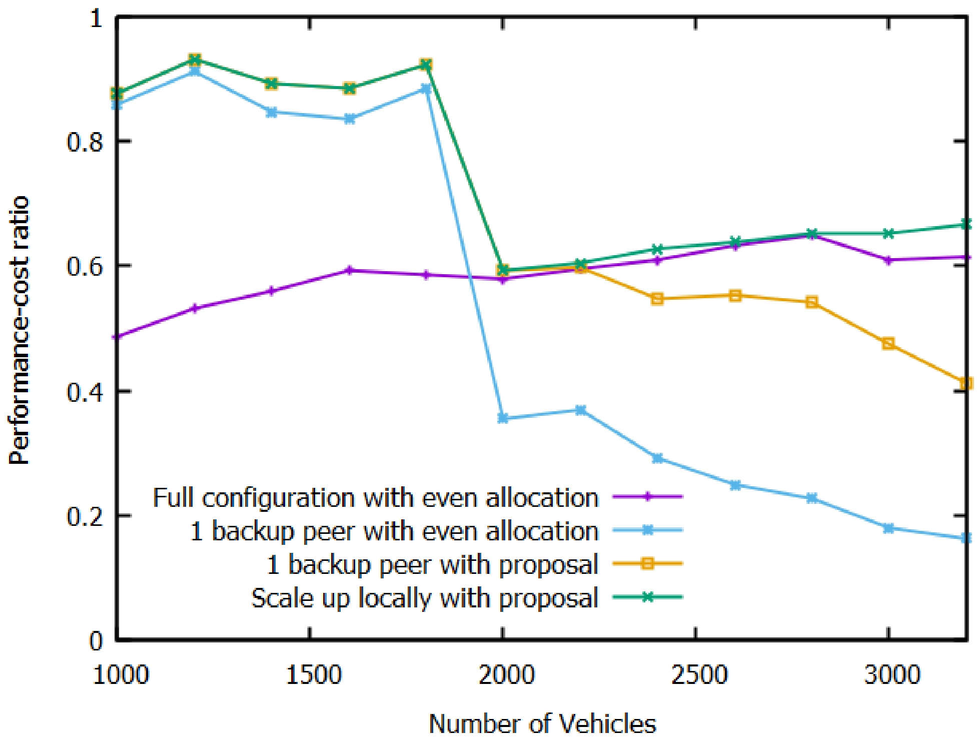


Figure 4-15 System latency performance-cost ratio.

The cost here is expressed in terms of the number of used CPU cores for convenience. Thereby, the cost of the full configuration is 2 since 2 CPU cores are allocated to the system from the beginning of the simulation. Similarly, the cost of the others is 1 at the beginning and 2 after the system scales up. In a practical scenario, such as an actual blockchain-enabled ITS system, the cost should be calculated as the total computing resources allocated to the blockchain system.

Even though the system under full configuration has better performance at the beginning of the simulation result, it can be found from Figure 4-15 that its performance-cost ratio is much lower than those with the proposal. This means it will need more construction and operating cost to achieve similar performance with the system with the proposal.

The performance of the system success ratio is shown in Figure 4-16. The success ratio of the system in all simulations is similar to each other until the number of vehicles surpasses 1,800. The success ratio of the system with full configuration and the one that scales up locally perform stable and only drop a little through the whole test. The success ratio of the system under 1 backup peer activated with even allocation begins to drop first when its allocated resources run out and activates 1 backup peer after the number of vehicles exceeds 1,800.

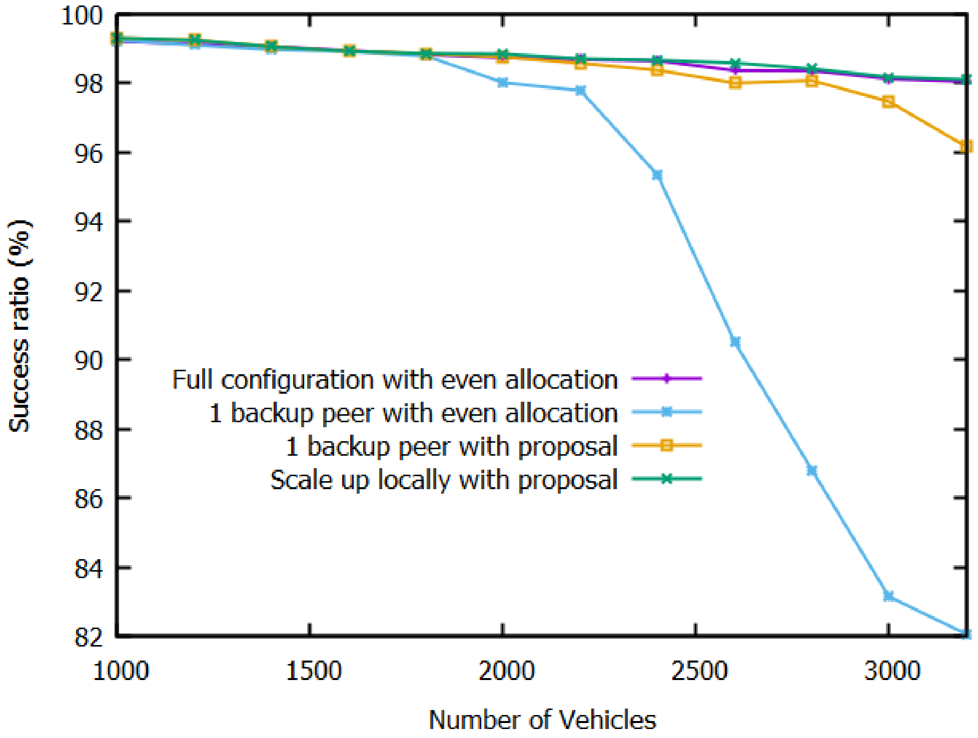


Figure 4-16 Success ratio performance of the system with a backup peer.

The performance of the system under 1 backup peer activated with proposal starts to drop slowly along with the increasing number of vehicles. It starts to drop faster when the number of vehicles comes to 2,800. This is because the proposal can help balance the load among the peers, but this method does not increase average resources for each peer. Thus, the resources of the peers under this situation will run out earlier than that in the scaling locally scenario.

To further illustrate the superiority of the proposal, the performance-cost ratio of the system is shown in Figure 4-17. The performance-cost ratio is equal to the success ratio of the system divided by the cost of resources. We can see that the performance-cost ratio of the proposal is almost twice as much as the system with full configuration before the number of vehicles reaches 2,000. Although the one with 1 backup peer has a similar performance-cost ratio to those with the proposal, it drops quickly as soon as the number of vehicles exceeds 2,000.

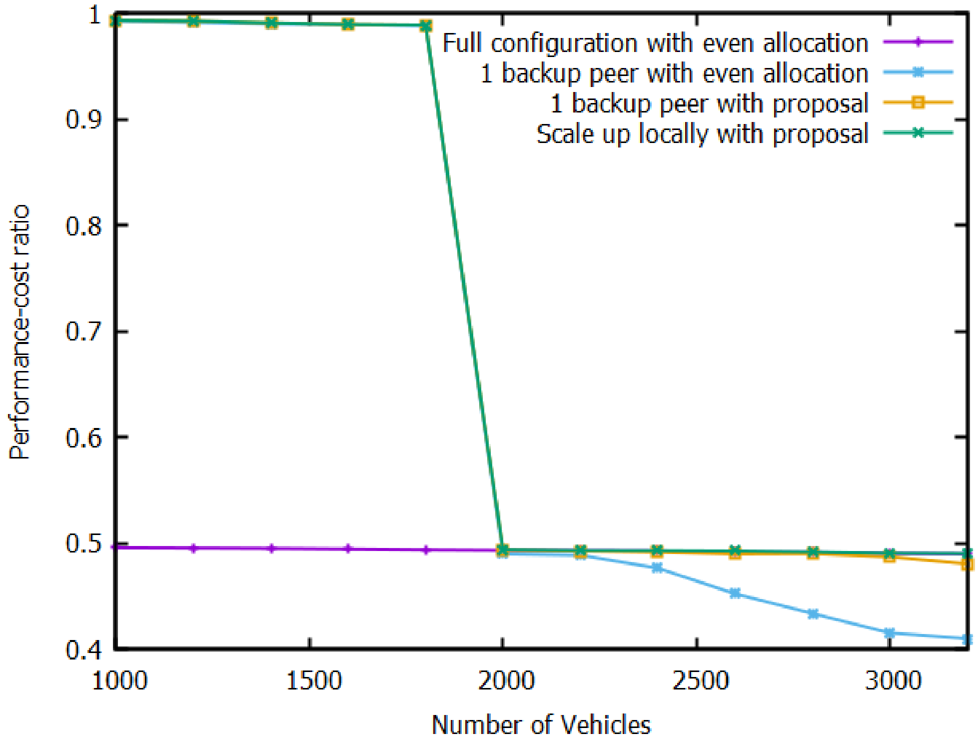


Figure 4-17 Performance-cost ratio of the system success ratio.

