Vehicle Information Management System Using Hyperledger Fabric

Anwesha Banik, Sukanta Chakraborty, and Abhishek Majumder

Abstract The growing popularity of motor vehicles among us has accelerated the growth of the automotive industry. But with the increase in sales, there is a huge burden on the Regional Transport Office (RTO) to manage vehicle data. One of the most difficult challenges in today's vehicle data management systems is ensuring the integrity and confidentiality of vehicle data. The RTO has full control over vehicle data, which in turn encourages dishonest employees to abuse their data manipulation rights. By submitting false documents, a stolen or smuggled vehicle from one state can be legitimately driven in another state. The aforementioned problems in vehicle data management systems can be addressed by "blockchain" technology. A blockchain is a decentralised, append-only ledger that is replicated among all the peers present in the network. A blockchain-based way out has been proposed for the smooth functioning of the vehicle registration system. The proposed framework consists of three sub-modules that provide a blockchain-based architecture for registration of new vehicles, querying vehicles, and interstate transfer of vehicles. The framework is built on Hyperledger Fabric. Hyperledger Fabric is a permissioned blockchain and has certain features that are best suitable for business applications. Each sub-module is evaluated with respect to throughput, latency, and send rate.

Keywords Regional Transport Office · Data mutability · Permissioned blockchain

1 Introduction

The automotive industry is one of the largest sectors in India. It accounts for about 7.14% of country's GDP [1]. India was the largest producer of two-wheeler across the globe in 2019. In the financial year 2021, there were about 3.8 million two-wheeler and passenger vehicles sold in India [2]. According to Section 39 corresponds to Section 22 of the Motor Vehicles Act, 1939 [3], no individual has the privilege to drive vehicles on public places until or unless the vehicle is enroled with a registering

A. Banik (✉) · S. Chakraborty · A. Majumder
Tripura University, Suryamaninagar, Tripura(W) 799022, India
e-mail: banikanwesha00@gmail.com

P. P. Joby et al. (eds.), *IoT Based Control Networks and Intelligent Systems*, Lecture Notes in Networks and Systems 789, https://doi.org/10.1007/978-981-99-6586-1_2

authority. The owner of the vehicle needs to fill in an application to the Regional Transport Office (RTO) with the relevant documents. After the document verification phase, the RTO officials inspect the vehicle physically and after that, issue a registration certificate to the user. In that case, if the vehicle has resided for more than twelve months in a state where it is not registered, a no objection certificate needs to be issued from the registered state and an application for a new registration mark needs to be filed from the new state. The vehicle records are stored in a centralised database and are maintained by the State Transport Office. Each state maintains its own vehicle database. The existence of centralisation and opacity in the system may encourage legitimate users to inject false records or do ill-modification of data. There is a possibility of having some dishonest officials in the RTO who knowingly reject a true application during the vehicle inspection phase or in the document verification phase as the applicant does not satisfy the personal interests of the RTO officials. The misuse of privileges given to RTO officials for their job roles is a key factor in not being able to recover the stolen vehicle. The robbed vehicle becomes legal in the eyes of the police with a fake number plate. The stolen vehicle can be resold as the buyer will not be able to know the past history of the vehicle. The lack of a communication pathway between different state RTOs paved the way for making illegal vehicles in one state legal in another state. The whole vehicle registration system is paper-based. There may be a chance of falsification of documents at any step of the current vehicle registration system. The remaining part of the paper is organised as follows: Sect. 2 goes through the current vehicle registration system. Existing Techniques for Vehicle Registration Using Blockchain are described in Sect. 3. The proposed technique for new vehicle registration, vehicle enquiries, and interstate vehicle transfers is discussed in Sect. 4. Section 5 discusses how the proposed modules will be implemented. Section 6 summarises the assessment metric, and Sect. 7 wraps up the work and considers future directions.

2 Existing Vehicle Registration System

For vehicle owners in India, it is compulsory to register his/her own vehicle with the RTO to ply on Indian roads. An RTO has various responsibilities; some of the responsibilities are listed below:

1. The RTO collects road tax that an individual needs to pay while registering the vehicle for the very first time.
2. The RTO provides a VIN and a driving license, which are necessary documents for driving on the road.
3. RTO also issues permits for commercial vehicles.

Earlier, the management of vehicle data was based on files. Then, with the advancement of technology, the RTO became computerised. But, with the inflation in the sales of motor vehicles, there is an increase in the workload on the RTO. It is necessary for human–machine interface for the smooth functioning of the department.

For this reason, the Ministry of Road Transport and Highways (MoRTH) launched a "VAHAN" web portal [4] under the e-governance initiative, which provides a flexible interface for facilitating online RTO services like registration of vehicles, issuing driving licenses, etc. "Vahan" is a repository of digital vehicle records [5]. MoRTH also came up with a mobile application known as mParivahan, through which one can get instant access to vehicle information with just one click [6]. In the case of the offline mode of new vehicle registration, one needs to visit the nearest RTO office and fill in an application with necessary documents like valid identity proof, purchase invoice for the vehicle, copy of vehicle insurance, PAN Card copy, etc [7]. After proper verification and validation, a unique vehicle identification number is allotted to the owner and the record is stored in their own database [8].

