

Quantum-Resistant Safety Message Authentication for NextG C-V2X: A Cross-Layer Approach

Geoff Twardokus, *Member, IEEE*, and Hanif Rahbari, *Member, IEEE*

Abstract—The safety and reliability of next-generation connected vehicles (CVs) depend on authenticating vehicle-to-vehicle (V2V) safety messages. Today, these basic safety messages (BSMs) are secured using elliptic-curve signatures that are vulnerable to attacks using a quantum computer, making a transition to quantum-resistant cryptography (QRC) imperative. However, the overhead of QRC (i.e., large keys and signatures) conflicts with the message size and latency constraints of Cellular Vehicle-to-Everything (C-V2X) protocols, while splitting them across lossy transmissions can delay BSM verification. In this work, we develop CHASM, a cross-layer solution to address this problem. At the physical layer, CHASM combines a kinetic safety model with feedback-free estimation of neighbors’ signal-to-noise ratios to select a modulation and coding scheme (MCS) that accelerates reliable certificate delivery to safety-critical receivers, thereby reducing BSM verification delay. At the application layer, CHASM introduces a generalized certificate distribution protocol that splits large hybrid certificates across multiple transmissions using fragmentation, erasure coding, or other approaches. CHASM couples certificate splitting with safety-aware MCS selection to meet receiver-specific certificate-delivery deadlines. We show that CHASM provides quantum-resistant message authentication while preserving classical security guarantees—a backward-compatible, crypto-agile solution. We prove BSMs under CHASM are EUF-CMA unforgeable against both classical and quantum adversaries. Benchmarking on commercial C-V2X hardware shows that CHASM adds less than 3.5 ms of average BSM verification delay. Further, city-scale simulations recording over 65,000 safety events per evaluation show CHASM reduces the safety violation rate to 1.94%, from 12.06% in prior work and 24.13% under IEEE 1609.2.

I. INTRODUCTION

Connected vehicle (CV) technology is rapidly growing as a critical means to enhance safety and coordination on roadways. By enabling vehicles to exchange messages that improve situational awareness for split-second decision-making—e.g., whether to brake or swerve to avoid an out-of-sight hazard—CVs may save tens of thousands of lives each year [1]. However, the ability of CVs to authenticate these safety messages may be compromised in as little as 10–15 years by the development of cryptographically relevant quantum computers [2].

Copyright (c) 2026 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

This material is based upon work supported by the U.S. National Science Foundation under Award No. 2239931. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. We thank Sameer Gopali for his assistance with hardware experiments.

Geoff Twardokus is with the College of Computing, Grand Valley State University, Grand Rapids, MI, USA (email: twardokg@gvsu.edu).

Hanif Rahbari is with the College of Computing and Information Sciences, Rochester Institute of Technology, Rochester, NY, USA (email: hanif.rahbari@rit.edu).

An adversary with access to such a computer could bypass today’s cryptographic protections and inject falsified messages, enabling attacks that mislead human or autonomous drivers, manipulate traffic flow, and even engineer crashes, negating safety benefits. Mitigating these anticipated quantum attacks requires transitioning systems to *quantum-resistant cryptography* (QRC), which is already underway for everyday protocols such as TLS on the Internet [3]. However, accommodating QRC’s overhead—namely, signatures and keys that are orders of magnitude larger than current cryptographic primitives—is far from trivial in constrained, low-latency CV protocols.

CVs rely on 4G and 5G cellular vehicle-to-everything (C-V2X) technology to directly exchange time-sensitive safety messages using vehicle-to-vehicle (V2V) communication [4]–[6]. To meet latency and reliability requirements for a large number of vehicles sharing limited bandwidth in dynamic, fast-fading wireless channels, C-V2X *sidelink* strictly limits payload size to as little as 867 bytes per packet [7, Table 5.1.3.2-1]. This constraint has previously been an obstacle only for bandwidth-hungry applications, such as cooperative sensing [8], but now also poses a challenge for safety and security. Under current standards [9]–[11], small (64-byte) elliptic-curve digital signatures and public keys are combined with a 64-byte basic safety message (BSM) and metadata to reach a total of 200–300 bytes per packet (depending on whether a certificate is included with the transmission), easily meeting the size constraint. However, standardized¹ QRC requires ~1,500–17,000 bytes just for a public key and signature, let alone the actual BSM, headers, and metadata. Fig. 1 illustrates how QRC does not fit within C-V2X constraints.

Prior work has largely considered physical (PHY) and application-layer approaches to the sidelink packet size problem in isolation, falling short for C-V2X. For example, Sloop et al. [12] use federated learning to optimize modulation and coding scheme (MCS) selection and improve data rates at the PHY layer, which could help loosen the packet size limit. However, their target domain (sidelink for disaster settings) assumes a stable set of nodes that *cooperate* on selecting optimal parameters, which is inapplicable to V2V environments where vehicles interact only briefly with transient neighbors. On the other hand, multiple prior works have introduced certificate fragmentation or signature splitting at the application layer [13]–[15]. These mitigate the size constraint at the expense of increasing BSM verification’s sensitivity to packet loss—we show that merely splitting a certificate across multiple BSMs can have safety consequences: losing a single message from a signed batch delays verification of

¹Emerging “on-ramp” QRC does not solve the problem—see Section II.

all associated safety data. Techniques used outside of V2V, such as key negotiation in quantum-resistant TLS [3], usually depend on bidirectional, persistent connections not present in ephemeral, broadcast-only V2V.

In this paper, we argue that efficiently integrating QRC under reliability, latency, and associated safety constraints requires careful *cross-layer* attention. While overhauling the protocol stack (e.g., supplanting public-key authentication with blockchain security [16], [17]) would require revising multiple standards and retrofitting every deployed system—a costly, multi-year endeavor—we instead aim to revise and extend existing standards. Our premise is to increase alignment between the application and PHY layers, enabling QRC adoption while preserving the critical safety responsibilities of C-V2X.

Safety-Aware Adaptive MCS Selection—As evident in Fig. 1, adopting QRC without addressing the payload size constraint is a non-starter. We tackle this by first examining the PHY layer. In high-mobility V2V, transmitters cannot rely on receiver-specific feedback to aggregate into one stable channel state information (CSI) estimate for broadcast transmissions. Current standards thus disregard instantaneous channel conditions for MCS selection and rely on a lookup table based only on the amount of scheduled data and the sender’s speed [11, Table 11]. This approach is biased towards low-order MCS and prioritizes reliable delivery to *every* neighbor—overlooking both the safety relevance of individual receivers and their potentially favorable signal-to-noise ratio (SNR). Another reason for this omission is the difficulty of blindly estimating SNR in dynamic, fast-fading settings—challenges we aim to address.

Current standards set a heuristic requirement for each vehicle to share its certificate every 500 ms [11], implicitly assuming that this is sufficient to prevent safety risks from arising while unverified BSMs await sender certificates. Instead of a heuristic approach, we propose adapting MCS to prioritize maximizing the reliable data rate—and thus accelerating certificate distribution—to receivers who pose a safety risk. This requires ① identifying which neighbors pose a more imminent threat (e.g., those on a collision course), and ② determining the maximum time those neighbors can wait to receive the sender’s certificate and still verify any queued BSMs that might trigger evasive action. We jointly address these by developing a kinetic safety model for scenarios that captures rear-end, cross-traffic, and lane-merging conflicts, allowing a vehicle to use past BSMs to predict conflicts and derive the acceptable certificate delay for critical receivers.