The performance of the system throughput is shown in Figure 4-18. The throughput performances of the four simulations are similar most of the time during the simulation. The reason is similar to that of the former simulations. At the beginning of the simulations, the system is not under pressure, so the throughput of the system is close to the transaction sending rate. We can see that around 2,000-2,200, the throughput of the system under 1 backup peer with even allocation starts to drop which means the system here is running out of resources, but the backup peer is activated from here and the throughput returns similar to others.

Since the proposed scheme focuses on the performance improvement of the peer nodes, it can significantly enhance the efficiency of the validation process, thereby reducing the verification waiting time and the probability of the transactions being marked as invalid. However, no matter if the transactions are valid or not, they will all be calculated as effective throughput and packed into the blocks as a record. Thus, the impact of the proposal is not as obvious as it is in latency or success ratio performance and it is not necessary to show the performance-cost ratio of the throughput performance. Nevertheless, as we can see that the throughputs of the scenarios with the proposal are still slightly higher than that under 1 backup peer activated with even allocation.

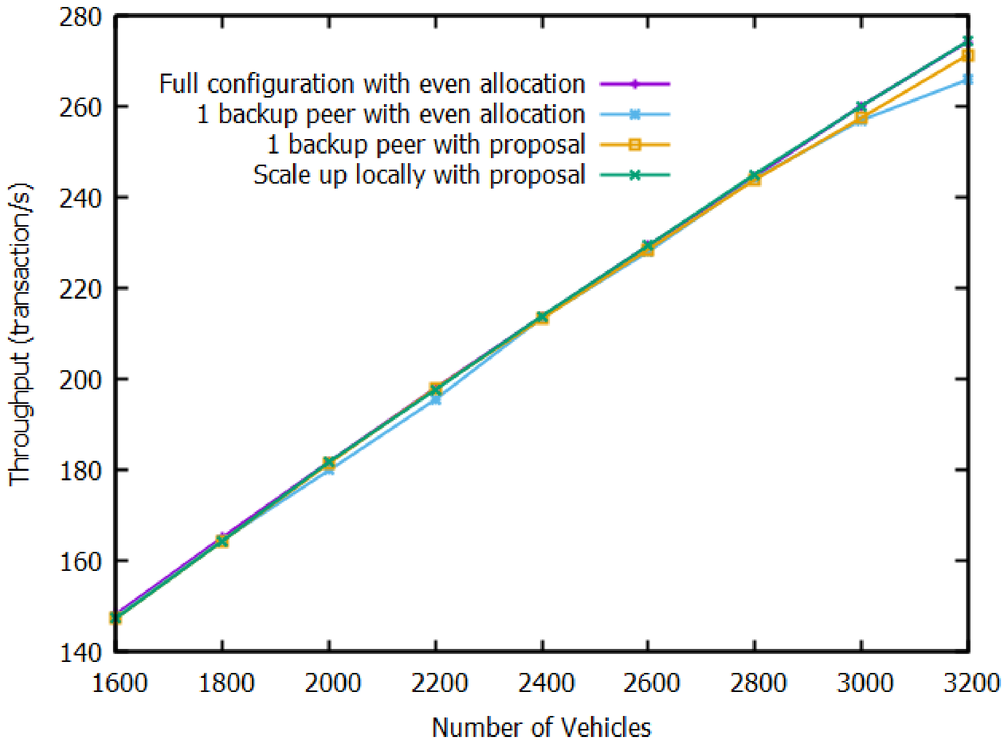


Figure 4-18 Throughput performance of the system with a backup peer.

4.7.4 Scenario with 2 backup peers activated

To evaluate the impact of activating 2 backup peers, an additional scenario is tested and shown in this section. There are two ways to activate more backup peers. The first one is preparing more backup peers on the same infrastructure node which means the backup peers on the infrastructure node share the same resources as shown in Figure 4-19. The second way is preparing more backup peers on different infrastructure nodes which means it will bring new computing resources to the system as shown in Figure 4-20.

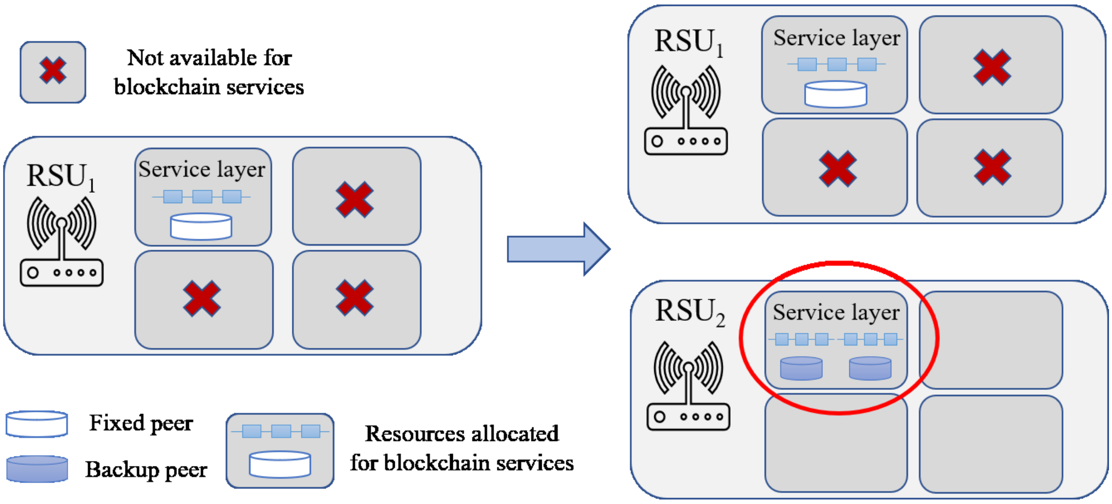


Figure 4-19 Two backup peers activated on the same infrastructure node.

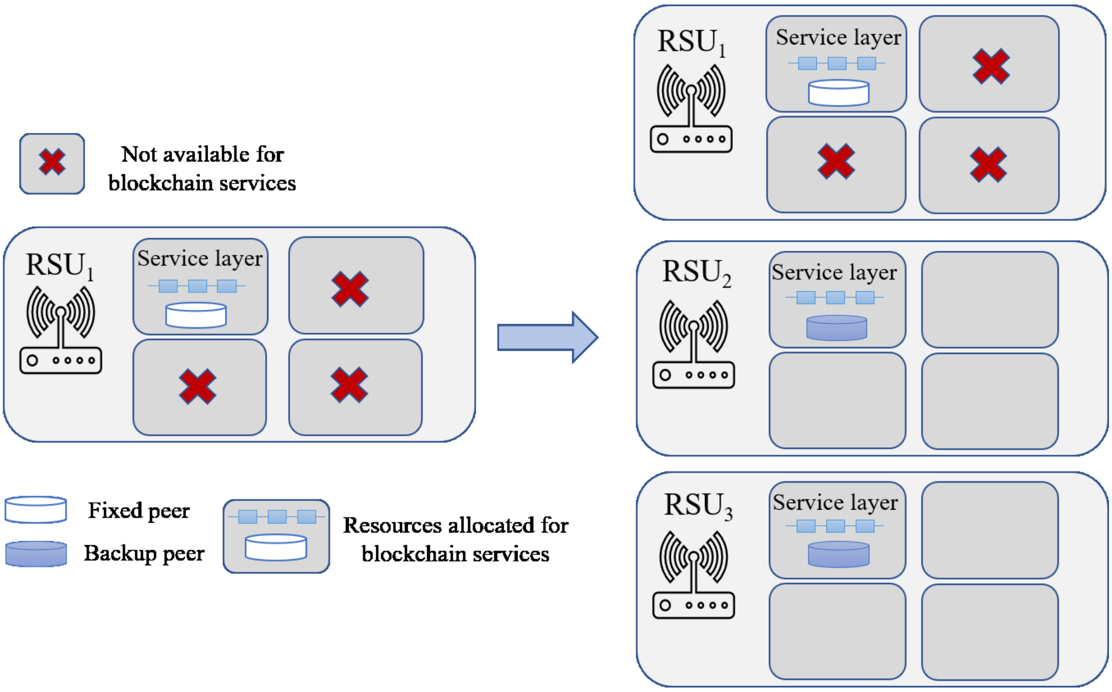


Figure 4-20 Two backup peers activated on different infrastructure nodes.