As existing systems become more vulnerable to both internal and external threats, ensuring the integrity, confidentiality, and security of vehicle data becomes more difficult [9]. One of the principal reasons for the presence of malicious activities in an existing system is that records are stored in a database system [10]. The database system poses certain security threats which can expose confidential vehicle data to outsiders.

3 Existing Techniques for Vehicle Registration Using Blockchain

Some of the work in blockchain-based vehicle registration systems is summarised here. Hossain et al. [11] proposed blockchain-based vehicle registration system from Bangladesh perspective. Here, they considered each road transport authority (RTA) to act as a peer in the blockchain network. When an individual sends a registration request to RTA, their documents are verified, and if the documents are valid, they are converted to a hash digest using the SHA-256 algorithm. A hash digest is stored in the newly generated blocks, and it is propagated through the network among the peers. For a larger country like India, the proposed technique is inappropriate as it will cause scalability issues. Moreover, the work does not depict the imprisonment procedure.

Sanepara et al. [12] proposed a framework which begins with request of Chassis number from RTO. The RTO provides Chassis number to the manufacturer. Manufacturer produces vehicles with that Chassis number and assigns the vehicle to the dealer for sale. All transactions between manufacturer and RTO, manufacturer and dealer are recorded on the blockchain. The dealer sells the vehicle to a verified buyer and the transaction is recorded on the blockchain. Then, the buyer moves to the respective state RTO for registration of their vehicle. The proposed solution considers that all information about a particular vehicle is stored in a single block, and the chain will grow as new blocks are added. If a country has a large population and more than 50,000 vehicles are registered each day, the chain will grow exponentially, resulting in electricity waste and a delayed response rate.

Benarous et al. [13] has proposed a mechanism for registering vehicles. The manufacturer, certifying authority, customs, state, vehicle registering authority, and end users are the key entities in the proposed system. Each entity, excluding end users, maintains its own permissioned blockchain. Each blockchain has a set of validator nodes which have write-rights and others have only read-right. A manufacturer records fabricated vehicles, whereas a custom store imported vehicles on its blockchain. A certifying authority generates a pair of keys which are necessary to subscribe in the proposed vehicle registration system (BC-VRS). End users are the purchasers or sellers of vehicles who are authorised to confirm transactions or query the blockchain. The blockchains are connected together to form "Blockchains of Blockchain". The registration process begins with logging into BC-VRS with valid key pairs. After authentication, the user initiates a registration transaction to the state registering authority. The proposed technique does not have any provisions for those countries that have a large number of states and face issues during the transfer of vehicles from one state to another.

4 Proposed Scheme

The database system gives a simple and logical view of data. Because of its centralization, it attracts a variety of security threats that can expose vehicle data to an unauthorised persons. The abstraction property of a database invites legitimate privilege abuse. Hence, all these properties of the database are now proven to be inefficient for storing vehicle data. The RTOs are the registering authorities for vehicles in India. As a result, they have manipulation rights, which could open the door to false record injection. There is no collaboration between manufacturers and the RTO. For this reason, the process becomes complex as well as time-consuming. The RTO of a particular state acts as independent entities. Establishing the link between different state RTOs is necessary for hassle-free vehicle transfer from one state to another and to reduce the unethical activities related to this.

Hence, a framework is proposed based on blockchain technology, which aims to provide a solution to the issues present in the traditional system. Blockchain is chosen because vehicle data once stored, it cannot be manipulated illegitimately. Due to decentralisation property of blockchain, no one is sole proprietor and every transaction is transparent to each and every peer in blockchain. The proposed system is named the permissioned blockchain-based vehicle registration system (PB-VRS) system. The proposed framework considers that every state in a country maintains a blockchain for registering vehicles. Each state consists of district level RTOs. This district level RTO is configured to be peer in the consensus process. The automobile manufacturers of a country will participate as peers in the consensus process of the state blockchain but with no access to blockchain resources. Manufacturer peers are only validating peers, ensuring that the client transaction satisfies the endorsement policy and that the proposed operation on the channel is authorised.