After determining the certificate communication deadline, the sender must ③ estimate the expected SNR at those receivers without using CSI; and ④ select the best MCS such that certificate delivery speed and reliability are balanced across safety-critical receivers. We tackle problem ③ with a new method of coarsely predicting SNR *without feedback*, leveraging a Rician fading model with parameters (e.g., K -factor) predicted based on local conditions (e.g., urban vs. rural, presence or absence of line of sight). Finally, we address ④ by unifying these components to select an MCS that meets both the SNR thresholds and certificate delay requirements, balancing reliability with safety in a unified approach.

Generalized Split-Distribution of Hybrid Certificates—

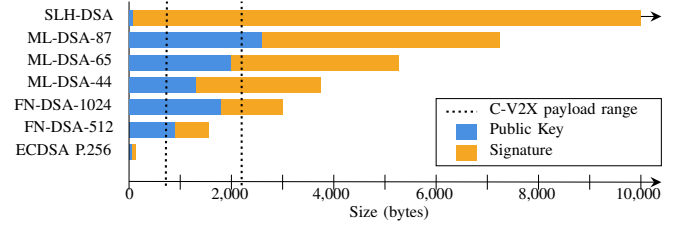


Fig. 1. Public key and signature sizes for ECDSA vs. NIST-approved QRC. Dashed lines indicate the payload range in a typical 20 MHz C-V2X channel.

Adaptive MCS can increase reliable throughput to safety-critical receivers, but the application layer must still distribute certificates efficiently over those links. This requires ⑤ defining a secure certificate-signature scheme that integrates QRC while meeting existing security guarantees and supporting legacy vehicles throughout the transition period; and ⑥ efficiently spreading certificates across multiple transmissions in a reliable and latency-sensitive manner.

To address challenge ⑤, we must balance the need to support legacy ECDSA-only vehicles against the certainty that classical signatures will ultimately be removed (e.g., NIST proposes disallowing ECDSA in 2035 [18]). We address this by developing a *hybrid* protocol. In the short term, legacy vehicles can continue to use ECDSA, while newer vehicles can use QRC instead of (or alongside) it. In the long term, as legacy vehicles reach end of life (or ECDSA is disallowed), support for ECDSA can be seamlessly removed by setting a single protocol flag to disallow ECDSA. This *crypto-agility*, which enables a fully quantum-secure ecosystem without disrupting operations, is critical for practical deployment.

We approach problem ⑥ by designing a *generalized* protocol wherein certificates are spread across multiple transmissions, while leaving the specific splitting method open-ended. We flexibly support different mechanisms for encoding and transmitting certificates. In our evaluations, we consider instantiations based on certificate fragmentation and systematic Reed-Solomon erasure codes, contrasting their performance and computational requirements while leaving the door open for future alternatives. Our analysis provides evidence to guide standards bodies in selecting an optimal approach for real-world deployment.

Contributions—We develop a hybrid, cross-layer, and *safety-conscious* security protocol to bridge the gap between today’s 4G/5G, ECDSA-only environment and fully quantum-resistant, mixed-generation future fleets. Our intergenerational support, from today’s 4G C-V2X through emerging 5G and future 6G systems [19], accommodates vehicles’ 13–15-year lifetimes [20], ensuring legacy vehicles are not prematurely excluded due to hardware or radio limitations. Specifically:

- **Safety-Aware Adaptive MCS Selection.** We develop a new technique for MCS selection in C-V2X, leveraging application-layer safety data and a novel PHY-layer technique for feedback-free SNR estimation to accommodate the overhead of QRC in a manner that prioritizes limiting BSM verification delay to avoid unsafe outcomes.
- **Certificate-based Hybrid Authentication of Safety Messages (CHASM).** We introduce CHASM to support

efficient distribution of large *hybrid* certificates across all generations of C-V2X, using flexible techniques (e.g., fragmentation or erasure coding) to balance BSM reliability and verification latency in an adaptive system.

- **Message Unforgeability.** We provide security reductions for CHASM that prove its security against signature forgery under both classical and quantum threat models.
- **Experimental Validation.** We demonstrate the advantage of CHASM through city-scale simulations that compare different instantiations of our protocol against an IEEE 1609.2 baseline and the most relevant approach from prior work. Over more than 65,000 recorded safety events per tested configuration, CHASM with FN-DSA² signatures, adaptive MCS selection, and erasure-coded certificate distribution accelerates reliable certificate delivery, reducing the violation rate to as little as 1.94% vs. ≈ 12 –24% under baseline and prior-work approaches, while benchmarking on commercial C-V2X hardware shows CHASM adds an average delay of < 3.5 ms for BSM processing and increases CPU load by $\leq 5\%$.

We describe our motivation and threat model in Section II. We present CHASM in Section III and formally prove its security in Section IV. We present experimental results in Section V, discuss related work in Section VI, and conclude with Section VII.

II. MOTIVATION AND THREAT LANDSCAPE

To enhance awareness for safer driving, each CV periodically broadcasts BSMs (every ~ 100 ms) that convey its motion—location, speed, heading, etc.—to neighboring vehicles within a ~ 600 m communication range. BSMs are authenticated following procedures defined by the IEEE 1609.2 standard [9], which protects BSMs by encapsulating them in secure protocol data units (SPDUs). In addition to a BSM, each SPDU contains the sender’s ECDSA signature over that BSM and (sometimes) the sender’s certificate, which contains the ECDSA public key needed to verify its message signatures. As Fig. 1 illustrates, this minimal security overhead fits within the limits of C-V2X packet size.

For efficiency, current standards dictate that only 20% of SPDUs (roughly every fifth transmission) should include the full certificate, with the remaining 80% instead carrying only a hash of the certificate to identify the sender [11]. This heuristic mechanism conserves bandwidth while nominally limiting the delay for verifying any BSM to under 500 ms, attempting to balance latency with efficiency, but without directly considering safety—which CHASM aims to address. To preserve privacy, each vehicle is issued several valid certificates at a time, and it rotates the certificate that is actively used to sign messages roughly every five minutes. To manage the resulting large volume of certificates, a public key infrastructure (PKI) is employed, wherein certificate authorities (CAs) issue certificates to vehicles and manage services such as revocation.

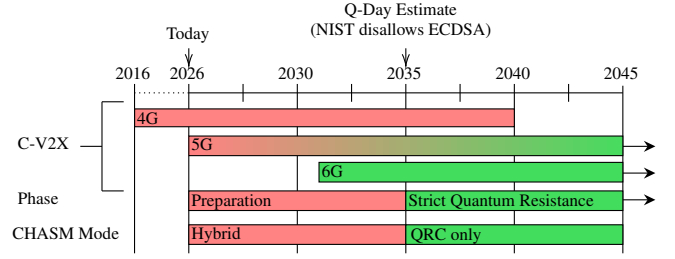


Fig. 2. CHASM is designed to keep up with expected evolution of C-V2X and increasing power of quantum computers over the next 10–20 years.