Two different situations are tested. The first one is 2 backup peers activated on the same infrastructure node, and these two backup peers are sharing the same resources with the 1 backup peer situation. The second one is 2 backup peers activated separately on different infrastructure nodes and each peer has the same resources with the 1 backup peer situation. The biggest difference between these two is whether the new backup peer is activated directly next to the original backup peer on the same allocated resources or the new backup peer has brought new resources to the system.

The latency performance of the system under different situations is shown in Figure 4-21. The results are clearer when compared in pairs, such as 2 backup peers with even allocation versus 2 backup peers with the proposal. We can see that the overall results of 2 backup peers are worse than 1 backup peer unless the new activated backup brings new resources to the system. The performance of the system with the proposal can perform better than those with even allocation.

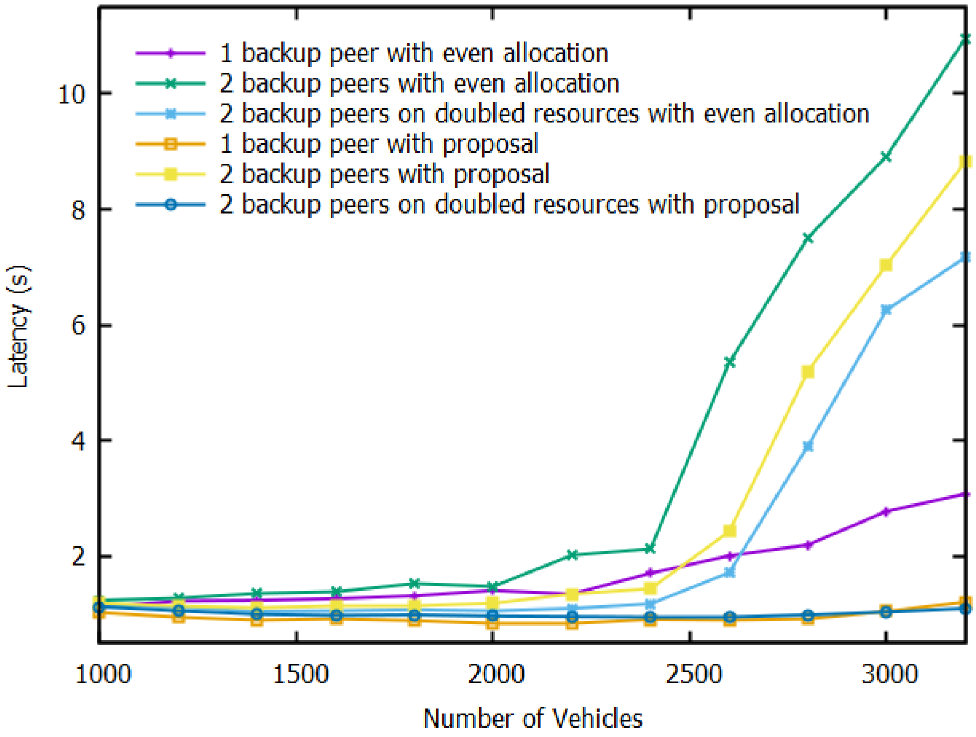


Figure 4-21 Latency performance of the system with 2 backup peers.

The success ratio performance of the system under different situations is shown in Figure 4-22. Similar to the results in latency performance, the overall results of 2 backup peers are worse than 1 backup peer under the same resources and the proposal can improve the performance of the system under all the different situations.

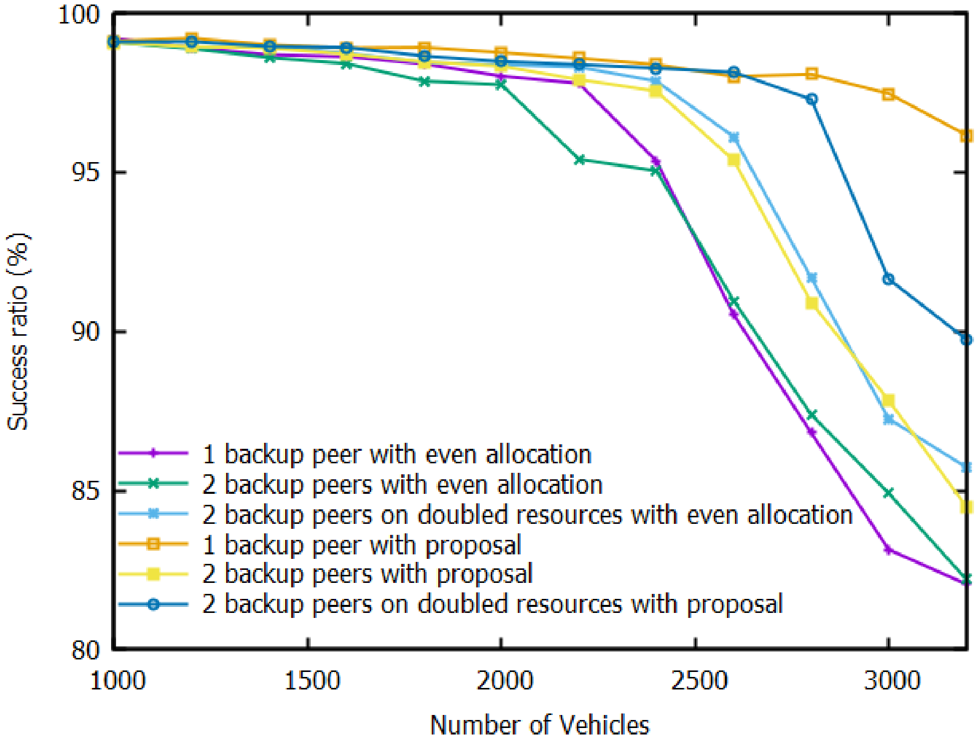


Figure 4-22 Success ratio performance of the system with 2 backup peers.

4.8 Conclusion

In this chapter, a hierarchical blockchain resource scheduling scheme for the IoV system is presented to enhance the resource utilization efficiency and improve the system performance of the blockchain-enabled IoV system under limited resources.

The implementation of the system in this Chapter is also an extension of the implementation from Chapter 3. The proposed scheme in Chapter 3 is implemented on a single desktop, so its capability for the sending rate is limited. Thus, in this Chapter, the proposed scheme is implemented on more powerful equipment that can provide larger vehicle capacity.

The problems of the conventional blockchain architecture are discussed in section 4.1. The difficulties of deploying blockchain systems in IoV systems are also analyzed. Resource utilization efficiency and flexibility should be important concerns while building a blockchain-enabled IoV system.

The design of the hierarchical blockchain resource scheduling scheme is presented in section 4.2. This section also introduces the three main components of the proposal system and their functions. The resource distribution and layering principles of the proposed hierarchical blockchain resource scheduling scheme are also elaborated. The computing resources in the network are divided into three layers which are the blockchain service layer, infrastructure layer, and network layer.

The working flow of the proposed blockchain resource scheduling scheme is presented in section 4.3. The priority of resource utilization is also discussed in this section. The proposed scheme acquires the system operating status through the proposed monitoring system and adjusts the system resource scheduling with the proposed resource control algorithm and scaling control algorithm. The system under control will utilize system resources in the order of service layer, infrastructure layer, and network layer.

