Securing Connected Vehicle Applications with an Efficient Dual Cyber-Physical Blockchain Framework

Xiangguo Liu[1], Baiting Luo[1], Ahmed Abdo[2], Nael Abu-Ghazaleh[2], Qi Zhu[1]

***Abstract*— While connected vehicle (CV) applications have the potential to revolutionize traditional transportation system, cyber and physical attacks on them may lead to disastrous consequences. In this work, we propose an efficient dual cyber-physical blockchain framework to build trust and secure communication for CV applications. Our approach incorporates blockchain technology and physical sensing capabilities of vehicles to quickly react to attacks in a large-scale vehicular network, with low resource overhead. We explore the application of our framework to three CV applications, i.e., highway merging, intelligent intersection management, and traffic network with route choices. Simulation results demonstrate the effectiveness of our blockchain-based framework in defending against spoofing attacks, bad mouthing attacks, and Sybil and voting attacks. We also provide analysis to show the timing and resource efficiency of our framework.**

I. Introduction

Connected vehicle (CV) applications are expected to revolutionize traditional transportation system. The connected vehicles exchange messages with each other and surrounding infrastructure units to extend perception range, exchange traffic status in downstream, and coordinate planning and control actions to improve transportation safety, efficiency, and mobility. The U.S. Department of Transportation (USDOT) has identified a number of promising CV applications and started deploying them at test sites in Florida, New York, and Wyoming [1]. Many other academic and industry testbeds and deployments are also under the way [2]. However, cyber and physical attacks targeting these systems can have severe safety implications, causing accidents or disrupting traffic flow. For example, an intelligent intersection management system can experience deadlock if messages have a long transmission delay or get lost under the denial of service attack [3], [4]. Merging in the highway will be more prone to accidents if vehicles get the wrong position values of the surrounding vehicles under spoofing and false message attacks [5]. As the impact of cyber threats [6] on CV operations can be so destructive, it is essential to develop security solutions against them.

A central issue in CV security is to build *trust* among vehicles. The authors in [7] reviewed methods to evaluate message trustworthiness in the vehicular network, including entity-oriented, data-centric, and collaborative trust models. The entity-oriented trust models evaluate messages' trustworthiness based on the trustworthiness of their senders. A Certification Authority is often leveraged to record vehicles' behavior and provide estimations of trust; otherwise, a vehicle needs to collect information and make evaluation by itself. The data-centric trust models evaluate the content of messages. In this category, Bayesian inference and Dempster-Shafer theory are popular methods, however they may result in false positives [8]. The collaborative trust models are based on integrating trust estimates from other peer vehicles, which could be time-consuming and attacked by malicious players. The Security Credential Management System (SCMS) is a proof-of-concept message security solution that is supported and developed by USDOT. Instead of evaluating message trustworthiness by each vehicle, message senders with credentials can be trusted in the system. However, since a certified vehicle can be attacked later, it is challenging for certificate authorities to track and update vehicles' status quickly. Thus, SCMS on its own does not prevent application level attacks [5].

Blockchain [9] technology has natural strength in recording transactions/events and reaching a secure consensus among all users. It provides a promising direction for building trust in CV applications, as shown in [10], [11], [12]. However, the required secure consensus operations in those earlier works introduce high overhead that makes it difficult

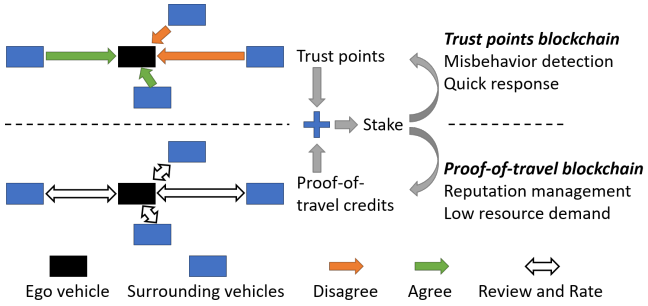

Fig. 1. Overview of our proposed dual cyber-physical blockchain framework. The ego vehicle is traveling and sharing traffic information with surrounding vehicles. In the trust points blockchain, surrounding vehicles leverage their physical sensing capability to verify the messages sent from the ego vehicle. If a falsified message is detected and reported, trust points of relevant vehicles will be adjusted. In the proof-of-travel blockchain, ego vehicle and surrounding vehicles record the number of received messages from different vehicles, which can reflect the travel activities and their contributions. The stake of each vehicle is computed from both its trust points and proof-of-travel credits. Our dual blockchain design is secure as long as more than 2/3 of the stake is held by honest vehicles.

[1]Xiangguo Liu, Baiting Luo and Qi Zhu are with the Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL 60201, USA. xg.liu@u.northwestern.edu, baitingluo2019@u.northwestern.edu, qzhu@northwestern.edu.

[2]Ahmed Abdo and Nael Abu-Ghazaleh are with the Department of Electrical and Computer Engineering, University of California, Riverside, CA 92521, USA. aabdo003@ucr.edu, nael@cs.ucr.edu.

for applying them in practical CV applications.