A. Motivation: The Emerging Quantum Threat

The transition from classical to quantum-capable attackers will track the increasing capabilities of quantum computers, which we divide into two stages (Fig. 2). We are currently in the *preparation phase*, in which a quantum computer that can easily break algorithms like ECDSA does not exist yet. Threats in this phase are from adversaries with classical computing power. Systems should be transitioned to QRC during this period, but QRC is not yet mandatory.

Once a quantum computer powerful enough to break ECDSA is developed—an event commonly referred to as “Q-Day”—an attacker may have local or online (e.g., subscription-based) access to a quantum computer that can break the 256-bit ECDSA keys and signatures used under IEEE 1609.2, unlocking new attacks. For example, an attacker could harvest other vehicles’ public keys from their SPDUs (which are unencrypted) and then use Shor’s algorithm [22] to efficiently recover the corresponding private keys before they expire. Armed with those private keys, the attacker could then forge valid signatures on bogus messages and inject them into the environment, manipulating drivers’ perceptions and potentially tricking them into making unsafe decisions or even crashing. Since legitimate signatures and forgeries would be cryptographically indistinguishable, this attack would be extremely difficult to detect, undermining trust in C-V2X networks.

To mitigate these and other attacks, we delineate the era after Q-Day as the *strictly quantum-resistant* phase in which QRC will be mandatory, and vulnerable algorithms like ECDSA will be disallowed. Conceptually, this is straightforward, but smoothly transitioning between the preparation and quantum-resistant phases is not easy. Transitioning too early would unnecessarily kick out legacy (ECDSA-only) equipment, while transitioning too late would expose all users to quantum threats. We therefore contend it is necessary to equip all new vehicles with *both* security regimes during the preparation phase—ECDSA to communicate with legacy vehicles, and QRC for newer units.

Choice of QRC. NIST has standardized two QRC signature algorithms: Module Lattice-based Digital Signature Algorithm (ML-DSA) [23], and Stateless Hash-Based Digital Signature Algorithm (SLH-DSA) [24]. A third, FN-DSA [21]—formerly known as Falcon—is pending finalization by NIST [25]. SLH-DSA is fundamentally incompatible with C-V2X due to its $\geq 17,000$ -byte signatures (see Fig. 1) and slow runtime on V2V hardware [13]; therefore, we target CHASM for ML-DSA-44

²Fast-Fourier Transform over NTRU-Lattice-Based Digital Signature Algorithm (FN-DSA) [21].

and FN-DSA-512.

We focus on (near-)standardized QRC because emerging “on-ramp” algorithms [26], while slightly reducing overhead, do not completely resolve the packet size problem. These new algorithms may also harbor undiscovered vulnerabilities (e.g., [27]), making them unsuitable for safety-critical systems.

Threat Model. We consider an attacker who seeks to disrupt, delay, or otherwise degrade normal C-V2X operations in a target area, and we assume this attacker is able to send and receive standard C-V2X signals. We consider two attackers—a classical adversary in the preparation phase, and a classical or quantum adversary in the strictly quantum-resistant phase. Critically, *CHASM does not use ECDSA for security after Q-Day*. Security in the strictly quantum-resistant phase relies solely on QRC keys and signatures, with ECDSA disallowed, and any provided ECDSA signatures disregarded. Therefore, breaking an ECDSA signature after Q-Day is of no use to the attacker. Our formal analysis (Section IV) carefully delineates these threats, and we prove the security of CHASM for each era.

Denial-of-Service Attacks—CHASM may remain susceptible to generic denial-of-service (DoS) attacks such as flooding and jamming, which also affect today’s IEEE 1609.2 systems and are beyond our scope. Mitigations for straightforward jamming attacks have long been under development, e.g., packet (or fragment) retransmissions using C-V2X HARQ or NACK mechanisms [28]. However, CHASM avoids introducing *new* resource-exhaustion vulnerabilities; for example, timeouts discard incomplete, unverified SPDUs to prevent its added caching from exhausting receiver resources.

B. Challenges: C-V2X Sidelink and MCS Constraints

C-V2X standards are maintained by 3GPP in their Releases 14–21 and beyond [6]. In this paper, we specifically focus on C-V2X *sidelink* for direct V2V communication.

Challenge 1: Sidelink-Only Operation. 4G and 5G C-V2X support two sidelink modes: in-coverage, where base stations (gNBs) allocate resources, and out-of-coverage, where vehicles operate autonomously. Operating in-coverage allows offloading sensing and scheduling to base stations and enables high-throughput (>50 Mbps) communication over cellular uplink and downlink. This vehicle-to-network (V2N) communication does not have the same packet size limit as more constrained sidelink, which would make it ideal for tasks like exchanging large QRC security data. However, the out-of-coverage (sidelink-only) mode is essential to ensure critical safety messages are not dependent on cellular service, which may be inconsistently available. We therefore focus on the more constrained out-of-coverage mode in this work.

Challenge 2: MCS Selection Without CSI. At the PHY and MAC layers, C-V2X divides a channel into 1 ms time slots, and further divides each slot’s bandwidth into equal-size *subchannels* for allocation. The total number of subchannels is derived from channel bandwidth [29, Table 5.3.5-1]. Importantly, since each subchannel has a fixed number of data symbols for a given bandwidth and subcarrier spacing, the choice of MCS is the *only* factor that the sender can control in an effort to send larger packets. Optimally selecting MCS

is therefore key to increasing the size of C-V2X messages, but current standards do not fully exploit this potential.

As there is no bidirectional communication in V2V safety services, and thus no channel feedback or reciprocity that can be used to estimate CSI³, traditional channel estimation is inapplicable, and the selection of MCS is channel- and *receiver-agnostic*. Vehicles choose their MCS for transmitting a BSM using lookup tables [11, Tables 11–12], which consider only two factors: (1) the size (in bytes) of the message to be sent, and (2) the speed of the sending vehicle. Critically, this approach does not consider the varying importance of different messages to different neighbors. For example, safety messages are far more relevant to neighbors that are on a potential collision course with the sender than to vehicles that are departing the area, but today’s approach treats delivering a BSM to both of those receivers as equally important—potentially selecting a low MCS that delays how quickly a message (or verification certificate) can be conveyed to the more dangerous neighbor. We argue that from a safety perspective, MCS selection should prioritize reliable and rapid delivery of BSMs and certificates to receivers that pose a safety risk to the sender rather than treating all receivers uniformly. We move in that direction with CHASM’s PHY-layer design.

III. CHASM: CERTIFICATE-BASED HYBRID AUTHENTICATION OF SAFETY MESSAGES

In this section, we present CHASM after describing its system model, assumptions, and requirements.

A. System Model and Operational Assumptions

Even though 5G C-V2X standards were published in 2020 (3GPP Rel. 16 [5]), devices on the market today largely still use 4G technology [19]. We anticipate greater uptake of 5G V2V in the near future, but with 13–15-year average vehicle lifetimes [31], 4G C-V2X equipment will be on the roads well into the 2030s, presumably alongside increasing numbers of 5G-capable vehicles. We therefore consider an environment with a mixed fleet of CVs using 4G, 5G, and potentially future 6G-capable C-V2X. Based on current spectrum allocations [32], we assume a single 20 MHz channel allocated for C-V2X safety services. While future regulations may increase the amount of available C-V2X spectrum, constraining our model to a single licensed band and not relying on potentially intermittent cellular connectivity provides a conservative setting for evaluating CHASM. This ensures that CHASM does not depend on additional communication resources and can also operate under less constrained conditions that might emerge later. Based on current industry standards, each vehicle is expected to broadcast a BSM every 100 ms [10], [11]. For security, we require all BSMs to be verified using digital signatures backed by public-key certificates in an approach aligned with the most recent IEEE 1609.2 standard for V2V security [9]. We shortly provide more specific security requirements.