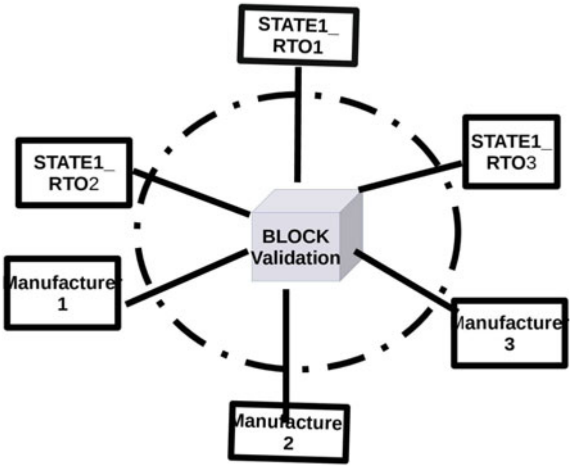


Fig. 1 Consensus process in PBCVRS

Figure 1 depicts the consensus process in PB-VRS system. RTO1, RTO2, and RTO3 are the district level RTOs of state "STATE1". Manufacturer1, Manufacturer2, and Manufacturer3 are the automobile manufacturers. Table 1 explains the meaning of notations used in the algorithms.

The PB-VRS system consists of following modules:

- New vehicle registration
- Query
- Interstate vehicle transfer.

4.1 New Vehicle Registration

Every state in India has their own RTO. The proposed framework considers that each RTO maintains its data on blockchain. The blockchain used by the state RTO is named as STATE-BC. The STATE-BC blockchain is chosen to be permissioned because it will decrease the overall latency and throughput as only authorised person will join the consensus process. The steps involved in registering a vehicle are listed below:

1. Certifying Authority: A certifying authority is a trusted third party who generates a pair of keys for a user.
2. Buyer: The buyer purchases a vehicle from a valid manufacturer. The buyer is provided with the transaction ID (MTX), engine number (EN), manufacturer ID (MID), owner name (Owner).
3. Manufacturer-RTO: Whenever a buyer purchases a vehicle from a manufacturer, the manufacturer initiates a transaction proposal to that RTO where the buyer wishes to enrol.

Table 1 Notation

SN	Notation	Meaning	SN	Notation	Meaning
1	VIN	Vehicle identification number inputted by the manufacturer	7	mid	Manufacturer Id inputted by buyer or by police
2	MID	Manufacturer identification number inputted by manufacturer	8	mtx	Transaction Id provided by buyer or by police
3	EN	Engine number inputted by manufacturer	9	owner	Owner name provided by buyer or by police
4	MTX	Special number which define the transaction between buyer and manufacturer	10	RTO	Regional Transport Office
5	Owner	Name of the buyer inputted by manufacturer	11	RTO_{old}	The state where user is registered
6	vin	Vehicle identification number provided by the user or by the police as input	12	RTO_{new}	The state where user wishes to register
13	PB-VRS	Permissioned blockchain-based vehicle registration system			

4. Smart-Contract Execution: After receiving the transaction proposal, the concerned RTO peers invokes smart contract Manufacturer() with the owner name, Manufacturer ID, Transaction ID, and engine number. The predefined logic in the smart contract validates whether the manufacturer is one of the registered automobile manufacturers or not. If the manufacturer is one of the registered automobile IDs, then the “user-confirmed” attribute of the ledger is set to zero. The ledger consists of a “user-confirmed” attribute, which is set to 0 when the

buyer of the vehicle does not initiate any request to add it, and it is updated to 1 when the user confirms that he is the owner of the vehicle. The transaction, along with the endorsing peer's signature, is broadcasted to all members, i.e. all district level RTOs and all manufacturers. If the transaction satisfies the endorsement policy, it is added to the ledger. Algorithm 1 explains the smart contract.

5. User-RTO: An application user subscribes to the PB-VRS system with valid key credentials and sends a transaction to the RTO. The endorsing peer authenticates the transaction and invokes the smart contract CreateVehicle() with the given parameters. The smart contracts check whether the input parameters like mid, mtx, en, owner match with the parameters provided by the manufacturer. If there is correspondence between the user's inputted data and the data stored in the ledger, then the "Confirmed-user" field is set to 1. And the user is provided with a VIN. Algorithm 2 explains the smart contract for CreateVehicle(). This module eliminates false injection of record as transaction for adding new vehicle are transparent to manufacturer, RTO, and user. With this framework, RTO don't remain centralised authority. Flow chart for new vehicle registration is shown in Fig. 2.

Algorithm 1: Smart contract For Manufacturer()

```
Data: MID,MTX,Owner,EN
Result: User is registered.
Lat A be the array of registered automobile manufacturer.
i=0
while i < n do
    if A[i] == MID then
        "MID is valid" submit(MID,MTX,Owner,EN);
    else
        "MID is Invalid"
    end
    i=i+1
end
```

4.2 Query

The "Query" module avails user to retrieves information from respective state RTO blockchain. The smart contract consists of QueryVehicle(), which define set of rules for querying the blockchain. The user inputs Vehicle Identification Number(vin), Manufacturer Transaction ID(mid). If the set of predefined condition in QueryVehicle() is satisfied with the user input, the corresponding information is shown to the user. Algorithm 3 depicts smart contract for Query (Fig. 3).