In this work, we present an efficient dual cyber-physical blockchain framework to build trust and secure communication for CV applications. The framework enables us to efficiently track and update a trust estimate for each vehicle in a large-scale traffic network, with low resource overhead. The design of our framework is shown in Fig. 1. It builds and updates two blockchains for recording vehicles' communication activities and establishing trust among vehicles. The first blockchain, named *trust points blockchain*, is used to quickly identify and record malicious misbehavior. Intuitively, telling the truth and getting acknowledged by most neighbors will earn trust points, while telling a lie will lose trust points. The second blockchain, named *proof-of-travel blockchain*, accumulates and records each vehicle's long-term contribution to the CV community. Intuitively, the more traffic information it shares with others, the more contribution to the CV community it gets and the higher proof-of-travel credits it can receive from others. This blockchain serves as a low-cost conceptually-centralized tracker of vehicle trust, enabling two vehicles to establish trust when they do not have extended experience with each other, in a way that is secure and dependent on consensus.

Our proposed framework makes it difficult to launch attacks, and facilitates quick detection and reaction to misbehavior. Specifically, once a suspicious message is reported, surrounding vehicles can adapt to more cautious and conservative actions within just hundreds of milliseconds. Within one minute, surrounding vehicles can leverage their sensing capabilities to verify the message and reach consensus on it. Then, the trust points of the vehicle that sent the suspicious message will get updated in the trust points blockchain. On the other hand, as travel history is recorded in the proof-of-travel blockchain, vehicles with poor travel history cannot easily start certain attacks such as Sybil attack or flooding attack (or other attacks that require repeated transmission).

The technical contribution of our work can be summarized as follows:

- We developed a novel dual blockchain framework that leverages cyber security techniques, physical sensing capabilities of vehicles, and their travel histories to build trust and secure communication in a large-scale vehicular network, in which every vehicle can get updates timely with low resource overhead.
- We designed a stake-based consensus mechanism across the faster trust points blockchain and the slower proof-of-travel blockchain, and a sharding technique for partitioning vehicles into regions, to reduce computation, communication and storage costs of our framework so that it is practical for CV applications.
- We demonstrated the effectiveness of our framework against a few prevalent attacks, including message spoofing attack, bad mouthing attack, and Sybil attack. We illustrated the performance and timing efficiency of our defense system via simulations in the SUMO [13] simulator, and analyzed its resource overhead and trade-offs.

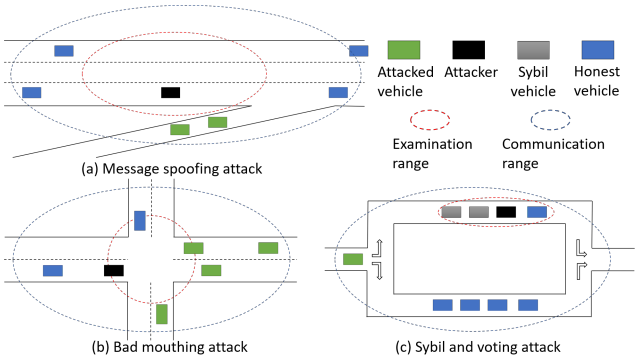

Fig. 2. Threat models considered in our work. Subplot (a) displays that an attacker sends falsified messages with wrong traffic status on the highway to mislead the green merging vehicles. Subplot (b) displays that an attacker forges misbehavior of surrounding honest vehicles to hurt their reputation and traffic efficiency in the intelligent intersection. Subplot (c) displays that an attacker forges two pseudonymous identities by launching a Sybil attack. Together they report falsified traffic accidents in one of the routes, and may dominate the voice in voting among the surrounding vehicles. This attack may lead to congestion because more honest vehicles will choose the other route that is without the fake accident.

The rest of the paper is organized as follows. Section II introduces the security threat models we address in this work, related background on blockchain, and previous work in applying blockchain to transportation systems. Section III presents the design of our dual blockchain framework. Section IV analyzes our framework's defense performance against message spoofing attack, bad mouthing attack, and Sybil and voting attack, and presents the simulation results in SUMO. Section V concludes the paper.

II. Background and Related Work

A. Threat Models to Connected Vehicles

In this work, we focus on the attacks from vehicle-side devices, e.g., those from the On-Board Units (OBUs) for CV applications [3]. While the techniques can be extended to protect against attacks on infrastructure units, it is reasonable to expect that it is typically harder to attack the infrastructure. In particular, we consider malicious vehicles that can generate falsified messages and broadcast them to other vehicles. It is important to note that we do not assume that the attacker can spoof the sender identities in the messages. We assume that such an identity is verifiable and non-forgeable with digital signatures techniques (e.g., SCMS). Moreover, we assume that the communication infrastructure is mostly resilient and most surrounding vehicles within a communication range should receive the same message within a time bound. We also assume that most honest vehicles have loosely synchronized clocks (e.g., using Network Time Protocol (NTP)) for the liveness of blockchain. Finally, we assume that most vehicles in the traffic network are honest, such that most of the stake (more than 2/3) can be held by honest vehicles to ensure the safety of blockchain.

Under these assumptions, malicious attackers can compromise a small fraction of vehicles to broadcast falsified position and velocity data, forge traffic accidents and destroy

others' reputation. Specifically, we consider three threat models in this paper with popular and representative CV applications, as shown in Fig. 2. Among them, the message spoofing attack could target many other CV applications and be defended by trust frameworks such as ours. The bad mouthing attack and Sybil and voting attack in fact target the trust framework itself, and as shown later, our blockchain framework can effectively mitigate such attacks as well.