³The physical feedback channel (PSFCH) introduced in 5G C-V2X applies only to unicast or multicast, which are not used for V2V safety services [30].

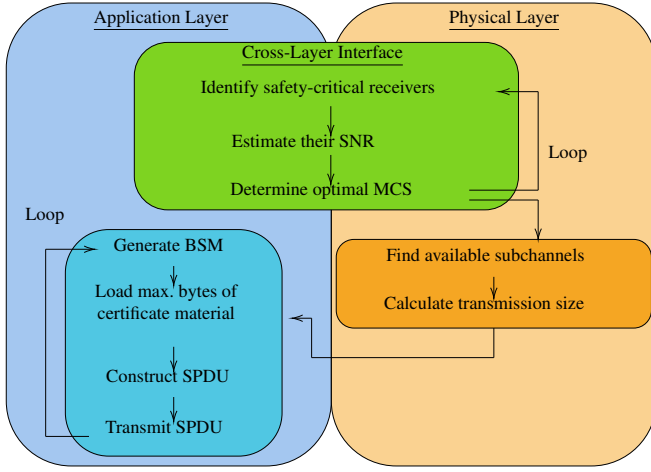


Fig. 3. Overview of CHASM's cross-layer interactions and architecture.

B. Design Goals and Requirements

In order to support both new and legacy equipment that will interoperate over the next several years, our protocol must:

- 1) Preserve existing (classical) security guarantees under IEEE 1609.2, as detailed shortly.
- 2) Provide quantum-resistant security guarantees to match the classical guarantees of existing systems.
- 3) Support ECDSA for as long as possible, while enabling a rapid and seamless switch to strict quantum-resistance (QRC-only) on Q-Day.

Further, to avoid depending on a single QRC algorithm, which might harbor unknown vulnerabilities due to the relative immaturity of QRC, CHASM must:

- 4) Support multiple QRC algorithms and allow switching algorithms without major disruption; i.e., be *crypto-agile*.

Finally, in order to ensure that enhanced security does not compromise the safety functions of C-V2X, our protocol must:

- 5) Avoid degrading or interfering with the efficient and reliable operation of C-V2X safety services.

Security Requirements. We base our security requirements for CHASM on the stated security goals of IEEE 1609.2: protecting messages against spoofing, alteration, and replay attacks [9, Sec. 1.2]. CHASM protects against spoofing and message alteration attacks by using digital signatures to authenticate and verify the integrity of BSMs. Replay attacks are mitigated by including a timestamp in each signed message. The signature prevents timestamp alteration, while freshness checks at the receiver are employed to ensure that outdated (i.e., possibly replayed) messages are rejected.

Privacy Requirements. BSM contents (e.g., vehicle speed) are not considered confidential; therefore, BSMs are not encrypted in commercial V2V systems or CHASM. However, IEEE 1609.2 allows using AES (a quantum-resistant cipher) to encrypt SPDUs if desired [9, Clause 5.3.8]. Carrying an encrypted payload does not change the message size, which is our primary concern, and CHASM supports encrypted SPDUs if required (subject to symmetric key agreement—a separate problem, beyond our scope).

IEEE 1609.2 protects a vehicle's permanent identity by employing *pseudonym certificates*, which are cryptographically unlinkable to each other or to the owner's long-term identity. Unlinkability ensures that rotating the active certificate every five minutes, as mentioned in Section II, prevents tracking based on certificates. Pseudonym certificates have already been extended to QRC [33], and CHASM is compatible with quantum-resistant pseudonymous identification.

Scope. We limit our scope to the V2V exchange of safety messages between vehicles, as this is by itself a complicated problem to solve. We therefore make a few assumptions:

- 1) We assume CAs generate and issue certificates securely and only to authorized parties.
- 2) We assume all communication between CAs and vehicles is secure over separate channels (e.g., V2N) and cannot be decrypted or influenced by an adversary.
- 3) We exclude attacks that require physical access to vehicles or infrastructure (e.g., side-channels, tampering).

C. Cross-Layer Overview

We show in Fig. 3 how CHASM couples the PHY and application layers to transmit large hybrid certificates over constrained C-V2X sidelink channels. At a high level, the PHY layer continuously evaluates the safety risk of neighboring vehicles, coarsely estimates their SNR, and determines the best MCS to optimize safety utility based on critical receivers. Because the selected MCS dictates payload size, the application layer queries the PHY layer for the byte limit immediately before constructing the next SPDU, thereby determining the amount of certificate data it must read from its buffer. This cycle repeats for every broadcast SPDU.

D. PHY-Layer: Safety-Aware Adaptive Modulation

The PHY-layer components of CHASM include (1) our kinetics-informed safety model, (2) identifying safety-critical receivers, (3) performing sender-side SNR estimation for critical receivers, (4) selecting the best MCS for safety, and (5) selecting wireless model parameters (e.g., Rician K -factor). In the remainder of this subsection, we detail these elements.

1) Safety Model: The sender can continually update a set of neighboring receivers using the received BSMs that it verifies. For each neighbor i , let the estimated time until that neighbor's path will conflict with the sender be $\hat{T}_{conflict,i}$. Then, denote the time to communicate a complete certificate to that receiver as $\tau_{cert,i}$ and the additional constant delay to verify the certificate as T_{verify} . Additionally, denote $\hat{T}_{avoid,i}$ as the estimated minimum time required for neighbor i to avoid the conflict after detection (e.g., based on braking distance). The distance interpretation of $\hat{T}_{avoid,i}$ is shown by the green circle in Fig. 4. We now define the safety requirement as

$$\tau_{cert,i}^{max} = \hat{T}_{conflict,i} - T_{verify} - \hat{T}_{avoid,i} \quad (1)$$

where $\tau_{cert,i}^{max}$ is the maximum tolerable delay for communicating a complete certificate to this receiver⁴. From a safety

⁴Safety models sometimes include a delay term for the (human) driver's reaction time. We assume future CVs will automate safety maneuvers, similar to automatic braking in today's vehicles. Thus, we omit this from our model.

perspective, conveying the sender's certificate to receivers with smaller $\tau_{cert,i}^{max}$ is more urgent. $\hat{T}_{conflict}$ and $\hat{T}_{avoid}$ can be estimated using the safety model below, while T_{verify} is based on known cryptographic speed in V2V hardware (we benchmark this on commercial devices in Section V).