In section 4.4, the design of the proposed resource monitoring system is introduced. The monitor collects the operating status of the system through the Docker 2375 port on each infrastructure node and calculates the current throughput, latency, and transaction success ratio of the system. The detailed mechanism of the proposed resource control algorithm and scaling control algorithm are elaborated in this section. The resource control algorithm increases the CPU share of the Fabric peer to improve the performance of the system according to the allocated resources and the number of vehicles. The scaling control algorithm helps the system to scale up locally or remotely according to the resource utilization of the system. The scaling control algorithm will also update the peer with the most available resources as anchor peer to further improve the system resource utilization.

In section 4.5, the generality of the proposed scheme is discussed. Finally, extensional simulations are conducted to evaluate the proposed hierarchical blockchain resource scheduling scheme. The simulation design including the environmental parameters and the setup of the simulations are introduced in section 4.6. The simulation assumptions are also presented here. The experimental results under different situations are analyzed and explained in section 4.7 to evaluate the performance of the proposal.

CHAPTER 5

5 Conclusion and future work

This chapter concludes the research on efficient blockchain for IoV with a multi-channel blockchain scheme and hierarchical resource scheduling. Section 5.1 summarizes the thesis including both the proposed multi-channel blockchain scheme and hierarchical resource scheduling scheme. Then, section 5.2 discusses possible future works.

5.1 Conclusion

Blockchain is a burgeoning technology that enables value transfer and storage in a decentralized manner without a trusted third party. It can help protect the value of a digital asset from being copied and infringed which is very important in the data era we are now experiencing. Researchers from various fields are trying to apply blockchain technology to different research areas to explore new solutions for the traditional problems that are hard to solve with other existing technologies. Vehicular networks are one of the most frequently mentioned research fields that are trying to integrate with blockchain technology.

Although blockchain technology has become one of the most popular topics in either industrial or academic fields, it is still in the early stages of its development. The current blockchain technology is still ineligible when it is used to combine with systems that have high transaction frequency and dynamic networks like the IoT system. Especially when it is used in combination with the IoV system, its shortcomings are particularly obvious. To make the blockchain technology more suitable for the IoV systems, the customized design of the blockchain system is essential to improve system performance and resource utilization efficiency

In chapter 3, a multi-channel blockchain scheme for IoV is proposed to improve the performance of the blockchain system in the IoV environment. The performance of the deployed blockchain system with different parameters under different vehicular densities is investigated first to find the best configuration under different circumstances. Then multiple channels with different parameters are configured in the system where each channel is optimized for a certain level of vehicle density and application requirements. A channel selection algorithm is proposed in this chapter to help the vehicles select the most suitable channel to send their messages according to the application requirements and the traffic conditions.

The performance of the proposal is evaluated under the Hyperledger Fabric platform. Simulation results show that the proposed multi-channel scheme can significantly increase the performance of the blockchain system under a different number of vehicles. The throughput performance of the system can be improved up to 28.45%, from 92.29 transaction/s (with 100 TNX/block) to 118.55 transaction/s (with the proposal) as shown in Figure 3-9. The latency performance of the system can be reduced up to 35.7%, from 6.333 seconds (with 600 TNX/block) to 4.073 seconds (with the proposal) as shown in Figure 3-14.

In chapter 4, the difficulties of deploying blockchain systems in IoV systems are investigated. It is essential to improve resource utilization efficiency and the flexibility of the system while applying blockchain technology to vehicular networks. To address these issues, a hierarchical blockchain resource scheduling scheme for the IoV system is proposed to make blockchain technology more suitable for IoV networks.

The computing resources in the network are divided into three layers and the proposed scheme improves the performance of blockchain-enabled IoV systems by efficiently adjusting the resource scheduling with the proposed resource control algorithm and scaling control algorithm. The resource control algorithm increases the CPU share of the Fabric peer to improve the priority of the peer which is the performance bottleneck of the Execute-Order-Validate-based blockchain architecture. The scaling control algorithm helps the system to scale up locally or remotely according to the resource operating status. A resource monitoring system is developed to cooperate with the above algorithms to implement the resource scheduling of the system. The monitor collects the operating status of the system and calculates the throughput, latency, and transaction success ratio of the system.

Extensional simulations are conducted to evaluate the proposed hierarchical blockchain resource scheduling scheme and the results fully demonstrate the superiority of the proposal. The proposal can reduce the latency performance of the system up to 49.13% when the sending rate is 320 transactions per second, from 4.05 seconds (with even allocation under limited resources) to 2.06 seconds (with the proposal under limited resources) as shown in Figure 4-8.

The proposed scheme can also provide a better scaling strategy which has better performance than the situation under the same resources with even allocation (up to 60.7% when the sending rate is 320 transactions per second) as shown in Figure 4-14 and better performance-cost ratio than the situation under full configuration with even allocation (up to 71.23% when the sending rate is 110 transactions per second) as shown in Figure 4-15.

5.2 Future work

This thesis proposes a multi-channel blockchain scheme for the IoV system to improve the performance of the blockchain-enabled IoV systems from the blockchain parameter perspective. The proposed multi-channel blockchain scheme and channel selection algorithm are independent of the consensus mechanism of the blockchain system, thereby they are applicable on many different types of blockchain platforms. In future work, the proposed scheme will be tested on multiple other platforms to further evaluate the generality of the proposed scheme. In addition, other parameters such as batch time out and adaptive block size will also be taken into consideration to further improve the flexibility of the system.

A hierarchical blockchain resource scheduling scheme for the IoV system is proposed to improve the resource utilization efficiency and performance of the blockchain-enabled IoV system. It mainly helps improve the performance of the peers in the networks. In the future, the characteristics of the ordering services should also be tested and customized control methods for the ordering peers should also be taken into consideration to achieve an integrated optimization for the blockchain technology in the IoV networks. Multistep thresholds and more elaborate resource division are also under consideration for making the performance of the system more flexible and graceful. The integration of the two contributions is also a potential direction for future research.

Moreover, during simulations, the Caliper cannot fully satisfy the experimental requirements in some particular situations. Therefore, some customized modification of the Caliper benchmark for vehicular networks is needed to simulate a more realistic blockchain-enabled IoV environment.

Reference

[1] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswam, “Internet of Things (IoT): A vision, architectural elements, and future directions,” Future Generation Computer Systems, vol. 29, pp. 1645-1660,2013.

[2] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, “A Survey on the Internet of Things (IoT) Forensics: Challenges, Approaches, and Open Issues,” IEEE Communications Surveys & Tutorials, vol. 22, no. 2, pp. 1191--1221, Second quarter, 2020.

[3] I. Lee, K. Lee, “The Internet of Things (IoT): Applications, investments, and challenges for enterprises,” Business Horizons, vol. 58, pp. 431-440, Jul-Aug. 2015.

[4] S. Buda et al., “Empowering Blockchain in Vehicular Environments With Decentralized Edges,” IEEE Access, vol. 8, pp. 202032-202041, Nov. 2020.

[5] C. Wu, Y. Ji, F. Liu, S. Ohzahata and T. Kato, “Toward Practical and Intelligent Routing in Vehicular Ad Hoc Networks,” IEEE Transactions on Vehicular Technology, vol. 64, no. 12, pp. 5503-5519, Dec. 2015.

[6] S. Jaffry, R. Hussain, X. Gui, and S. F. Hasan, “A Comprehensive Survey on Moving Networks,” IEEE Communications Surveys & Tutorials, vol. 23, no. 1, pp. 110-136, First quarter 2021.

[7] J.A. Onieva, R. Rios, R. Roman, J. Lopez, “Edge-Assisted Vehicular Networks Security,” IEEE Internet Things Journal, vol. 6, pp. 8038-8045, Oct. 2019.

[8] F. Zhu, Y. Lv, Y. Chen, X. Wang, G. Xiong and F. -Y. Wang, “Parallel Transportation Systems: Toward IoT-Enabled Smart Urban Traffic Control and Management,” IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 10, pp. 4063-4071, Oct. 2020.