1) Message spoofing attack: In Fig. 2(a), two green vehicles communicate with the vehicles on the highway to adjust their speed for merging onto the highway. The attacker (shown in black) can send out falsified position and velocity data of itself, which may induce the green victim vehicles to accelerate/decelerate. This may damage traffic efficiency and even put vehicles near the on-ramp at the risk of collisions.

2) Bad mouthing attack: In Fig. 2(b), all vehicles send their current and destination lanes as well as estimated arriving time to a centralized intersection manager, which then decides the passing order and timing of the vehicles [4]. When a trust framework is deployed to secure the communication, an attacker may maliciously report misbehavior to degrade the trust for honest vehicles [14]. In this case, the intersection manager cannot decide the passing order and timing for vehicles, and they will have to adopt the traditional traffic rules as if there were stop signs in every direction.

3) Sybil and voting attack: In Fig. 2(c), there are two route choices in the region. An attacker can generate fake and pseudonymous vehicles by Sybil attack. Together they can broadcast a falsified traffic accident in one of the routes. In a trust framework based on consensus, while honest vehicles may disagree with the falsified accident, their voices could be dominated by the attacker and Sybil vehicles in the voting process for assessing the falsified accident, in which case, other vehicles coming to this region will believe the falsified accident and go through the other route, resulting in congestion and low traffic efficiency.

B. Blockchain and Algorand

Blockchain consists of a chain of blocks (called a ledger), in which a block stores a set of transactions and the hash value of the previous block. Transactions are public to every user, and thus blocks can be verified by every user. Blockchain is updated based on consensus algorithms to guarantee that the state of the ledger is identical at all nodes. The most well-known blockchain, bitcoin, is based on proof-of-work. It assumes that a group of miners will build up new blocks by consuming computational resources to find the right cryptographic nonce. Once a new block is proposed by a miner, other miners will verify it and start working on the next block. Because the block generation process is not deterministic, a hacker may successfully build a new block with a slight possibility. In this way, a block is considered to be confirmed if it is followed by several other blocks. For bitcoin, the block generating period is about 10 minutes, and the confirmation time can be 1 hour.

Algorand [15] is a recently proposed blockchain design that enables efficient block confirmation time. It is based on proof-of-stake, where users do not need to exhaust computation resources. Instead, users need to deposit money as the stake. The more money a user owns, the higher chance the user can be selected as a block proposer or verifier. Its security is guaranteed theoretically as long as the majority of stake is owned by honest users. Furthermore, its scalability is demonstrated with simulations of up to 500,000 users, in which transaction confirmation time can be shortened to be within one minute [16], making it feasible for use in some systems at real time.

Our framework leverages Algorand, combining its protocol design with our ideas of designing dual cyber-physical blockchains, using vehicles' physical sensing capabilities for verifying messages, and introducing the concept of proof-of-travel credits in stake computation. Moreover, to scale Algorand to CV applications in a large traffic network, we leverage the sharding technique [9] for record transferring and sharing between blockchains in different regions. By choosing an appropriate region size, our framework provides short round latency and low resource overhead, while preventing vehicles from transferring records too frequently.

C. Blockchain in Transportation

Blockchain has been leveraged by researchers in the literature to advance applications in transportation systems. For instance, [17] presents a blockchain based negotiation process to select the most convenient electric vehicle (EV) charging station. [18] leverages blockchain to record EV charging activities, which enables information sharing while securing sensitive user information. [19] presents a seven-layer conceptual model for blockchain in transportation and discusses ride sharing in the application layer.

There are also several works that consider using blockchain in vehicular networks. [10] reviews different models for calculating reputations of vehicles, discusses challenges of previously centralized and distributed methods, and indicates new directions on blockchain and fog computing. [11] proposes to secure vehicular communication by recording each message in proof-of-work based blockchain, which is not scalable. [12] proposes a blockchain based trust management scheme for vehicular networks, in which road side units (RSUs) update and maintain the blockchain. Its consensus is partially based on proof-of-work, which still leads to high computation cost, and its security in one region cannot be well protected once the RSU is compromised. [20] and [21] also suffer from high computation overhead and the failure of one node. [22] attempts to mitigate the resource demands by proposing concepts of sub-blockchains. However, it lacks implementation details such as communication across different sub-blockchains.

Our work addresses the challenge in applying blockchain to CV applications with a novel framework design that is introduced below in details.

III. Design of Our Framework

In our framework design, there are two types of information being exchanged among vehicles and infrastructure

units – *messages* and *transactions*. Messages carry traffic information (position, velocity, etc.) that vehicles share with others, e.g., Basic Safety Messages (BSMs). Transactions are generated by vehicles and broadcast to other vehicles in the same region to build trust via blockchain. Transactions generally include the report of malicious misbehavior, application of transferring records, etc. By aggregating all transactions within a period, one vehicle can then update all vehicles' records and build a block accordingly.

A. Framework Overview

Our blockchain design leverages the block generation and consensus mechanism from Algorand [15]. Technically speaking, it is based on proof-of-stake. In our design, the stake is computed from trust points and proof-of-travel credits. The higher stake a vehicle holds, the higher chance it can be selected as a leader or verifier to maintain the blockchains. When driving, only traffic information sent from vehicles with high trust points and proof-of-travel credits can be viewed as trustworthy.