Case 1: Rear-End Collision. For receiver i , let d_i be the distance from the sender, $v_{r,i}$ be the receiver speed, v_s be the sender speed, $a_{r,i}$ be the receiver deceleration (braking) capability, and a_s be the sender deceleration capability. Our model builds on fundamental kinematic equations such as

$$d = v_i t + \frac{1}{2} a t^2, \quad v_f = v_i + a t, \quad v_f^2 = v_i^2 + 2 a d,$$

where acceleration values are based on road conditions and braking capacity. Applying these equations allows us to express the initial distance d_i , adjusted for error (explained shortly), such that the trailing vehicle can stop safely:

$$d_i^{adj} > v_{r,i}^{adj} (T_{verify} + \tau_{cert}) + \frac{(v_{r,i}^{adj})^2}{2|a_{r,i}^{adj}|} - \frac{v_s^2}{2|a_s|} \quad (2)$$

Then, the maximum tolerable certificate delay is:

$$\tau_{cert,i}^{max} = \frac{d_i^{adj} + \frac{v_s^2}{2|a_s|} - \frac{(v_{r,i}^{adj})^2}{2|a_{r,i}^{adj}|}}{v_{r,i}^{adj}} - T_{verify}. \quad (3)$$

In (2)–(3), the receiver's values (d_i , $v_{r,i}$, and $a_{r,i}$) can be obtained from its BSMs; however, they may contain error, e.g., imprecision of d_i due to GPS positioning error. Accordingly, we use values adjusted for measurement error (denoted ϵ):

$$d_i^{adj} = d_i - \epsilon_{d,i} \quad v_{r,i}^{adj} = v_{r,i} + \epsilon_{v,i} \quad a_{r,i}^{adj} = a_{r,i} + \epsilon_{a,i}. \quad (4)$$

Each ϵ value represents an error bound on the corresponding distance, velocity, and acceleration estimate. The bound can be derived from known or empirically characterized error distributions (e.g., heavy precipitation or entering an urban canyon) and may change on the fly. For our evaluations, we assume representative standard deviations consistent with typical GPS and vehicle state-estimation performance. Concretely: $\epsilon_{d,i} = s\sigma_d$, $\epsilon_{v,i} = s\sigma_v$, $\epsilon_{a,i} = s\sigma_a$, where $s = 2$ for a 95% confidence interval under a Gaussian approximation of estimation error (capturing the dominant error, without overcorrecting for extreme outliers). These conservative adjustments ensure safety decisions are made on worst-case assumptions rather than perfect knowledge.

Case 2: Lane-Change/Merge Conflict. In this scenario, distance alone is insufficient to predict a conflict, as we must also consider the time until sender and receiver i occupy the same lane space. Omitting the full derivation (which follows from kinematic equations, as above), we can write

$$\tau_{cert,i}^{max} = \frac{\left(d_i^{adj} - d_{safe}\right) - \frac{(v_{r,i}^{adj})^2 - v_m^2}{2|a_{r,i}^{adj}|}}{v_{r,i}^{adj}} - T_{verify}, \quad (5)$$

where v_m is the safe velocity at the merge point and $d_{safe} = \frac{v_m^2}{2|a_s|}$ is the corresponding safe following distance.

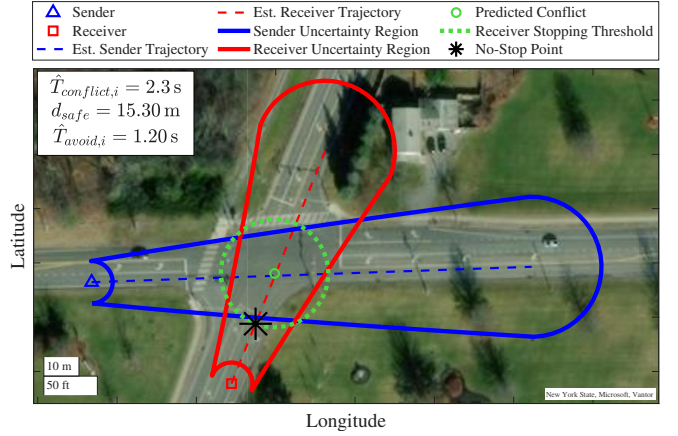


Fig. 4. Illustrative example of our safety model for an intersection conflict scenario. The sender (blue) can use information from a neighbor's (red) BSMs to predict a conflict, determine the time needed to avoid the conflict, and consequently deduce a bound on safety message and certificate delivery time.

Case 3: Intersection Conflict. For crossing lanes, we derive

$$\tau_{cert,i}^{max} = \frac{d_{entry} - \frac{(v_{r,i}^{adj})^2 - v_{final}^2}{2|a_{r,i}^{adj}|}}{v_{r,i}^{adj}} - T_{verify}, \quad (6)$$

where d_{entry} is the distance until i enters the conflict zone.

Example—Fig. 4 illustrates how our safety model works in an intersection scenario. Using verified BSMs from a neighbor (shown in red), the sender (blue) is able to predict its own and the neighbor's motion, with uncertainty based on (4) using typical GPS, speedometer, and accelerometer error [34]. The sender can then predict a future conflict, estimate the neighbor's braking distance, and determine the required stopping time (i.e., $\hat{T}_{avoid}$). In this case, the sender computes $\hat{T}_{conflict} = 2.3$ s and $\hat{T}_{avoid} = 1.2$ s. With $T_{verify} = 2.4$ ms based on hardware benchmarks described in Section V, they apply (1) to calculate the certificate communication bound as $\tau_{cert}^{max} = 1.0976$ s. In concrete terms, with an SPDU transmitted every 100 ms, this requires transmitting a complete certificate to this receiver spread over at most 10 SPDUs.

2) **Critical Receiver Set:** If we denote the set of all nearby receivers by $\mathcal{R}$, the subset of safety-critical receivers, $\mathcal{R}_{crit} \subseteq \mathcal{R}$, is constructed by selecting receivers in ascending order of $\tau_{cert,i}^{max}$ (i.e., the smallest delay margins) until:

$$|\mathcal{R}_{crit}| \geq P_{req} \cdot |\mathcal{R}|. \quad (7)$$

Here, $P_{req} \in [0, 1)$ denotes the desired fraction of receivers for which the safety condition should be satisfied. Ideally, the MCS would be optimized for all critical receivers; however, significant variation in link quality may cause a single receiver with a particularly poor, e.g., long-distance or non-line-of-sight (NLOS), channel to constrain the MCS for all receivers. Setting $P_{req} < 1$ mitigates the influence of such outliers when they do not represent the dominant safety risks.

3) **Sender-Side SNR Estimation:** Based on information from incoming BSMs, the sender obtains the error-adjusted location of nearby receivers and estimates their SNR. The SNR

TABLE I
RICIAN K -FACTOR ESTIMATION TABLE

Link Status	K -Factor estimate	Interpretation
NLOS	$K = 0$	Obstructed—revert to Rayleigh model.
line-of-sight (LOS) - Far	$K \approx 2$ to 5	Weak LOS path, high multipath effect.
LOS - Near	$K > 10$	Dominant LOS path.

at receiver i , prior to applying small-scale fading, is given by the link budget:

$$\hat{\gamma}_i = P_t + G_t + G_r - PL(d_i) - N. \quad (8)$$

where P_t is the transmit power, G_t and G_r are the transmitter and receiver antenna gains, and N denotes the noise power.