4.3 Interstate Vehicle Transfer

The interstate transfer module provides a blockchain-based framework for transferring vehicles from one state to another. This module promises to prevent the

Algorithm 2: Smart contract for CreateVehicle()

```
Data: mid,mtx,owner,en
Result: User is registered.
Let n be the number of records present in the ledger.
i=1
while i != n do
    if world state[i] == mtx then
        car=getstate(mtx) * Value of mtx key is assigned to car variable */
        if car.MID == mid then
            if car.Owner == owner then
                if car.En == en then
                    car.confirmed_user=1
                    break;
                else
                    "Engine number doesn't matched"
                end
            else
                "Owner name mismatched"
            end
        else
            "Invalid Manufacturer ID"
        end
    else
        "Wrong Transaction Id" ;
    end
    i=i+1 */increment the value of I */
end
```

Algorithm 3: Smart contract for QueryVehicle

```
Data: mid,vin
Result: record is displayed
i=1
while i=-1 do
    if world state[i] == vin then
        car=getstate(vin) * Value of vin key is assiagned to car */
        i=0
        if car.MID == mid then
            print("Show the information")
        else
            Invalid mid
        end
    else
        Invalid vin ;
    end
    i=i+1 */increment the value of I */
end
```

re-registration of any stolen or smuggled vehicle. The proposed framework makes the transfer of vehicles less complex. The state RTOs, automobile manufacturers are the endorsing peers. On considering the security and privacy of public vehicle data, manufacturers are not permitted to use ledger data. They cannot query or write data. Following steps involved in this module:

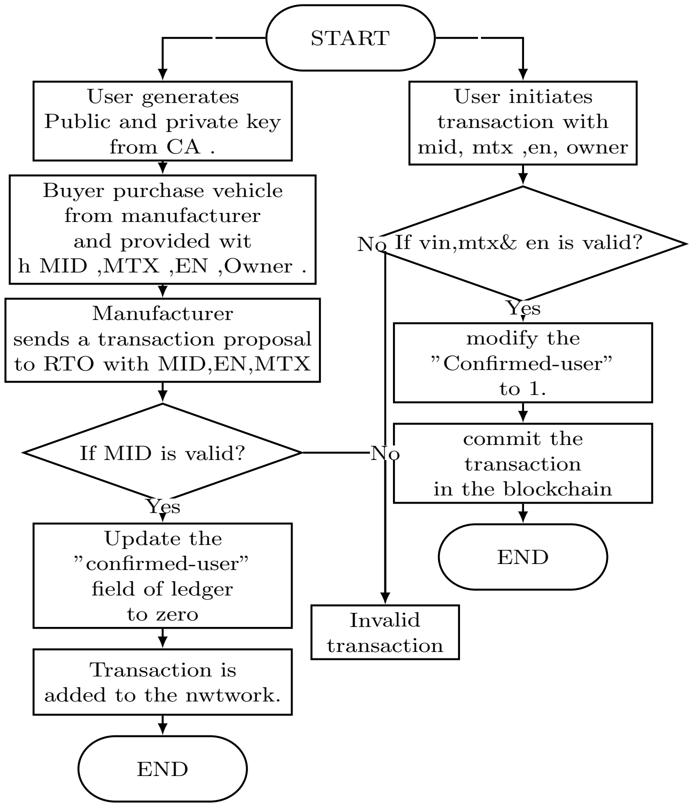


Fig. 2 Flow chart for new vehicle registration in PB-VRS

Algorithm 4: Smart contract for Revocation

```
Data: mid,vin
Result: Revok field of ledger is updated.
i=0
while i=-1 do
    if world state[i] == vin then
        car=getstate(vin) * Value of VIN key is assiagned to car */
        i=0
        if car.mid == MID then
            car.revok=1
        else
            Invalid Manufacturer Id
        end
    else
        Invalid VIN ;
    end
    i=i+1 */increment the value of I */
end
```

1. **Police**: The police is a unit that initiates a transaction in the PB-VRS system when there is any involvement of motor vehicles in criminal activity.