To update blockchain on time, and alleviate resource demand for vehicles, we leverage the sharding method [9] to partition vehicles into subsets according to their traveling region. Each vehicle has a permanent address and a current active address. For trust points blockchain, attackers' misbehavior should be exposed as soon as possible, and surrounding vehicles in the current active region can respond within a short time. Thus, trust points blockchain in one region is maintained by those vehicles that claim to be active in the current region and will only record those vehicles' trust points. A vehicle that moves across different regions can claim its new active region and have its trust points record transferred. The detailed mechanism is introduced in Section III-B. For proof-of-travel blockchain, since it records a vehicle's historical information over a long period (e.g., 100 days), it is maintained by those vehicles that have their permanent address in this region. It will only record those vehicles' proof-of-travel credits. In this way, different regions will have their blockchains maintained and updated independently.

Other details, such as the transaction generation and physical verification processes for the two blockchains, are introduced in the following Sections III-B and III-C, respectively.

B. Trust Points Blockchain

Trust points blockchain is mainly for identifying and exposing malicious attackers. When one vehicle sends out a BSM message, surrounding vehicles within the communication range can receive and verify it based on information from their own on-board sensors or other sources (e.g., a message with falsified position data may be deemed as suspicious by surrounding vehicles using their own sensors[1]). Then depending on the situation, various types of transactions can be generated and sent. Each transaction includes transaction ID (TID), smart contract type ID (SCID), senderID, debateID, regionID, location, time and payload. Transaction senders will encrypt the transactions with their private keys and broadcast them. Receivers can verify the identity of senders with the public keys. SCID identifies the transaction type and the corresponding smart contract for updating vehicles' points/credits. Location and time are the transaction generation location and time. DebateID is the ID of the vehicle that sends out the suspicious message. RegionID is the ID of the region that a vehicle is moving into. TID is the hash value of senderID, debateID, regionID, location and time, and is considered unique. Various transactions are sent in the following situations:

- If a vehicle disagrees with a message sent by another vehicle, it can report this disagreement in a transaction with SCID=0000.
- Upon receiving a disagreement transaction (SCID=0000), other vehicles can take a stand on agreeing or disagreeing in a new transaction with SCID=0000 if they have not done so.
- If a vehicle disagrees with the judgement made in the previous voting process (details of the process are introduced later in Section III-B.1), it can report the disagreement after enough evidence is gathered (i.e., when stake of honest vehicles gets larger) in a new transaction with SCID=0001.
- A vehicle moving to another region may apply to transfer its records in a transactions with SCID=0002.

The transactions received within a period (i.e, the round latency of trust points blockchain) are processed with our designed contracts below.

1) Instant voting contract: Transactions with SCID=0000 are to report attackers and falsified messages by our designed instant voting contract. Let us use the highway merging application in Fig. 2(a) to help explain the contract. First, the attacker near the ramp sends out a falsified message with wrong traffic status to mislead vehicles on the ramp. If there are honest vehicles within both the examination and communication ranges, they can also receive the message and may deem it suspicious based on the information from their own on-board sensors. They can then generate transactions with SCID=0000 to report such findings and broadcast the transactions to other vehicles. Within a period, surrounding vehicles are all supposed to take a stand. Then other vehicles in the region collect all transactions and start the smart contract based voting process. The voting process for this contract instance will classify all transaction senders into two groups. One group of vehicles agree with the message's content sent by the vehicle with debateID, and the other group of vehicles disagrees with that. The contract will make a final judgment and update vehicles' trust points by comparing the two groups' accumulated stake. The group with the higher stake will be called majority and have their trust points increased by 1, while the other group will be called minority and have trust points of -1.

[1]How vehicles may verify messages based on physical information from their own sensors or other sources is beyond the scope of this paper, which focuses on the design of the blockchain framework.

2) Redressing contract: A group of attackers with high stake may dominate the voting process if the number of surrounding honest vehicles is initially limited. To avoid repeated attacks from the group of attackers, we design the redressing contract, which allows re-evaluation of vehicles' opinions on the message content from the vehicle with debateID. In particular, if a vehicle disagrees with the previous voting result, it can send a transaction with SCID=0001, which triggers the redressing process. The redressing contract finds all the ended contracts that involve the debateID, and form groups G_s and G_o that represent all the vehicles agree and disagree with debateID, respectively. Let $N(G_s)$ and $N(G_o)$ denote the accumulated stake for group G_s and G_o, respectively. If the stake difference between the two groups is larger than a threshold N_{th}, previous judgement can be redressed.

3) Transferring records contract: When a vehicle moves across different regions, it is required to update its current active region. This way the vehicles within the same active region can communicate efficiently and the real-time performance can be improved. To update the active region, a vehicle needs to transfer trust points and copy its proof-of-travel credits to the new region. Note that the trust points of this vehicle in the original region should be set to zero.