Let P_{out} denote the outage budget that constrains the probability of decoding failure. Under the assumption that the outage probabilities for fast and slow fading are independent, P_{out} can be approximated as

$$P_{out} = P_{out}^{fast} + P_{out}^{shadow} - P_{out}^{fast} P_{out}^{shadow} \approx P_{out}^{fast} + P_{out}^{shadow} \quad (9)$$

for $P_{out}^{fast}, P_{out}^{shadow} \ll 1$. With Rician fast-fading, the outage probability is a well-known expression [35]. Defining the fading margin as a function of outage probability, $M_f(P_{out}^{fast})$, the required fading margin (in dB) is given by

$$M_f(P_{out}^{fast}) = 10 \log_{10} \left(\frac{2(1+K)}{[Q_1^{-1}(\sqrt{2K}, 1 - P_{out}^{fast})]^2} \right) \quad (10)$$

where $Q_1^{-1}(\cdot, \cdot)$ is the inverse Marcum Q-function⁵ with respect to the second argument, and K is the Rician K -factor. We note that Doppler effects are not explicitly modeled here. In this setting, their primary impact is a reduction in channel coherence time, which is implicitly captured through the instantaneous fast-fading outage margin defined above.

For shadowing, we assume a Gaussian distribution with variance estimated from incoming signals [37], i.e., $X_s \sim \mathcal{N}(0, \sigma_s^2)$. The shadowing margin can be given as:

$$M_s(P_{out}^{shadow}) = -\sigma_s Q^{-1}(1 - P_{out}^{shadow}). \quad (11)$$

4) *MCS Selection for Critical Receivers*: The proposed safety-aware MCS selection integrates the above pieces. The instantaneous estimated SNR at receiver i is adjusted to $\hat{\gamma}_i^{adj} = \hat{\gamma}_i - M_s - M_f$, yielding a more reliable SNR estimate to reduce the outage probability $P(\hat{\gamma}_i < \gamma_{th}(m))$, where $\gamma_{th}(m)$ is the threshold to use MCS m . This adjustment also helps mitigate the impact of rapid decorrelation of V2V channels between state observation and transmission. Rather than assuming a stationary channel, this design enables a less aggressive MCS selection based on outage-constrained SNR estimates, improving resilience to short-term channel variation.

To predict if i can decode the transmission, we use:

$$p_i^{dec}(m) = \begin{cases} 1 - P_{out}, & \text{if } \hat{\gamma}_i^{adj} \geq \gamma_{th}(m) \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

⁵An integral involving a modified Bessel function, commonly used in error-probability calculations for fading channels [36].

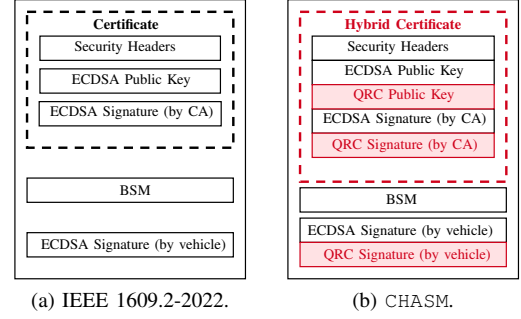


Fig. 5. Extension of classical SPDU security (a) to hybrid security (b) under CHASM. New elements to support quantum-resistant security are highlighted.

Then, the expected proportion of safety-critical receivers that successfully decode the transmission under MCS m is:

$$p_{\mathcal{R}}(m) = \frac{1}{|\mathcal{R}_{crit}|} \sum_{i \in \mathcal{R}_{crit}} p_i^{dec}(m). \quad (13)$$

Finally, the sender chooses the highest-rate MCS that maximizes the data rate $R(m)$, subject to the P_{req} threshold,

$$m^* = \arg \max_m R(m) \quad \text{s.t.} \quad p_{\mathcal{R}}(m) \geq P_{req}, \quad (14)$$

where m^* is the selected MCS.

5) *Parameter Selection*: As shadowing is spatially correlated in V2V channels, it varies slowly with distance and can be modeled in our experiments as the sliding-window standard deviation (σ_s) of a Gaussian log-scale residual from recent BSMs, after subtracting distance-dependent path loss [37]. In contrast, precisely determining the Rician K -factor in high-mobility, rich-scattering V2V is difficult. To address this, we leverage the fact that connected vehicles typically have on-board sensors that can assist (with quantifiable error) in inferring LOS conditions for a given receiver. This requires only a lightweight geometric obstruction query between the sender and reported receiver location using existing onboard perception data (e.g., bird's-eye view maps [38]), with indeterminate cases conservatively assumed to be NLOS. Detecting LOS allows distinguishing $K = 0$ (a NLOS Rayleigh link) from $K \geq 1$, which is crucial because $K : 0 \rightarrow 1$ represents the largest single-step change in channel quality as K increases. Moreover, C-V2X standards support only a few widely separated MCS options [7], suggesting that an exact K value is unnecessary. For example, estimating $K = 4$ when the true value is $K = 2$ results in an SNR error of approximately 2 dB, whereas the SNR thresholds for consecutive MCS levels (e.g., moving from QPSK to 16-QAM) differ by about 6–7 dB. Therefore, for $K \geq 1$, even a K -factor estimation error exceeding 100% is unlikely to cause the wrong MCS to be selected. Our experiments (Section V) show that coarse K estimation (Table I) is sufficient to maintain reliability and safety while increasing data rate.

E. APP Layer: Split-Agnostic Hybrid Certificate Distribution

At the application layer, CHASM fulfills two objectives: authenticating C-V2X messages using digital signatures, and

facilitating verification by efficiently distributing hybrid certificates. To accomplish these, we next describe (1) hybrid certificate construction and splitting, (2) the process of transmitting hybrid SPDUs, (3) receiving hybrid SPDUs, (4) determining how often to include certificate in SPDUs, and (5) ensuring certificates are delivered within a safety-relevance time bound.

1) *Hybrid Certificate Construction and Splitting*: The structure of today's ECDSA-secured SPDUs (Fig. 5(a)) can be straightforwardly extended to add a QRC key and signatures (Fig. 5(b)). However, the size of QRC keys and signatures prevents a certificate from fitting in one transmission (Fig. 1).

CHASM attacks this challenge by splitting each certificate and sharing it in pieces across a few transmissions (accelerated by the adaptive MCS scheme we introduced above). However, the specific splitting method may vary. Fragmentation is one option, similar to [13], but it requires a receiver to obtain all individual fragments before any BSMs can be verified and acted upon. This makes fragmentation-based schemes problematic in both poor communication environments, where packet loss can delay certificate reconstruction, and for BSMs received from an unseen vehicle, which cannot be verified until its certificate is fully reconstructed: a delayed delivery could postpone BSM verification long enough for safety to be compromised. We explicitly account for this delay in our formulations (provided shortly), but an alternative is desirable for particularly hostile wireless environments.

To that end, an erasure coding scheme (e.g., systematic Reed-Solomon) may be used instead of fragmentation—by generating a stream of coded certificate data rather than transmitting fixed fragments, each certificate-bearing SPDU carries a portion of coded certificate data. Receiving a minimum number of distinct coded symbols (rather than specific fragments) enables reconstruction of the original certificate. Erasure codes are thus less sensitive to packet loss, but at the expense of greater computational load on V2V equipment. CHASM therefore provides a generalized protocol for conveying fragmented, coded, or otherwise divided certificates between vehicles. In our experiments (Section V), we empirically compare the upsides and downsides of fragmentation and erasure code instantiations of CHASM.

2) *Transmitting Hybrid SPDUs*: Algorithm 1 defines how SPDUs are constructed and transmitted while a particular pseudonym certificate is in use (for roughly 5 minutes—see Section II). On certificate rotation, the algorithm restarts.