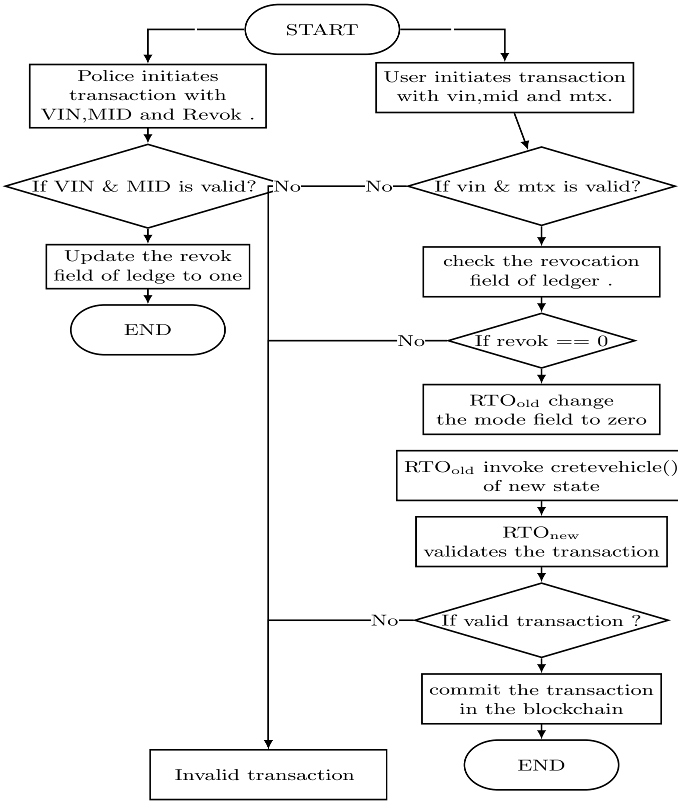
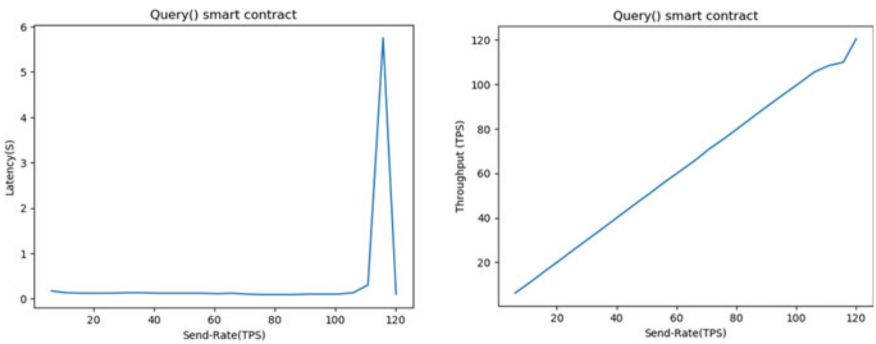


Fig. 3 Flow chart for Query module in PB-VRS

2. **Police-RTO**: Police as a client in the blockchain network invokes Revocation() with the VIN, MID, and revoke number. A revocation number is a field in a blockchain ledger that specifies whither the vehicle is involved in criminal activity or not. Revoke number is set to '1' means the vehicle owner has criminal records, while revok number '0' means the vehicle owner has no criminal records. If MID and VIN exist for the owner in the ledger, then the revoke number will be updated to 1. Algorithm 4 explains the revokation() smart contract.
3. **User**: The user is the one who wishes to transfer a vehicle from one state to another. Endorsing peers must grant the user request to use the smart contract TransferVehicle().
4. **Smart contract execution**: Algorithm 5 explains the smart contract for interstatetransfer(). The algorithm checks whether the entered parameters like mid, mtx, en, and owner correspond with the data present in the ledger or not. If the inputted data is valid than, the revocation field is checked. If the vehicle is not involved in criminal activity, then the mode bit is changed to 1. The mode bit in the ledger describes whether the vehicle is transferred or not. Mode bit 1 indi-

Algorithm 5: Smart contract for Interstate Vehicle Transfer

```
Data: mid,mtx,en,vin,mode,owner
Result: record is stored in new state RTO blockchain
i=0
while i=-1 do
    if world state[i] == vin then
        car=getstate(vin) * Value of vin key is assiagned to car */
        i=0
        if car.MTX == mtx and car.EN==en and car.mode== 0 then
            if car.Owner== owner then
                if car.revok==0 then
                    print" Transaction accepted"  car.mode=1
                else
                    "Owner involved in crimnal activity
                end
            else
                " Invalid Owner name "
            end
        else
            Invalid Manufacturer Transaction Id or Engine number
        end
    else
        Invalid VIN ;
    end
    i=i+1 */increment the value of I */
end
```



(a) Send-rate v/s latency (b) Send-rate v/s Theoughput

Fig. 4 Flow chart for interstate transfer module in PB-VRS

cates that the vehicle has been transferred, whereas mode bit 0 indicates that the vehicle is still in the place of registration.

5. **RTO_{old} to RTO_{new}**: RTO_{old} is the state RTO where the user is registered. RTO_{new} is the state RTO where the user wishes to register itself. RTO_{old} on behalf of user, invokes CreateVehicle() module of RTO_{New} with user credentials like en, mid, mtx, owner.
6. **RTO_{new}**: RTO_{new} verifies the transaction and commit to the blockchain. Figure 4 explains the interstate vehicle transfer.