C. Proof-of-Travel Blockchain

Proof-of-travel blockchain is mainly to record vehicles' accumulated contributions to other vehicles. We assume that CVs will broadcast traffic information, e.g., through BSMs, in an average frequency of f_m Hz. Surrounding vehicles within the communication range will record the public keys of message senders and the number of valid messages. After each travel period T_{pot} (which is much longer than the period for trust points blockchain), every vehicle v_i will have a list that records the public keys of message senders v_j and the number of messages n_{j2i} from every sender during this period. We propose a metric n_j^{pot}, named "proof-of-travel credits", to summarize all messages commenting on vehicle v_j from all other vehicles v_i, i.e.,

$$n_j^{pot} = \sum_i n_{j2i} \tag{1}$$

For a vehicle, its accumulated proof-of-travel credits in the latest N_{sum} periods is computed by:

$$N_j^{pot}[m] = \sum_{k=0}^{N_{sum}-1} \alpha^k \; n_j^{pot}[m-k] \tag{2}$$

where $n_j^{pot}[m-k]$ is the proof-of-travel credits in the $(m-k)_{th}$ period, α is a discounting factor, $N_j^{pot}[m]$ is the accumulated N_{sum} periods proof-of-travel credits by the m_{th} period. Vehicles that make a persistent contribution to others can earn more credits. It is noted that honest vehicles can also generate transactions to reveal falsified number report, in a similar way as in the trust points blockchain.

Fig. 3 presents the general process of updating the proof-of-travel blockchain. Different vehicles may get registered in different permanent regions, e.g., region A, region B, and so on. We assume that each vehicle is uniquely identified by the regionID and the vehicle index in that region, e.g., A1 is a vehicle with the index of 1 in region A. Since a vehicle may change lane, overtake, choose different routes, or even enter another region, it will encounter different vehicles in the same or different regions. It will record those vehicles and the number of messages they sent in a transaction and broadcast it in the region. As in the figure, each region will have a number of aggregated transactions. Each transaction is denoted by the vehicle senderID with a set including all vehicles it encountered, e.g., E1: {E2, B3, F2, $\cdots$} is the transaction sent by vehicle E1. The selected block proposer will then arrange these transactions and build a *pre-block* by summarizing credits for all vehicles in the sets. After the consensus process, all vehicles in the region will agree with this pre-block. We name it pre-block because it may not include the complete information for vehicles in the region, but record vehicles that are not in this region. Then vehicles in neighboring regions will communicate with each other regarding the pre-blocks. Finally block proposers will build a block containing the complete information of vehicles in their region.

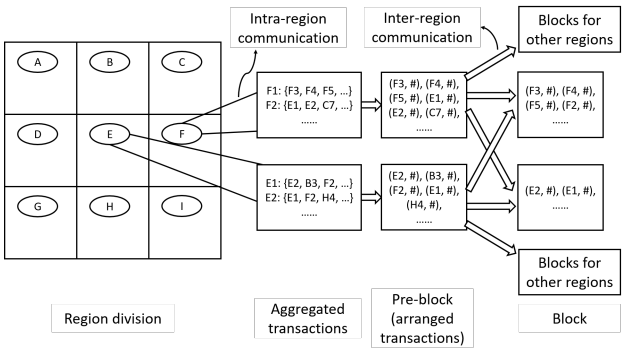

Fig. 3. Proof-of-travel blockchain updating across multiple regions. With intra-region communication, each vehicle can aggregate transactions received from others in the same region. By computation and arrangement, vehicles can build a pre-block and reach consensus on it, which includes vehicle ID and proof-of-travel credits. Note that records in pre-blocks may not be complete as vehicles may move across multiple regions; but with inter-region communication, blocks for each region will be eventually built.

D. Stake Computation and Region Size

In our framework, the stake N^{stk} is computed from the accumulated proof-of-travel credits N^{pot} in the proof-of-travel blockchain and the trust points N^{tp} in the trust points blockchain:

$$N^{stk} = \frac{N^{pot}}{1+M^{pot}} + \frac{N^{tp}}{1+M^{tp}} \tag{3}$$

where M^{pot} and M^{tp} denote the mean value of N^{pot} and N^{tp} for all vehicles, respectively. Here we use $1+M^{pot}$ and $1+M^{tp}$ as denominator to avoid the near-zero case. This stake is the basis of consensus mechanism for both the trust points blockchain and the proof-of-travel blockchain.

The region size that one blockchain can cover is decided by various factors. First, it is limited by the round latency that we desire. A larger size may result in longer latencies to reach consensus, thus slowing down system's response to attacks. Second, for the trust points blockchain, the smaller the region is, the lower the resource demand is for all vehicles in the region to maintain and update the blockchain. However, a small region size typically means that vehicles will frequently move across different regions, thus frequently transferring their records. For the proof-of-travel blockchain, a smaller region size leads to lower overhead in intra-region communications but higher overhead in inter-region ones.

IV. Experimental Results

As discussed in the threat models in Section II-A, a malicious vehicle can broadcast falsified messages, report falsified misbehavior of honest vehicles, or create other pseudonymous identities for manipulating the voting process.

We evaluate the effectiveness of our proposed framework in defending against the three threat models, based on simulations in SUMO. We also analyze the relationship between round latency of the trust points blockchain and the region size in our framework.

A. Defense Against Message Spoofing Attack

For the threat model described in Fig. 2(a), under message spoofing attack and without our framework, the merging vehicle in on-ramp can be fooled to speed up when the highway is congested or to decelerate when there are few vehicles on the highway. When the merging vehicle is able to perceive the traffic environment, it has to adjust its velocity significantly, which leads to low efficiency and possible failure to merge onto the highway. With the protection of our framework, other honest vehicles on the highway may report this spoofing attack and launch the voting process to assess the message. Before the assessment is finished, the merging vehicle can start taking cautious actions within milliseconds, e.g., speeding up only a little bit for reacting to uncertain traffic status on the highway, preventing sudden velocity changes. When all vehicles reach consensus after the voting process, the identity of the attacker is exposed, and its messages are not trustworthy any more.