CHASM begins with loading parameters such as the required interval T_b between BSM transmissions (line 1), followed by loading the active certificate (line 2) and initializing a buffer storing the encoded certificate data, whether split into fragments or encoded using an erasure coding scheme (e.g., systematic Reed-Solomon). The transmitter then proceeds into the main execution loop (lines 4–18), which repeats every T_b seconds. At each transmission opportunity t , the transmitter polls the PHY layer for the current best, safety-aware MCS (line 11). Acquiring this information immediately before packet construction minimizes lag between observation and transmission, reducing sensitivity to rapid channel decorrelation. The maximum payload size is determined, and the available bandwidth for certificate data is denoted by

Algorithm 1 CHASM: Transmitting SPDUs

```

1:  $T_b \leftarrow$  BSM interval.
2:  $HC^{id} \leftarrow$  current certificate
3:  $mode \leftarrow$  Frag | EraseCode ▷ Select method of certificate splitting
4: while  $HC^{id}$  is active do
5:   if  $mode =$  Frag then
6:      $CertStream := HC^{id}$  as buffered queue of fragments.
7:   else
8:      $CertStream := HC^{id}$  as erasure-coded data stream.
9:   end if
10:  for each transmission at time  $t$ , spaced by  $T_b$  do
11:    Obtain adaptive  $m_t$  from PHY layer.
12:    Compute  $B_{cert}(m_t)$ 
13:     $C \leftarrow B_{cert}(m_t)$  bytes from  $CertStream$ .
14:     $Sig^c, Sig^{qr} \leftarrow$  signatures over  $M$  and coded data.
15:     $S \leftarrow$  SPDU containing  $M, C, Sig^c, Sig^{qr}$ .
16:    Transmit  $S$ 
17:  end for
18: end while

```

$B_{cert}(m)$. A fragment or amount of coded certificate data of at most $B_{cert}(m)$ bytes is then extracted from the certificate buffer (line 13). The BSM and accompanying certificate data are signed classically and with QRC (line 14), and an SPDU is constructed (line 15) and transmitted (line 16).

3) *Receiving Hybrid SPDUs*: When an unverified SPDU (denoted by S^*) is received, a vehicle executes the PROCESSPDU procedure described in Algorithm 2. First, the sender is identified (line 2) using the included certificate or hash thereof—see Section II. If that certificate is already known and verified, then S^* is immediately passed to VERIFYSPDU (line 5). Otherwise, S^* is cached for later verification; to avoid resource exhaustion, cached SPDUs that cannot be verified are discarded after a configurable timeout interval. The receiver also extracts any certificate data contained in S^* (lines 9–33), using either fragmentation (lines 10–19) or erasure coding (lines 21–30). For a particular hybrid profile a —a distinct combination of classical and quantum-resistant cryptography, e.g., ECDSA and FN-DSA-512—let $L_{cert}(a)$ denote the total size (in bytes) of the sender's active hybrid certificate (Fig. 5(b)), and let CertBuffer denote the total amount of useful certificate data accumulated for sender i . After each update, the receiver checks if it has enough data—all fragments, or sufficient coded data—to reconstruct the certificate. If so, the reconstructed certificate is verified with VERIFYCERTIFICATE. Upon success, all cached SPDUs associated with that certificate are processed.

4) *Certificate Completion Under Delay Constraint*: In CHASM, we connect safety risk with how long a receiver may remain unable to verify BSMs, and hence how quickly a complete certificate needs to be shared, and in turn, with how frequently certificate data should be included in outgoing SPDUs. By explicitly incorporating the probability of packet loss and the number of fragments or erasure-coded symbols required to convey a complete certificate—which directly yields the associated delay interval—we ensure that calculated safety margins account for delays in certificate transmission, reconstruction, and verification. For brevity, we only detail the procedure for the erasure-coding mode, but similar logic applies when using fragmentation.

As above, let $L_{cert}(a)$ denote the total size of the certificate.

Algorithm 2 CHASM: Receiving and Verifying SPDUs

```

1: procedure PROCESSSPDU( $S^*$ )
2:    $id \leftarrow$  extract vehicle ID from  $S^*$ 
3:   if  $id$  matches a known and previously verified certificate then
4:      $HC^{id} \leftarrow$  load known certificate
5:      $VERIFYSPDU(HC^{id}, S^*)$ 
6:   else
7:     Load or create new CertBuffer for sender  $id$ .
8:      $C \leftarrow$  extract certificate data from  $S^*$ 
9:     if  $mode = \text{Frag}$  and  $C$  is a new (unseen) fragment then
10:       $CertBuffer = CertBuffer + C$ 
11:      Cache  $S^*$  for later verification
12:      if  $CertBuffer$  contains all expected fragments then
13:         $HC^{id} \leftarrow$  reassemble all fragments
14:         $r \leftarrow VERIFYCERTIFICATE(HC^{id})$ 
15:        if  $r == \text{True}$  then
16:          Record  $HC^{id}$  as known and valid for  $id$ 
17:           $VerifyCachedSPDUs(HC^{id})$ 
18:        end if
19:      end if
20:    else if  $mode = \text{EraseCode}$  and  $C$  is new (distinct) data then
21:       $CertBuffer = CertBuffer + C$ 
22:      Cache  $S^*$  for later verification
23:      if  $|CertBuffer| \geq L_{cert}(a)$  then
24:         $HC^{id} \leftarrow$  decode accumulated coded data
25:         $r \leftarrow VERIFYCERTIFICATE(HC^{id})$ 
26:        if  $r == \text{True}$  then
27:          Record  $HC^{id}$  as known and valid for  $id$ 
28:           $VerifyCachedSPDUs(HC^{id})$ 
29:        end if
30:      end if
31:    else  $\triangleright$  if unknown mode, or certificate data is not new
32:      return
33:    end if
34:  end if
35: end procedure

36: procedure  $VERIFYSPDU(HC^{id}, S^*)$ 
37:    $(pk_{id}^c, pk_{id}^{qr}) \leftarrow$  extract public keys from  $HC^{id}$ 
38:   Verify signatures of  $S^*$  using  $(pk_{id}^c, pk_{id}^{qr})$ 
39:   if Signatures verify successfully then  $\triangleright$  Phase-dependent (Fig. 2).
40:     if freshness check on timestamp on  $S^*$  passes then
41:       Accept  $S^*$ 
42:     else
43:       Reject  $S^*$ 
44:     end if
45:   else
46:     Reject  $S^*$ 
47:   end if
48: end procedure

49: procedure  $VERIFYCERTIFICATE(HC^{id})$ 
50:   Verify certificate using PKI chain as per IEEE 1609.2 [9]
51:   if verification is successful then
52:     return True
53:   else
54:     return False
55:   end if
56: end procedure

57: procedure  $VERIFYCACHEDSPDUs(HC^{id})$ 
58:   for each cached  $S^*$  from  $id$  do
59:      $VERIFYSPDU(HC^{id}, S^*)$ 
60:   end for
61: end procedure