5 Implementation

The proposed framework has been implemented using Hyperledger Fabric. Hyperledger Fabric is one of the projects under hyperledger, which provides an enterprise-based blockchain solution [14]. As Hyperledger Fabric is permissioned, only known identities can join the consensus. It also has channel functionality that allows different members to build their own ledgers of transactions. It also includes an endorsement policy, which allows you to specify which peers have the authority to execute smart contracts and to approve transactions. The endorsement policies are set during the chincode definition and are approved by the Chanel members. In Hyperledger Fabric, it consists of two parts: the world state and the blockchain. The world state is a database that holds the current value of a ledger state, which in turn makes querying and updating the ledger programmatically easier. World states are stored as a key-value pair. The fabric-contract-api provides users with a high-level interface for developing smart contracts. PutState(), getState(), and getTxID() operations are provided by ctx.stub, which are required for transaction processing. In this proposed framework, we populated the ledger with initial data, which consists of the following attributes: vehicle identification number, engine number, manufacturer ID, manufacturer transaction ID, owner name, revok, user-confirmed, and mode. The proposed modules are implemented by taking into account five state RTOs, each with two district level RTOs and three manufacturers. The smart contract is written in the GO programming language and includes the following functions: Manufacturer(), CreateVehicle(), Revocation(), Transfer(), and Query().

6 Result

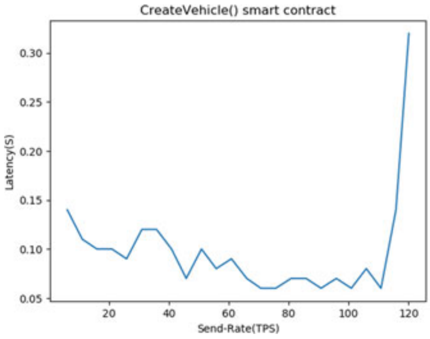
Hyperledger caliper is used for evaluating the performance of modules [15]. Hyperledger caliper is a blockchain benchmark tool. It allows users to measure the performance of a blockchain implementation against a set of predefined use cases. Transaction send rate, transaction throughput, transaction latency, success, or failure are the parameters of hyperledger caliper.

6.1 Performance Evaluation of Query Smart Contract

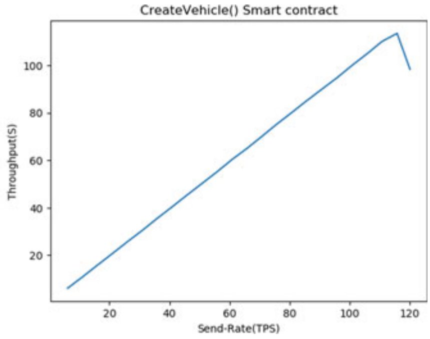
For evaluating the Query module, the benchmark is configured to have 24 rounds of testing. At each round, the transaction per second and transaction duration were set to be multiples of 50 and 5, respectively. The rounds are set to have a multiple of 50 transactions per second in order to visualise the behaviour of the model with an increasing rate of transactions per second. A summary of the benchmark caliper is depicted in Table 2.

Table 2 Performance evaluation metric for Query()

Name	Succ	Fail	Send rate (TPS)	Latency (S)	Throughput (TPS)	Name	Succ	Fail	Send rate	Latency	Throughput
Round1	30	20	6.2	0.17	6.2	Round13	390	260	65.8	0.12	656
Round2	60	40	11.1	0.13	11.0	Round14	420	280	70.9	0.10	7
Round3	90	60	16.0	0.12	16.0	Round15	450	300	75.7	0.09	75
Round4	120	80	21.0	0.12	20.9	Round16	480	320	80.8	0.09	80
Round5	150	100	26.0	0.12	25.9	Round17	510	340	85.8	0.09	85
Round6	180	120	31.0	0.13	30.9	Round18	540	360	90.8	0.10	90.6
Round7	210	140	35.9	0.13	35.8	Round19	570	380	95.8	0.10	95
Round8	240	160	40.9	0.12	40.8	Round20	600	400	100.5	0.10	100.4
Round9	270	180	45.9	0.12	45.8	Round21	630	420	105.8	0.13	105.4
Round10	300	200	51.0	0.12	50.8	Round22	660	440	110.8	0.30	108.4
Round11	330	220	55.9	0.12	55.8	Round23	690	460	115.5	5.75	42.8
Round12	360	240	60.8	0.11	60.7	Round24	720	480	120.9	0.10	120.5



(a) Send-rate v/s latency



(b) Send-rate v/s Theoughput

Fig. 5 Performance evaluation of Transfer()

Figure 5a, b depict the relationship between the send rate-throughput and send rate-latency, respectively, for the query module. Figure 5a shows how latency remains constant for a certain number of transactions before increasing abruptly. The latency is minimal up to a send rate of 100 transactions per second. The graph in Fig. 5b shows that as the send rate increases, so does throughput. Throughput depends on the number of transactions that satisfy the endorsement policy and the number of transactions committed to the blockchain network. For evaluating, success or failure of transaction is in the ration of 3:2.