We compare the average speed, CO emission, and fuel consumption of the merging vehicles under three cases, i.e., without attack, under attack without our framework, and under attack with our framework. We perform 20 simulations for each case, and the average performance is recorded in Table I, where 'ours' is short for our framework. During simulations, vehicle arriving rate on highway and on-ramp are both 0.2 vehicle per second. The performance is measured over the 200 meters on-ramp and the following 200 meters highway. The table shows that without the protection from our framework, the attack can lead to a 42.66% decrease in average speed and an 87.06% increase in CO emission. With our framework, the vehicles can take cautious actions and effectively mitigate the attack, providing a significantly higher travel speed and lower CO emission.

TABLE I

Effectiveness of our framework in protecting against message spoofing attack in highway merging.

performance	without attack	under attack without ours	under attack with ours
average speed (m/s)	16.41	9.41	13.00
CO (mg)	741.06	1386.24	1006.36
fuel (mL)	41.17	41.91	42.96

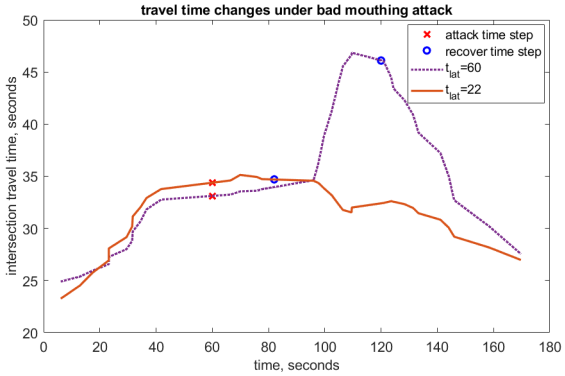

Fig. 4. Effectiveness of our framework against bad mouthing attack in an intelligent intersection. The red stars denote the moment that the attack starts and the blue circles denote the moment that our framework detects the attack and the traffic system starts recovering from the attack. Travel time curves with different round latency ($t_{lat} = 22$ and 60 seconds) of the trust points blockchain are plotted.

B. Defense Against Bad Mouthing Attack

Under the bad mouthing attack as shown in Fig. 2(b), all honest vehicles near the intersection are reported to have misbehavior by the attacker. Those honest vehicles may generate transactions to claim that they are innocent and being attacked. Before it is confirmed, all vehicles near the intersection will take cautious actions and choose not to trust others' messages; thus, the intelligent intersection may temporarily lose its functions.

Fig. 4 shows the traffic in an intelligent intersection that is under attack. The vehicle arriving rate is 0.05 vehicle per second in the intersection. Around time 60 (seconds), the attacker starts the bad mouthing attack. However, the design of our framework ensures that the attack can be soon discovered and the system can return to normal shortly. Specifically, in one round of trust points blockchain, the surrounding vehicles have reached consensus on the existence of this attack. An honest vehicle will then trust messages from other honest vehicles, and the function of intelligent intersection gets recovered. From the figure, we can see that if our framework has a smaller round latency of $t_{lat} = 22$ seconds for the trust points blockchain, the system can recover before any significant increase of travel time. If the round latency reaches $t_{lat} = 60$ seconds, travel time starts increasing initially when the intelligent function is temporarily disabled, but starts decreasing around time 120 and fully recovers around time 140. More analysis of the impact of round latency is shown later in Section IV-D.

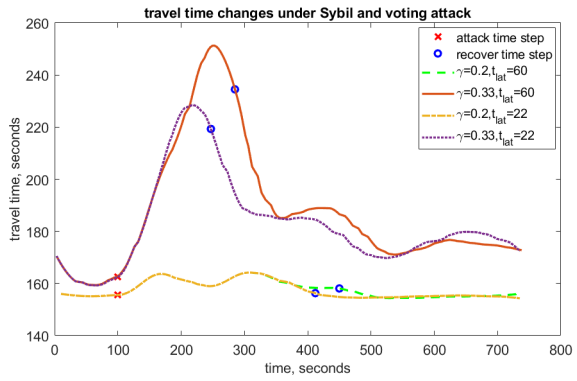

Fig. 5. Effectiveness of our framework against Sybil and voting attack. Red stars denote the moment the attack starts, and the blue circles denote the moment the attack is detected and the recovery starts. Travel time curves with different round latency ($t_{lat} = 22$ and 60 seconds) for the trust points blockchain and under different vehicle arriving rates ($\gamma = 0.2$ and 0.33 vehicle per second) are plotted.

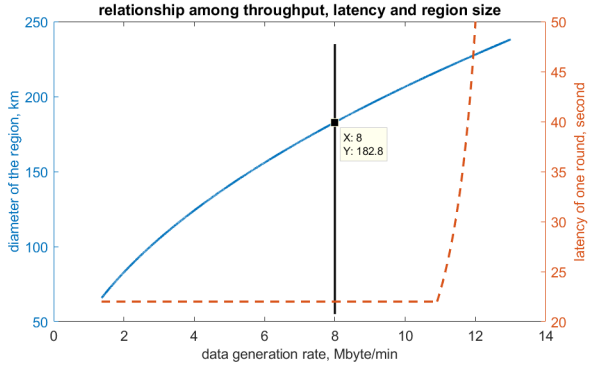

Fig. 6. Maximum region size and minimum round latency under different transaction generation rate (best viewed in color). The solid blue line and the dashed orange line represent the maximum region size and the minimum round latency under different transaction generation rate, respectively. Drawing a vertical line, we can see the minimum round latency corresponding to the region size, e.g., 22 seconds for a region with a radius of 182.8 km.