[9] Z. Wu, C. Wu, T. Yoshinaga, K.-L. A. Yau, Y. Ji, and J. Li, “Federated Learning for Vehicular Internet of Things: Recent Advances and Open Issues,” IEEE Open Journal of the Computer Society, vol. 1, no. 10, pp. 45-61, May 2020.

[10] W. He, G. Yan, and L. D. Xu, “Developing Vehicular Data Cloud Services in the IoT Environment, ” IEEE Transactions on Industrial Informatics, vol. 10, pp. 1587-1595, May 2014.

[11] C. Wu, Z. Liu, F. Liu, T. Yoshinaga, Y. Ji, and J. Li, “Collaborative Learning of Communication Routes in Edge-enabled Multi-access Vehicular Environment,” IEEE Transactions on Cognitive Communications and Networking, vol. 6, no. 4, pp. 1155-1165, Dec. 2020.

[12] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B.A.Y. Arcas, “Communication-Efficient Learning of Deep Networks from Decentralized Data,” In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Ft. Lauderdale, FL, USA, vol. 20-22, pp. 1273-1282, April 2017.

[13] M. M. Zanjireh, and H. Larijani, “A Survey on Centralised and Distributed Clustering Routing Algorithms for WSNs,” Proc. 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), Glasgow, 2015, pp. 1-6, 2015.

[14] H. I. Abbasi, R.C. Voicu, J. Copeland, and Y. Chang, “Towards Fast and Reliable Multi-hop Routing in VANETs. IEEE Trans,” IEEE Transactions on Mobile Computing, vol. 19, no. 10, pp. 2461-2474, Oct. 2019.

[15] 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, Sep/Oct 2020.

[16] C.Wu, X. Chen, Y. Ji, S. Ohzahata, and T. Kato, “Efficient broadcasting in VANETs using dynamic backbone and network coding,” IEEE Transactions on Wireless Communications, vol. 14, no. 11, pp. 6057-6071, Nov. 2015.

[17] C. Wu, Z. Liu, D. Zhang, T. Yoshinaga, and Y. Ji, “Spatial Intelligence towards Trustworthy Vehicular IoT”, IEEE Communications Magazine, vol. 56, no. 10, pp. 22-27, Oct. 2018.

[18] C. Wu, T. Yoshinaga, Y. Ji, T. Murase, and Y. Zhang, “A Reinforcement Learning-Based Data Storage Scheme for Vehicular Ad Hoc Networks,” IEEE Transactions on Vehicular Technology, vol. 66, pp. 6336-6348, 2017.

[19] 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, June 2019.

[20] H. Peng, Le Liang, X. Shen, and G. Y. Li, "Vehicular Communications: A Network Layer Perspective," IEEE Transactions on Vehicular Technology, vol. 68, no. 2, pp. 1064-1078, Feb. 2019.

[21] R. Shrestha and S.Y. Nam, "Regional Blockchain for Vehicular Networks to Prevent 51% Attacks," IEEE Access, vol. 7, pp. 95033-95045, 2019.

[22] Y. Wang, Z. Su, and N. Zhang, "BSIS: Blockchain-Based Secure Incentive Scheme for Energy Delivery in Vehicular Energy Network," IEEE Transactions on Industrial Informatics, vol. 15, pp. 3620-3631, 2019.

[23] I. Butun, P. Österberg, and H. Song, "Security of the Internet of Things: Vulnerabilities, Attacks, and Countermeasures, " in IEEE Communications Surveys & Tutorials, vol. 22, no. 1, pp. 616-644, First quarter 2020.

[24] K. Jia, Y. Hou, K. Niu, C. Dong, and Z. He, "The Delay-Constraint Broadcast Combined With Resource Reservation Mechanism and Field Test in VANET," IEEE Access, vol. 7, pp. 59600-59612, 2019.

[25] J. E. Siegel, D.C. Erb, and S.E. Sarma, "A Survey of the Connected Vehicle Landscape—Architectures, Enabling Technologies, Applications, and Development Areas," IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 8, pp. 2391-2406, Aug. 2018.

[26] L. Li, J. Liu, L. Cheng, S. Qiu, W. Wang, X. Zhang, and Z. Zhang, "CreditCoin: A Privacy-Preserving Blockchain-Based Incentive Announcement Network for Communications of Smart Vehicles," IEEE Transactions on Intelligent Transportation Systems, vol. 19, pp. 2204-2220, 2018.

[27] X. Chen, C. Wu, M. Bennis, Z. Zhao, and Z. Han, "Learning to Entangle Radio Resources in Vehicular Communications: An Oblivious Game-Theoretic Perspective," IEEE Transactions on Vehicular Technology, vol. 68, pp. 4262-4274, 2019.

[28] Nakamoto, S. "Bitcoin: A Peer-to-Peer Electronic Cash System." bitcoin.org. https://bitcoin.org/bitcoin.pdf (accessed on 11 Oct 2021).

[29] H. N. Dai, Z. Zheng, and Y. Zhang, “Blockchain for the Internet of Things: A Survey,” IEEE Internet Things J, vol.6, pp.8076-8094, 2019.

[30] W. Viriyasitavat, L.D. Xu, Z. Bi, and D. Hoonsopon, “Blockchain Technology for Applications in Internet of Things—Mapping From System Design Perspective,” IEEE Internet Things Journal, vol. 6, pp.8155-8168, 2019.

[31] M.A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke, “Blockchain Technologies for the Internet of Things: Research Issues and Challenges,” IEEE Internet Things Journal, vol. 6, pp. 2188-2204, 2019.

[32] H. A. Khattak, H. Farman, B. Jan, and I.U. Din, “Toward Integrating Vehicular Clouds with IoT for Smart City Services,” IEEE Network, vol. 33, pp. 65-71, 2019.

[33] Y. Wang, P. Lang, D. Tian, J. Zhou, X. Duan, Y. Cao, and D. Zhao, “A Game-based Computation Offloading Method in Vehicular Multi-access Edge Computing Networks,” IEEE Internet Things Journal, vol. 7, no. 6, pp. 4987-4996, June 2020.

[34] B. Alotaibi, “Utilizing Blockchain to Overcome Cyber Security Concerns in the Internet of Things: A Review,” IEEE Sensors Journal, vol. 19, pp. 10953-10971, 2019.

[35] X. Liu, H. Huang, F. Xiao, and Z. Ma, “A Blockchain-Based Trust Management With Conditional Privacy-Preserving Announcement Scheme for VANETs,” IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4101--4112, May 2020.

[36] D. Zhang, F.R. Yu, and R. Yang, “Blockchain-Based Distributed Software-Defined Vehicular Networks: A Dueling Deep Q -Learning Approach,” IEEE Transactions on Cognitive Communications and Networking, vol. 5, pp. 1086-1100, 2019.

[37] M. R. Jabbarpour, M. Sookhak, S. SeyedFarshi, and A. Zomaya, “Proposing a Secure Self-fining Vehicle Using Blockchain and Vehicular Edge Computing,” IEEE Consumer Electronics Magazine, vol. 11, no. 2, pp. 28-35, 1 March 2022.

[38] X. Zhang, J. Zhang, Z. Liu, Q. Cui, X. Tao, and S. Wang, “MDP-based Task Offloading for Vehicular Edge Computing under Certain and Uncertain Transition Probabilities,” IEEE Transactions on Vehicular Technology, vol. 69, no. 3, pp. 3296-3309, March 2020.

[39] X. Fu, R. Yu, J. Wang, Q. Qi, and J. Liao, “Performance Optimization for Blockchain-

Enabled Distributed Network Function Virtualization Management and Orchestration (NFV-MANO),” IEEE Transactions on Vehicular Technology, vol. 69, no. 6, pp. 6670-6679, June 2020.

[40] M. A. Khan and K. Salah, “IoT security: Review, blockchain solutions, and open challenges” Future Generation Computer Systems, vol. 82, pp. 395–411, 2018.

[41] S. F. Wamba, J. R. K. Kamdjoug, R. E. Bawack, and J. G. Keogh, “Bitcoin, Blockchain and Fintech: A systematic review and case studies in the supply chain,” Production Planning & Control, vol. 31, pp. 115–142, 2020.