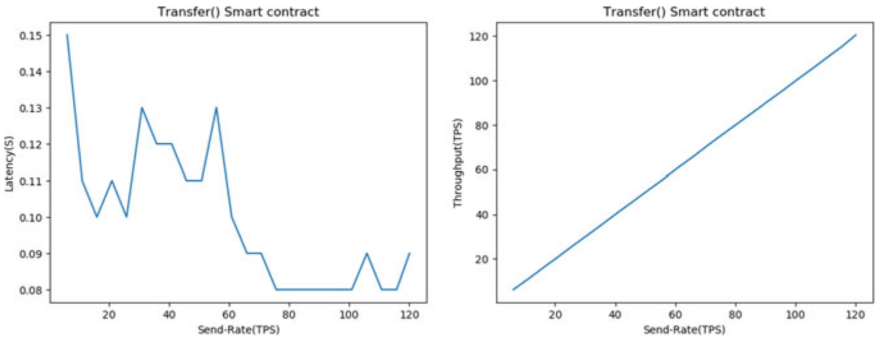
6.2 Performance Evaluation of CreateVehicle() Smart Contract

For evaluating the CreateVehicle() module, the benchmark is configured to have 24 rounds of testing. At each round, the transaction rate per second and transaction duration were set to be multiples of 50 and 5, respectively. The rounds are set to have a multiple of 50 transactions per second in order to visualise the behaviour of the model with an increasing rate of transactions per second. The number of transactions satisfied by the smart contract determines whether transaction succeeds or fails. Summary of benchmark caliper is depicted on Table 3.

Figure 6a, b shows relationship between send rate and throughput, send rate and latency respectively for CreateVehicle module. The graph of Fig. 6a depicts that latency is minimal up to 110 transaction per second. The graph of Fig. 6b suggests that throughput is proportional to send rate up to 116 transaction per second.

Table 3 Performance evaluation metric for CreateVehicle()

Name	Succ	Fail	Send rate (TPS)	Latency (S)	Throughput (TPS)	Name	Succ	Fail	Send rate	Latency	Throughput
Round1	20	30	6.2	0.14	6.1	Round15	260	390	66.0	0.07	65
Round2	40	60	11.1	0.11	10.9	Round14	280	420	70.7	0.10	70
Round3	60	90	16.0	0.10	15.	Round15	300	450	75.7	0.08	75
Round4	80	120	21.0	0.10	20.7	Round16	320	480	80.8	0.09	80.2
Round5	100	150	25.9	0.09	25.6	Round17	340	510	85.8	0.07	85.2
Round6	120	180	31.0	0.12	30.5	Round18	360	540	90.8	0.06	90.0
Round7	140	210	35.9	0.12	35.5	Round19	380	570	95.9	0.07	94.9
Round8	160	240	40.9	0.10	40.4	Round20	400	600	100.9	0.07	100.1
Round9	180	270	45.8	0.07	45.3	Round21	420	630	105.9	0.08	105.0
Round10	200	300	50.9	0.10	50.3	Round22	440	660	110.8	0.06	110.1
Round11	220	330	55.8	0.08	55.1	Round23	460	690	115.8	0.14	113.5
Round12	240	360	60.9	0.09	60.	Round24	479	721	120.1	0.32	98.4



(a) Send-rate v/s latency (b) Send-rate v/s Theoughput

Fig. 6 Performance evaluation of CreateVehicle()

6.3 Performance Evaluation of Transfer(): Smart Contract

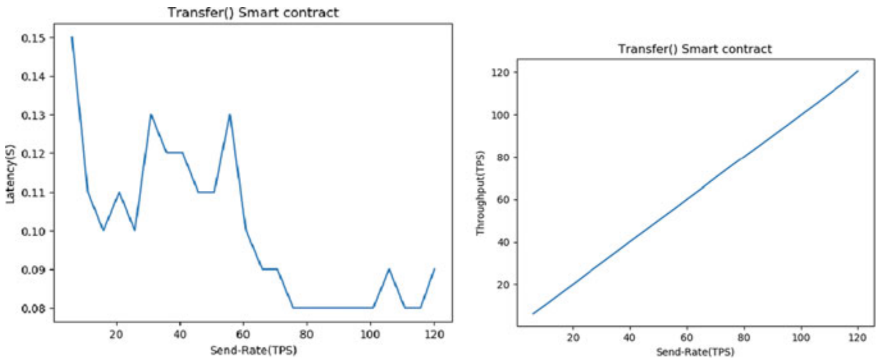
For evaluating the smart contract transfer() module, the benchmark is configured to have 24 rounds of testing, where each round has a send rate of a multiple of 5. The number of transaction in each benchmark round is configured to be multiple of 50. Summary of benchmark caliper is depicted on Table 4. Figure 7a shows relationship between send rate and latency. The graph depicts that latency remain minimal when the send rate is between 80 and 100 transactions per second. Figure 7b derives the interrelation between send rate and throughput. The graph depicts that throughput increases with the increase of send rate.