C. Defense Against Sybil and Voting Attack

Fig. 2(c) shows the threat model where a traffic network with route choices is attacked with Sybil and voting attack. Specifically, an attacker and the two Sybil vehicles it generates send transactions reporting a traffic accident in one route and win the voting process because they have more stake initially. Then other vehicles arriving at the area will have a higher probability (e.g., 0.9 in our simulations) to take another route, which leads to longer travel time. However, as several vehicles still choose the route with the fake accident and realize that it is a Sybil and voting attack, they trigger the redressing process as the honest vehicles have higher stake now. Fig. 5 shows that our framework can soon discover the Sybil and voting attack, and recover to normal status. We also analyze the impact of vehicle arriving rate and round latency of trust points blockchain, and observe that a higher vehicle arriving rate leads to longer travel time under attack but faster recovery time in our framework. A smaller round latency may reduce the travel time under attack and also shorten the recovery time.

The results above demonstrate the effectiveness of our framework in protecting against message spoofing, bad mouthing, and Sybil and voting attacks. Note that with our proof-of-travel blockchain, it is even harder for malicious attackers to perform these attacks (especially Sybil attack), as it takes more effort for them to build up a travel record in the region with high stake.

D. Round Latency and Resource Overhead

In the results above, we have seen the impact of round latency of the trust points blockchain on system performance. This latency is closely related to the region size. Here we conduct more in-depth analysis on the relationship between the two.

In [15] and [16], the authors simulated 50k Algorand users on Amazon's EC2 platform. The bandwidth for each Algorand process is set to 20 Mbps. Latency for one round of Algorand is about 22 seconds, dominated by the time to gossip a 1 MB block through the user network [16].

We assume that the blockchain will record all transactions sent by vehicles in a circular region with a radius of d_0. The size of a block is computed as:

$$S_b = \left(\frac{2\pi d_0 \beta_l \beta_c}{\beta_v} + \pi d_0^2 \beta_d \beta_t\right) t_{lat} S_t^{SCID} \tag{4}$$

where β_l is the number of lanes connect to outer space per perimeter of the region, β_c is the traffic capacity of a single lane with the unit as the number of vehicles traveled per hour per lane, β_v is the number of vehicles that can be claimed in a transaction of type SCID=0002, β_d is the density of vehicles in this region, β_t is the generation rate of transactions of type SCID=0000/0001, t_{lat} is the latency (period) to generate a block, S_t^{SCID} is the size of a transaction.

We set the capacity of lanes that connect to other regions, β_c, to be 3000 vehicles per hour per lane, as observed in the data collected from a section of the highway I-5 in Southern California [23] and should be similar in other regions. The density of vehicles β_d is set to 300 veh/km^2, which is similar to the density in Beijing [24]. Considering that attacking events are rare and vehicles are not driving on the road all the time, β_t is set to 0.05 transactions per hour per vehicle. S_t^{SCID} is 250 byte per transaction. β_v is set to 10 vehicles per transaction and β_l is 1 lane per km.

Taking a region with a similar size to Beijing, the transaction data generation rate is about 1.6 Mbyte/min. According to [16], the latency for one round of Algorand has little variation on the number of users. It basically remains the same when the block size is smaller than 4MB and then increases proportionally with a larger block size. We plot the relationship among throughput (data generation rate), minimum latency, and region size in Fig. 6, and we can clearly see the trade-offs between the minimum round latency and the maximum region size.

For the resource overhead analysis of the trust points blockchain, we consider an area with similar size to Beijing.

The computation cost mainly depends on the verifier committee size rather than the total number of vehicles, because only block proposers and verifiers will broadcast messages. If the computation cost is measured by the utility of one core in a 2.4 GHz Intel Xeon E5-2676 v3 (Haswell) Processor, then it is 6.5 percent when 80 percent of the vehicles are honest. As for the communication cost, each vehicle may receive about 10.2 Mbyte data per minute when 80 percent of the vehicles are honest and the round latency is 22 seconds. The communication cost will increase with a smaller fraction of honest vehicles, a smaller round latency, or a larger region size. As for the storage cost, we assume that selected block proposers will summarize the status of all vehicles in new blocks within a period of t_{sum}. By indicating the type of block in its header, a new vehicle can easily find the latest summary blocks and does not need to download blocks generated before those. In this way, a new vehicle may only need to download about 6.5 Gbyte data to get the latest status of blockchain when t_{sum} equals 24 hours. Please refer to [25] for more details of the resource overhead analysis.

To alleviate resource demand for the proof-of-travel blockchain, vehicles can generate blocks and update the blockchain in a much larger period than that of the trust points blockchain, e.g., one day or longer. Furthermore, With the pre-block arrangement step shown in Fig. 3, the resource demand for the proof-of-travel blockchain should be much less than that for the trust points blockchain.