[42] H. Treiblmaier, “The impact of the blockchain on the supply chain: a theory-based research framework and a call for action,” Supply Chain Management, vol. 23, pp. 545-559, 2018.

[43] F. Tschorsch and B. Scheuermann, “Bitcoin and beyond: A technical survey on decentralized digital currencies,” IEEE Communications Surveys & Tutorials., vol. 18, no. 3, pp. 2084--2123, 3rd Quart., 2016.

[44] M. Wu, K. Wang, X. Cai, S. Guo, M. Guo, and C. Rong, “A comprehensive survey of blockchain: From theory to IoT applications and beyond,” IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8114--8154, Oct. 2019.

[45] M. Singh, A. Singh, and S. Kim, “Blockchain: A game changer for securing IoT data,” In Proceedings of the IEEE 4th World Forum on Internet of Things (WF-IoT), Singapore, vol. 5-8, pp. 51-55, Feb. 2018.

[46] Ethereum. “Ethereum Whitepaper.” ethereum.org https://ethereum.org/en/whitepaper (accessed on 11 Oct 2021).

[47] W. Wang et al., “A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks,” IEEE Access, vol. 7, pp. 22328--22370, Jan. 2019.

[48] O. Novo, “Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT,” IEEE Internet Things Journal, vol. 5, pp. 1184-1195, 2018.

[49] L. Xie, Y. Ding, H. Yang, and X. Wang, “Blockchain-Based Secure and Trustworthy Internet of Things in SDN-Enabled 5G-VANETs,” IEEE Access, vol. 7, pp. 56656-56666, 2019.

[50] A. A. Monrat, O. Schelén, and K. Andersson, "A Survey of Blockchain From the Perspectives of Applications, Challenges, and Opportunities," in IEEE Access, vol. 7, pp. 117134-117151, 2019.

[51] K. Yeow, A. Gani, R.W. Ahmad, J.J.P.C. Rodrigues, and K. Ko, “Decentralized Consensus for Edge-Centric Internet of Things: A Review, Taxonomy, and Research Issues,” IEEE Access, vol. 6, pp. 1513-1524, 2018.

[52] T. M. Fernandez-Carames and P. Fraga-Lamas, “A Review on the Use of Blockchain for the Internet of Things,” IEEE Access, vol.6, pp. 32979-33001, 2018.

[53] M. S. Ali, M. Vecchio, M. Pincheira, K. Dolui, F. Antonelli, and M. H. Rehmani, “Applications of Blockchains in the Internet of Things: A Comprehensive Survey,” IEEE Communications Surveys & Tutorials, vol. 21, no. 2, pp. 1676-1717, Second quarter, 2019.

[54] W. Yang, E. Aghasian, S. Garg, D. Herbert, L. Disiuta, and B. Kang, “A Survey on Blockchain-Based Internet Service Architecture: Requirements, Challenges, Trends, and Future,” IEEE Access, vol. 7, pp. 75845-75872, 2019.

[55] F. Lombardi, L. Aniello, S. De Angelis, A. Margheri, and V. Sassone, “A blockchain-based infrastructure for reliable and cost-effective IoT-aided smart grids,” In Proceedings of the Living in the Internet of Things: Cybersecurity of the IoT—2018, London, UK, vol. 28-29, pp. 1-6, Mar. 2018.

[56] Z. Ma, J. Zhang, Y. Guo, Y. Liu, X. Liu, and W. He, “An Efficient Decentralized Key Management Mechanism for VANET with Blockchain,” IEEE Transactions on Vehicular Technology, vol. 69, no. 6, pp. 5836-5849, June 2020.

[57] L. Wang, X. Lin, E. Zima, and C. Ma, “Towards Airbnb-Like Privacy-Enhanced Private Parking Spot Sharing Based on Blockchain,” IEEE Transactions on Vehicular Technology, vol. 69, pp. 2411-2423, March 2020.

[58] Z. Su, Y. Wang, Q. Xu, and N. Zhang, “LVBS: Lightweight Vehicular Blockchain for Secure Data Sharing in Disaster Rescue,” IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 1, pp. 19-32, 1 Jan.-Feb. 2022.

[59] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, “When Mobile Blockchain Meets

Edge Computing," IEEE Communications Magazine, vol. 56, no. 8, pp. 33-39, Aug. 2018.

[60] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B.C. Ooi, and J. Wang, "Untangling Blockchain: A Data Processing View of Blockchain Systems," IEEE Transactions on Knowledge and Data Engineering, vol. 30, pp. 1366-1385, 2018.

[61] K. Gai, J. Guo, L. Zhu, and S. Yu, "Blockchain Meets Cloud Computing: A Survey, "IEEE Communications Surveys & Tutorials, vol. 22, no. 3, pp. 2009--2030, third quarter 2020.

[62] M. A. Uddin, A. Stranieri, I. Gondal and V. Balasubramanian, "Continuous Patient Monitoring With a Patient-Centric Agent: A Block Architecture," in IEEE Access, vol. 6, pp. 32700-32726, 2018.

[63] J. Al-Jaroodi and N. Mohamed, "Blockchain in Industries: A Survey," in IEEE Access, vol. 7, pp. 36500-36515, 2019.

[64] M. Shen, J. Zhang, L. Zhu, K. Xu, X. Tang, "Secure SVM Training over Vertically-Partitioned Datasets using Consortium Blockchain for Vehicular Social Networks," IEEE Transactions on Vehicular Technology, vol. 69, no. 6, pp. 5773-5783, June 2020.

[65] X. Zheng, M. Li, Y. Chen, J. Guo, M. Alam, and W. Hu, "Blockchain-Based Secure Computation Offloading in Vehicular Networks," IEEE Transactions on Vehicular Technology, vol. 69, no. 6, pp. 5773-5783, June 2020.

[66] M. B. Mollah et al., "Blockchain for the Internet of Vehicles towards Intelligent Transportation Systems: A Survey," IEEE Internet of Things Journal, vol. 8, no. 6, pp. 4157-4185, Mar. 2021.

[67] C. Peng, C. Wu, L. Gao, J. Zhang, K.-L. Alvin Yau, and Y. Ji, "Blockchain for Vehicular Internet of Things: Recent Advances and Open Issues," Sensors, vol. 20, no. 18: 5079, Sep. 2020.

[68] P. K. Singh, R. Singh, S. K. Nandi, K. Z. Ghafoor, D. B. Rawat, and S. Nandi, "Blockchain-Based Adaptive Trust Management in Internet of Vehicles Using Smart Contract," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 6, pp. 3616-3630, June 2021.

[69] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C. P. A. Ogah, and Z. Sun, "Blockchain-Based Dynamic Key Management for Heterogeneous Intelligent Transportation Systems, " IEEE

Internet of Things Journal, vol. 4, no. 6, pp. 1832-1843, Dec. 2017.

[70] S. Xia, F. Lin, Z. Chen, C. Tang, Y. Ma, and X. Yu, “A Bayesian Game Based Vehicle-to-Vehicle Electricity Trading Scheme for Blockchain-Enabled Internet of Vehicles,” in IEEE Transactions on Vehicular Technology, vol. 69, no. 7, pp. 6856-6868, July 2020.

[71] L. Luo, J. Feng, H. Yu, and G. Sun, “Blockchain-Enabled Two-way Auction Mechanism for Electricity Trading in Internet of Electric Vehicles,” in IEEE Internet of Things Journal, DOI: 10.1109/JIOT.2021.3082769.

[72] J. Gao et al., “A Blockchain-SDN-Enabled Internet of Vehicles Environment for Fog Computing and 5G Networks,” IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4278-4291, May. 2020.

[73] F. Ayaz, Z. Sheng, D. Tian, and Y. L. Guan, “A Proof-of-Quality-Factor (PoQF) based Blockchain and Edge Computing for Vehicular Message Dissemination,” IEEE Internet of Things Journal, vol. 8, no. 4, pp. 2468--2482, Feb. 2021.