7 Conclusion and Future Work

The work presented here modelled the vehicle data management system using permissioned blockchain. The proposed scheme consists of three modules: Query, new vehicle registration, and interstate vehicle transfer. Manufacturers and all district level RTOs are the endorsing peers. The work has been implemented using Hyperledger Fabric, and each module is evaluated through benchmark caliper. The Hyperledger Fabric is chosen because it is a permissioned blockchain and has an endorsement policy, which is necessary for specifying read-write access for peers. The modules are evaluated on the basis of throughput, latency, and send rate. These modules bring district level RTOs and manufacturers on one platform. Hence, there is no way of registering false vehicle records as well as re-registration of stolen vehicles. The work can be further extended by incorporating other entities like insurance, driving license generation, etc. The work can be modified by including high-end client authentication mechanisms and a way of verifying who is joining the consensus.

Table 4 Performance evaluation metric for Transfer()

Name	Succ	Fail	Send rate (TPS)	Latency (S)	Throughput (TPS)	Name	Succ	Fail	Send rate	Latency	Throughput
Round1	20	30	6.2	0.15	6.2	Round13	260	390	65.9	0.09	65.8
Round2	40	60	11.1	0.11	11.0 7	Round14	280	420	70.9	0.09	70.7
Round3	60	90	16.0	0.10	16.0	Round15	300	450	75.8	0.08	75.7
Round4	80	120	21.0	0.11	20.9	Round16	320	480	80.9	0.08	80.7
Round5	100	150	25.9	0.10	25.9	Round17	340	510	85.7	0.08	85.6
Round6	120	180	31.0	0.13	30.9	Round18	360	540	90.8	0.08	90.6
Round7	140	210	36.0	0.12	35.8	Round19	380	570	95.8	0.08	95.6
Round8	160	240	41.0	0.12	40.9	Round20	400	600	100.8	0.08	100.7
Round9	180	270	45.9	0.11	45.7	Round21	420	630	105.7	0.09	105.6
Round10	200	300	50.8	0.11	50.8	Round22	440	660	110.8	0.08	110.6
Round11	220	330	55.8	0.13	55.6	Round23	460	690	115.8	0.08	115.6
Round12	240	360	60.9	0.10	60.8 7	Round24	480	720	120.8	0.09	120.5



(a) Send-rate v/s latency (b) Send-rate v/s Theoughput

Fig. 7 Performance evaluation of Transfer()

References

1. Singh R, Malik A (2022) Auto industry pins hope on consumption-led demand in budget for revival of growth, Jan 2022
2. Statista Research Department (2022) Motor vehicle sales volume in India from 2005 to 2021, June 2022
3. Dhiman V (2017) Epilepsy and law in India. Neurol India 65(5):1201–1201
4. Katoch R (2016) E-governance: government to citizen (g2c) initiatives in India. Int J Res Soc Sci 6(10):225–244
5. Syed TA, Alzahrani A, Jan S, Siddiqui MS, Nadeem A, Alghamdi T (2019) A comparative analysis of blockchain architecture and its applications: problems and recommendations. IEEE Access 7:176838–176869
6. Rogaway P, Shrimpton T (2004) Crypto-graphic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. In: International workshop on fast software encryption. Springer, Berlin, pp 371–388
7. Jirwan N, Singh A, Vijay S (2013) Review and analysis of cryptography techniques. Int J Sci Eng Res 4(3):1–6
8. Bach LM, Mihaljevic B, Zagar M (2018) Comparative analysis of blockchain consensus algorithms. In: 2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO). IEEE, pp 1545–1550
9. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system. Decentralized business review, p 21260
10. Nguyen CT, Hoang DT, Nguyen DN, Niyato D, Nguyen HT, Dutkiewicz E (2019) Proof-of-stake consensus mechanisms for future blockchain networks: fundamentals, applications and opportunities. IEEE Access 7:85727–85745
11. Hossain MdP, Khaled Md, Saju SA, Roy S, Biswas M, Rahaman MA (2020) Vehicle registration and in- formation management using blockchain based distributed ledger from Bangladesh perspective. In: 2020 IEEE region 10 symposium (TENSYMP). IEEE, pp 900–903
12. Khokhariya S, Shah J, Desai K, Chaudhari H, Chaudhary J, Sanepara V, Savani D (2020) Complete vehicle registration process using blockchain technology. Int Res J Eng Technol (IRJET)
13. Benarous L, Kadri B, Bouridane A, Benkhelifa E (2021) Blockchain-based forgery resilient vehicle registration system. Trans Emerg Telecommun Technol e4237

14. Baliga A, Solanki N, Verekar S, Pednekar A, Kamat P, Chatterjee S (2018) Performance characterization of hyper-ledger fabric. In: 2018 Crypto valley conference on blockchain technology (CVCBT). IEEE, pp 65–74
15. Choi W, Hong JW-K (2021) Performance evaluation of Ethereum private and testnet networks using hyperledger caliper. In: 2021 22nd Asia-Pacific network operations and management symposium (APNOMS). IEEE, pp 325–329