V. Conclusion

In this work, we propose a dual cyber-physical blockchain framework that includes a trust points blockchain and a proof-of-travel blockchain for building trust and securing communication for CV applications. The trust points blockchain has a quick response to suspicious behavior. The proof-of-travel blockchain builds up reputations from vehicles' long-term travel records. The trust points and the proof-of-travel credits are used together to compute the vehicle stake for running the consensus mechanisms in both blockchains. Experimental results demonstrate the effectiveness of our framework. As this is the first step towards building a trust framework for CV applications, we plan to continue improving the efficiency of our framework and its prototyping.

Acknowledgment

We gratefully acknowledge the support from NSF grants CNS-1839511, CNS-1834701 and UC Lab fees grant LFR-18-548554.

References

[1] "Connected vehicle pilot deployment program." [Online]. Available: https://www.its.dot.gov/pilots/crosssite_cvp.htm

[2] "Connected vehicle test bed." [Online]. Available: https://www.its.dot.gov/research_archives/connected_vehicle/dot_cvbrochure.htm

[3] Q. A. Chen, Y. Yin, Y. Feng, Z. M. Mao, and H. X. Liu, "Exposing congestion attack on emerging connected vehicle based traffic signal control." in *NDSS*, 2018.

[4] B. Zheng, C.-W. Lin, S. Shiraishi, and Q. Zhu, "Design and analysis of delay-tolerant intelligent intersection management," *ACM Transactions on Cyber-Physical Systems*, vol. 4, no. 1, pp. 1–27, 2019.

[5] A. Abdo, S. M. B. Malek, Z. Qian, Q. Zhu, M. Barth, and N. Abu-Ghazaleh, "Application level attacks on connected vehicle protocols," in *22nd International Symposium on Research in Attacks, Intrusions and Defenses ({RAID} 2019)*, 2019, pp. 459–471.

[6] S. Parkinson, P. Ward, K. Wilson, and J. Miller, "Cyber threats facing autonomous and connected vehicles: Future challenges," *IEEE transactions on intelligent transportation systems*, vol. 18, no. 11, pp. 2898–2915, 2017.

[7] S. Gurung, D. Lin, A. Squicciarini, and E. Bertino, "Information-oriented trustworthiness evaluation in vehicular ad-hoc networks," in *International Conference on Network and System Security*. Springer, 2013, pp. 94–108.

[8] M. Raya, P. Papadimitratos, V. D. Gligor, and J.-P. Hubaux, "On data-centric trust establishment in ephemeral ad hoc networks," in *IEEE INFOCOM 2008-The 27th Conference on Computer Communications*. IEEE, 2008, pp. 1238–1246.

[9] J. Kolb, M. AbdelBaky, R. H. Katz, and D. E. Culler, "Core concepts, challenges, and future directions in blockchain: A centralized tutorial," *ACM Computing Surveys (CSUR)*, vol. 53, no. 1, pp. 1–39, 2020.

[10] R. Iqbal, T. A. Butt, M. Afzaal, and K. Salah, "Trust management in social internet of vehicles: Factors, challenges, blockchain, and fog solutions," *International Journal of Distributed Sensor Networks*, vol. 15, no. 1, p. 1550147719825820, 2019.

[11] M. Singh and S. Kim, "Branch based blockchain technology in intelligent vehicle," *Computer Networks*, vol. 145, pp. 219–231, 2018.

[12] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1495–1505, 2018.

[13] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. WieBner, "Microscopic traffic simulation using sumo," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2575–2582.

[14] Y. L. Sun, Z. Han, W. Yu, and K. R. Liu, "A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks," in *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*. IEEE, 2006, pp. 1–13.

[15] J. Chen and S. Micali, "Algorand: A secure and efficient distributed ledger," *Theoretical Computer Science*, vol. 777, pp. 155–183, 2019.

[16] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich, "Algorand: Scaling byzantine agreements for cryptocurrencies," in *Proceedings of the 26th Symposium on Operating Systems Principles*, 2017, pp. 51–68.

[17] M. Pustišek, A. Kos, and U. Sedlar, "Blockchain based autonomous selection of electric vehicle charging station," in *2016 international conference on identification, information and knowledge in the Internet of Things (IIKI)*. IEEE, 2016, pp. 217–222.

[18] F. Gao, L. Zhu, M. Shen, K. Sharif, Z. Wan, and K. Ren, "A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks," *IEEE network*, vol. 32, no. 6, pp. 184–192, 2018.

[19] Y. Yuan and F.-Y. Wang, "Towards blockchain-based intelligent transportation systems," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 2663–2668.

[20] W. Dong, Y. Li, R. Hou, X. Lv, H. Li, and B. Sun, "A blockchain-based hierarchical reputation management scheme in vehicular network," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[21] A. Dorri, S. S. Kanhere, and R. Jurdak, "Towards an optimized blockchain for iot," in *2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI)*. IEEE, 2017, pp. 173–178.

[22] T. Jiang, H. Fang, and H. Wang, "Blockchain-based internet of vehicles: Distributed network architecture and performance analysis," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4640–4649, 2018.

[23] "Caltrans pems." [Online]. Available: http://pems.dot.ca.gov/

[24] J. Yang, Y. Liu, P. Qin, and A. A. Liu, "A review of beijing's vehicle registration lottery: Short-term effects on vehicle growth and fuel consumption," *Energy Policy*, vol. 75, pp. 157–166, 2014.

[25] X. Liu, B. Luo, A. Abdo, N. Abu-Ghazaleh, and Q. Zhu, "Securing connected vehicle applications with an efficient dual cyber-physical blockchain framework," 2021.