[74] B. Yin, Y. Wu, T. Hu, J. Dong, and Z. Jiang, “An Efficient Collaboration and Incentive Mechanism for Internet of Vehicles (IoV) With Secured Information Exchange Based on Blockchains,” IEEE Internet of Things Journal, vol. 7, no. 3, pp. 1582-1593, March 2020.

[75] Z. Zhou, B. Wang, Y. Guo, and Y. Zhang, “Blockchain and Computational Intelligence Inspired Incentive-Compatible Demand Response in Internet of Electric Vehicles,” IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 3, no. 3, pp. 205-216, June 2019.

[76] W. Chen, Y. Chen, X. Chen and Z. Zheng, “Toward Secure Data Sharing for the IoV: A Quality-Driven Incentive Mechanism With On-Chain and Off-Chain Guarantees,” IEEE Internet of Things Journal, vol. 7, no. 3, pp. 1625-1640, March 2020.

[77] Z. Xiong, S. Feng, W. Wang, D. Niyato, P. Wang and Z. Han, “Cloud/Fog Computing Resource Management and Pricing for Blockchain Networks,” IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4585-4600, June. 2019.

[78] C. Chen, J. Wu, H. Lin, W. Chen and Z. Zheng, “A Secure and Efficient Blockchain-Based Data Trading Approach for Internet of Vehicles,” IEEE Transactions on Vehicular Technology, vol. 68, no. 9, pp. 9110-9121, Sept. 2019.

[79] L. Jiang, B. Chen, S. Xie, S. Maharjan and Y. Zhang, “Incentivizing Resource Cooperation for Blockchain Empowered Wireless Power Transfer in UAV Networks,” IEEE Transactions on Vehicular Technology, vol. 69, no. 12, pp. 15828--15841, Dec. 2020.

[80] X. Zhang and X. Chen, “Data Security Sharing and Storage Based on a Consortium Blockchain in a Vehicular Ad-hoc Network,” IEEE Access, vol. 7, pp. 58241--58254, Jan. 2019.

[81] Y. Song, Y. Fu, F. R. Yu, and L. Zhou, “Blockchain-Enabled Internet of Vehicles With Cooperative Positioning: A Deep Neural Network Approach,” in IEEE Internet of Things Journal, vol. 7, no. 4, pp. 3485-3498, April 2020.

[82] J. Feng, Y. Wang, J. Wang, and F. Ren, “Blockchain-based Data Management and Edge-assisted Trusted Cloaking Area Construction for Location Privacy Protection in Vehicular Networks,” IEEE Internet of Things Journal, vol. 8, no. 4, pp. 2087--2101, Feb. 2021.

[83] C. Li, Y. Fu, F. R. Yu, T. H. Luan, and Y. Zhang, “Vehicle Position Correction: A Vehicular Blockchain Networks-Based GPS Error Sharing Framework,” in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 2, pp. 898-912, Feb. 2021.

[84] S. Kim, “Impacts of Mobility on Performance of Blockchain in VANET,” IEEE Access, vol. 7, pp. 68646-68655, May 2019.

[85] T. S. L. Nguyen, G. Jourjon, M. Potop-Butucaru, and K. L. Thai, “Impact of network delays on Hyperledger Fabric," IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 222-227, April 2019.

[86] Parth Thakkar and Senthilnathan Natarajan, “Scaling Blockchains Using Pipelined Execution and Sparse Peers,” In Proceedings of the ACM Symposium on Cloud Computing (SoCC '21), Association for Computing Machinery, New York, NY, USA, pp. 489–502, 2021.

[87] A. Sharma, F. M. Schuhknecht, D. Agrawal, and J. P. Dittrich, “Blurring the Lines between Blockchains and Database Systems: The Case of Hyperledger Fabric,” In Proceedings of the 2019 International Conference on Management of Data (SIGMOD '19), Association for Computing Machinery, New York, NY, USA, 2019.

[88] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, “Distributed Resource Allocation

in Blockchain-Based Video Streaming Systems With Mobile Edge Computing," IEEE Transactions on Wireless Communications, vol. 18, no. 1, pp. 695-708, Jan. 2019.

[89] D. Liu, J. Ni, X. Lin, and X. S. Shen, "Transparent and Accountable Vehicular Local Advertising with Practical Blockchain Designs," IEEE Transactions on Vehicular Technology, vol. 69, no. 12, pp. 5694-15705, Dec. 2020.

[90] R. Li, T. Song, B. Mei, H. Li, X. Cheng, and L. Sun, "Blockchain For Large-Scale Internet of Things Data Storage and Protection," IEEE Transactions on Services Computing, vol. 12, pp.762-771, 2019.

[91] J. Chen, G. Mao, C. Li, A. Zafar, and A. Y. Zomaya, "Throughput of Infrastructure-Based Cooperative Vehicular Networks," IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 11, pp. 2964-2979, Nov. 2017.

[92] L. Cui et al., "A Blockchain-based Containerized Edge Computing Platform for the Internet of Vehicles," IEEE Internet of Things Journal, vol. 8, no. 4, pp.2395-2408, Feb. 2021.

[93] L. Nie, X. Wang, W. Sun, Y. Li, S. Li, and P. Zhang, "Imitation-Learning-Enabled Vehicular Edge Computing: Toward Online Task Scheduling," in IEEE Network, vol. 35, no. 3, pp. 102-108, May/June 2021.

[94] X. Ge, H. Cheng, G. Mao, Y. Yang, and S. Tu, "Vehicular Communications for 5G Cooperative Small-Cell Networks," IEEE Transactions on Vehicular Technology, vol. 65, no. 10, pp. 7882-7894, Oct. 2016.

[95] M. Chowdhury and K. Dey, "Intelligent Transportation Systems-A Frontier for Breaking Boundaries of Traditional Academic Engineering Disciplines [Education]," IEEE Intelligent Transportation Systems Magazine, vol. 8, no. 1, pp. 4-8, Spring 2016.

[96] H. Lin, S. Garg, J. Hu, G. Kaddoum, M. Peng and M. S. Hossain, "Blockchain and Deep Reinforcement Learning Empowered Spatial Crowdsourcing in Software-Defined Internet of Vehicles," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 6, pp. 3755-3764, June 2021.

[97] H. Wang, Q. Wang, D. He, Q. Li, and Z. Liu, "BBARS: Blockchain-Based Anonymous Rewarding Scheme for V2G Networks," IEEE Internet Things Journal, vol.6, pp.3676-

3687, 2019.

[98] S.R. Pokhrel and J. Choi, “Federated Learning with Blockchain for Autonomous Vehicles: Analysis and Design Challenges,” IEEE Transactions on Communications, vol. 68, no. 8, pp. 4734-4746, Aug. 2020.

[99] K. Christidis and M. Devetsikiotis, “Blockchains and Smart Contracts for the Internet of Things,” IEEE Access, vol.4, pp.2292-2303, 2016.

[100] N. Kshetri, “Can Blockchain Strengthen the Internet of Things?” IT Professional vol. 19, pp.68-72, 2017.

[101] D. Miller, “Blockchain and the Internet of Things in the Industrial Sector,” IT Professional, vol.20, pp.15-18, 2018.

[102] Y. Wang, Z. Su, K. Zhang, and A. Benslimane, “Challenges and Solutions in Autonomous Driving: A Blockchain Approach,” IEEE Network, vol. 34, no. 4, pp. 218-226, July/August 2020.

[103] G. Qiao, S. Leng, K. Zhang, and Y. He, “Collaborative Task Offloading in Vehicular Edge Multi-Access Networks,” IEEE Communications Magazine, vol. 56, no. 8, pp. 48-54, Aug. 2018.

[104] 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, pp.3278-3289, 2020.

[105] B. Li, R. Liang, D. Zhu, W. Chen, and Q. Lin, “Blockchain-Based Trust Management Model for Location Privacy Preserving in VANET,” IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 6, pp. 3765-3775, June 2021.

[106] S. S. Shah, A. W. Malik, A. U. Rahman, S. Iqbal, and S.U. Khan, “Time Barrier-Based Emergency Message Dissemination in Vehicular Ad-hoc Networks,” IEEE Access, vol. 7, pp.16494-16503, 2019.

[107] M. N. M. Bhutta et al., “A Survey on Blockchain Technology: Evolution, Architecture and Security, ” in IEEE Access, vol. 9, pp. 61048-61073, 2021.

[108] A. Hafid, A. S. Hafid, and M. Samih, “Scaling Blockchains: A Comprehensive Survey, ” in IEEE Access, vol. 8, pp. 125244-125262, 2020.

[109] 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, ” in IEEE Communications Surveys & Tutorials, vol. 21, no. 2, pp. 1508-1532, Second quarter 2019.

[110] W.-Y. Shieh, C.-C.J. Hsu, C.-H. Lin, and T.-H. Wang, “Investigation of Vehicle Positioning by Infrared Signal-Direction Discrimination for Short-Range Vehicle-to-Vehicle Communications,” IEEE Transactions on Vehicular Technology, vol.67, pp.11563-11574, 2018.

[111] W. Alasmary, H. Sadeghi, and S. Valaee, “Strategic Sensing in Vehicular Networks Using Known Mobility Information,” IEEE Transactions on Vehicular Technology, vol. 67, pp. 1932-1945, 2018.

[112] L. Ding, Y. Wang, P. Wu, L. Li, and J. Zhang, “Kinematic Information Aided User-Centric 5G Vehicular Networks in Support of Cooperative Perception for Automated Driving,” IEEE Access, vol. 7, pp. 40195-40209, 2019.

[113] C. Wu, X. Chen, Y. Ji, F. Liu, S. Ohzahata, T. Yoshinaga, and T. Kato, “Packet Size-Aware Broadcasting in VANETs With Fuzzy Logic and RL-Based Parameter Adaptation,” IEEE Access, vol.3, pp.2481-2491, 2015.

[114] X. Zhang and D. Wang, “Adaptive Traffic Signal Control Mechanism for Intelligent Transportation Based on a Consortium Blockchain,” IEEE Access, vol.7, pp.97281-97295, 2019.

[115] C. Fan, S. Ghaemi, H. Khazaei, and P. Musilek, “Performance Evaluation of Blockchain Systems: A Systematic Survey,” in IEEE Access, vol. 8, pp. 126927-126950, 2020.

[116] J. A. Chacko, R. Mayer, and H. Jacobsen, “Why Do My Blockchain Transactions Fail? A Study of Hyperledger Fabric,” In Proceedings of the 2021 International Conference on Management of Data, Association for Computing Machinery, New York, NY, USA, pp. 221–234, 2021.

[117] C. Xu, K. Wang, P. Li, S. Guo, J. Luo, B. Ye, and M. Guo, “Making Big Data Open in

Edges: A Resource-Efficient Blockchain-Based Approach," IEEE Transactions on Parallel and Distributed Systems, vol. 30, no. 4, pp. 870-882, April 2019.

[118] Y. T. Yang, L. D. Chou, C. W. Tseng, F. H. Tseng, and C. C. Liu, "Blockchain-Based Traffic Event Validation and Trust Verification for VANETs," IEEE Access, vol.6, pp.30868-30877, 2019.

[119] Z. Zhou, P. Liu, J. Feng, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Computation Resource Allocation and Task Assignment Optimization in Vehicular Fog Computing: A Contract-Matching Approach," IEEE Transactions on Vehicular Technology, vol. 68, no. 4, pp. 3113--3125, April. 2019.

[120] H. Huang, H. Li, C. Shao, T. Sun, W. Fang, and S. Dang, "Data Redundancy Mitigation in V2X Based Collective Perceptions," IEEE Access, vol. 8, pp.13405-13418, 2020.

[121] C. Wu, S. Ohzahata, and T. Kato, "Portable, and Practicable Solution for Routing in VANETs: A Fuzzy Constraint Q-Learning Approach," IEEE Transactions on Vehicular Technology, vol. 62, pp. 4251-4263, 2013.

[122] C. Wu, S. Ohzahata, and T. Kato, "VANET Broadcast Protocol Based on Fuzzy Logic and Lightweight Retransmission Mechanism," IEICE Transactions on Communications, vol.95-B, pp.415-425, 2012.

[123] H. Li, L. Pei, D. Liao, S. Chen, M. Zhang, and D. Xu, "FADB: A Fine-Grained Access Control Scheme for VANET Data Based on Blockchain," IEEE Access, vol.8, pp.85190-85203, 2020.

[124] H. Javaid, C. Hu, and G. Brebner, "Optimizing Validation Phase of Hyperledger Fabric," 2019 IEEE 27th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), pp. 269-275, 2019.

List of Abbreviation

AI	Artificial Intelligence
AP	Access Point
BIoV	Blockchain-enabled Internet of Vehicles
CBECI	Cambridge Bitcoin Electricity Consumption Index
CFT	Crash Fault Tolerance
DAG	Directed Acyclic Graph
DLT	Distributed ledger technology
DNN	Deep Neural Network
EV	Electric Vehicles
EOV	Execute-Order-Validate
ETC	Electronic Toll Collection
FFT	Fast Fourier Transform
IoT	Internet of Things
IoV	Internet of Vehicles
ITS	Intelligent Transportation System
MANET	Mobile Ad-hoc networks
MVCC	Multi-Version Concurrency Control
OE	Order-Execute
P2P	Peer-2-Peer
PoC	Proof-of-Collaboration
PoW	Proof-of-Work
SBM	Safety Beacon Messages
SDN	Software-Defined Networking
TTP	Trusted Third Party
UTXO	Unspent Transaction Output
V2C	Vehicle-to-Cloud
V2I	Vehicle-to-Infrastructure
V2P	Vehicle-to-Pedestrian

V2S	Vehicle-to-Sensor
V2V	Vehicle-to-Vehicle
VANET	Vehicular Ad Hoc Network
VSCC	Validation System Chaincode
WMN	Wireless Mesh Networks

Author Biography

Liming Gao received the BE degree from China Agricultural University, China, in 2010, and the ME degree from Coventry University, UK, in 2015.

Since 2017, he has been working as a Ph.D. candidate at the Department of Computer and Network Engineering, Graduate School of Informatics and Engineering, The University of Electro-Communications, Japan.

His current research interests include the Internet of Things, vehicular networks, blockchain technology.

List of Publications

Journal Paper

[1] Liming Gao, Celimuge Wu, Tsutomu Yoshinaga, Xianfu Chen, and Yusheng Ji, "Multi-Channel Blockchain Scheme for Internet of Vehicles," in IEEE Open Journal of the Computer Society, vol. 2, pp. 192-203, 2021.

(Related to the content of Chapter 3)

[2] Chunrong Peng, Liming Gao, Celimuge Wu, Jiefang Zhang, Kok-Lim Alvin Yau, and Yusheng Ji, "Blockchain for Vehicular Internet of Things: Recent Advances and Open Issues," Sensors 20, no. 18: 5079, 2020.

(Related to the content of Chapter 2)

[3] Celimuge Wu, Chunrong Peng, Zhaoyang Du, Liming Gao, Tsutomu Yoshinaga, and Yusheng Ji, "Toward Agile Information and Communication Framework for the Post-COVID-19 Era," in IEEE Open Journal of the Computer Society, vol. 2, no. 01, pp. 290-299, 2021.

(Related to the content of Chapter 2)

International Conference

[1] Liming Gao, Celimuge Wu, Tsutomu Yoshinaga, Wugedele Bao, and Siri Guleng, "Resource Management for Blockchain-enabled Internet of Vehicles," 2021 International Conference on Information and Communication Technologies for Disaster Management (ICT-DM), 6 pages, Oct 2021(accepted).

（Related to the content of Chapter 4）

[2] Liming Gao, Celimuge Wu, Tsutomu Yoshinaga, and Yusheng Ji, "Performance Evaluation of RPL-Based Sensor Data Collection in Challenging IoT Environment, " Proceedings of the 2nd International Conference on Healthcare Science and Engineering, pp.275-285, May 2019.

（Related to the content of Chapter 2）

Other Conference

[1] Liming Gao, Celimuge Wu, and Tsutomu Yoshinaga, “Performance Evaluation of RPL-Based Sensor Data Collection in Challenging IoT Environment,” IEICE Technical Report, vol. 116, no. 403, CQ2018-24, pp. 7-12, Jan 2017.