```

Then, let $B_{cert}(m)$ denote the usable certificate payload carried in one certificate-bearing SPDU under an MCS m . As channel conditions vary over time, a certificate-bearing transmission at time t is associated with the active MCS m_t , yielding payload $B_{cert}(m_t)$. For receiver i , define the

accumulated useful certificate data as:

$$|CertBuffer| = \sum_{t=1}^{N_{opp,i}} Z_{i,t} B_{cert}(m_t) \quad (15)$$

where $Z_{i,t} \in \{0, 1\}$ indicates whether transmission t is successfully received, i.e., successfully decoded at the PHY layer, and N_{opp} indicates the number of opportunities to decode certificate data from incoming SPDUs (elaborated shortly—see (22)). We assume the use of erasure coding (e.g., systematic Reed-Solomon), such that each successfully received transmission contributes new (i.e., linearly independent) information toward certificate reconstruction with high probability. A certificate is fully received when:

$$|CertBuffer| \geq L_{cert}(a). \quad (16)$$

As required by current standards, a vehicle must transmit an SPDU containing a BSM every $T_b = 0.1$ s. Certificate information needs to be distributed across multiple certificate-bearing SPDUs that occur every k -th BSM, which lets us write the certificate-bearing SPDU interval as $T_c = kT_b$, where $k \in \{1, 2, 3, \dots\}$. Let $\bar{B}_{cert} = \mathbb{E}[B_{cert}(m_t)]$ denote the expected usable certificate payload per transmission under the adaptive MCS policy. Then, for an arbitrary receiver i , the nominal certificate completion time is approximated as:

$$T_{comp,nom,i} = kT_b \left\lceil \frac{L_{cert}(a)}{\bar{B}_{cert}} \right\rceil. \quad (17)$$

This lets us formulate a first-order certificate delay model—separately address reliability below—as:

$$\tau_{cert,i} \approx T_{wait,i} + T_{comp,nom,i}. \quad (18)$$

Assuming the time waiting to send the first certificate-bearing SPDU is $T_{wait,i} \sim \text{Uniform}(0, T_b)$, we can derive:

$$\tau_{cert,i} \approx \frac{T_b}{2} + kT_b \left\lceil \frac{L_{cert}(a)}{\bar{B}_{cert}} \right\rceil. \quad (19)$$

This approximation replaces the time-varying payload $B_{cert}(m_t, a)$ with its expectation $\bar{B}_{cert}$ to obtain a tractable first-order delay model. Note that (19) combines with (1), (3), and (5)–(6) to directly connect certificate delay to safety.

To meet the certificate communication delay bound $\tau_{cert,i}^{max}$, certificate delay must satisfy

$$\frac{T_b}{2} + kT_b \left\lceil \frac{L_{cert}(a)}{\bar{B}_{cert}} \right\rceil \leq \tau_{cert,i}^{max}. \quad (20)$$

This allows us to bound k as:

$$k \leq \left\lceil \frac{\tau_{cert,i}^{max} - \frac{T_b}{2}}{T_b \left\lceil \frac{L_{cert}(a)}{\bar{B}_{cert}} \right\rceil} \right\rceil. \quad (21)$$

Example: Consider the scenario from Fig. 4, where applying (1) yields $\tau_{cert,i}^{max} = 1.0976$ s. Using a hybrid profile with ECDSA P.256 and FN-DSA-512, we determine that $L_{cert}(a) = 1691$ bytes. Given the line of sight between sender and receiver, a high MCS could be selected, such as 16-QAM at a 0.3 code rate. Then, in a typical 20 MHz channel, the maximum transmission size would be 1,764 bytes [39], yielding $\bar{B}_{cert} = 1,114$ bytes. With $T_{verify} = 1.53 \times 10^{-3}$ s

based on hardware benchmarks (see Section V), plugging everything into (21) gives us:

$$k \leq \left\lfloor \frac{1.0976 - \frac{0.1}{2}}{0.1 \left\lceil \frac{1691}{1114} \right\rceil} \right\rfloor \rightarrow k \leq 5.$$

That is, if every packet was sure to be delivered, then the sender could attach certificate data to every fifth SPDU and communicate the full certificate to the receiver before a critical safety risk would occur. However, in a real system, this ideal rate is insufficient—we must also consider *reliability*.

5) *Certificate Completion Reliability*: Recall that $\tau_{cert,i}^{max}$ denotes the maximum time to distribute a complete certificate before receiver i becomes safety-critical. The number of opportunities to transmit certificate data before time expires is:

$$N_{opp,i} = \left\lfloor \frac{\tau_{cert,i}^{max}}{kT_b} \right\rfloor. \quad (22)$$

Let $q_i(m_t)$ denote the probability that a certificate-bearing transmission at time t is successfully received by receiver i (in a similar manner to (12)). We define the expected success probability under the adaptive MCS policy as $\bar{q}_i = \mathbb{E}[q_i(m_t)]$, and further let

$$N_{req} = \left\lceil \frac{L_{cert}(a)}{\bar{B}_{cert}} \right\rceil \quad (23)$$

denote the nominal number of successful transmissions required to reconstruct the certificate. Since the channel coherence time is much less than the transmission interval, each attempt can be treated as an independent trial, allowing us to estimate the number of successful certificate-bearing transmissions at receiver i as $Y_i \sim \text{Binomial}(N_{opp,i}, \bar{q}_i)$.

When using erasure codes, each successfully received transmission contributes useful information toward certificate reconstruction, enabling a cumulative success model. Accordingly, the certificate completion probability for receiver i is

$$P_{comp,i} = \sum_{y=N_{req}}^{N_{opp,i}} \binom{N_{opp,i}}{y} \bar{q}_i^y (1 - \bar{q}_i)^{N_{opp,i}-y}, \quad (24)$$

where the system must combine with (13) to satisfy

$$P_{comp,i} \geq 1 - \epsilon_c, \quad \forall i \in p_{\mathcal{R}}(m_t). \quad (25)$$

Here, ϵ_c denotes the maximum tolerable probability that receiver i fails to complete certificate reconstruction before becoming safety-critical.

Example (cont.): This calculation is important to calibrate the calculation of k from the previous step. Extending the example from (21), which determined $k = 5$, we can use (22)–(24) to determine that the probability of successfully delivering the certificate to the receiver in time is $P_{comp} = 0.73$, which may not be considered sufficient. However, correcting to $k = 2$ yields $P_{comp} = 0.99$, a much more acceptable outcome for safety. Thus, depending on the choice of ϵ_c , k can be adjusted to make delivery within the safety bound more certain.

IV. FORMAL SECURITY ANALYSIS

We formally analyze CHASM's security by reducing it to the EUF-CMA unforgeability of its underlying digital signature algorithms. Superscripts c and q respectively denote classical and QRC elements or functions. We begin with the following standard definition of a digital signature scheme:

Definition 1 (Digital Signature Scheme). A digital signature scheme Σ is a tuple of polynomial-time algorithms $\Sigma = (\text{KG}, \text{SG}, \text{V})$ where:

- $(\text{sk}, \text{pk}) \leftarrow \text{KG}(1^\lambda)$: **key generation** takes security parameter λ , returns secret and public key pair (sk, pk) .
- $\text{Sig} \leftarrow \text{SG}(\text{sk}, m)$: **signature generation** takes secret key sk and message m , and returns signature Sig .
- $r \leftarrow \text{V}(\text{pk}, m, \text{Sig})$: **signature verification** takes public key pk , message m , signature Sig , and returns $r \in \{0, 1\}$, where 1 is verification success and 0 is failure.

The applicable reduction depends on CHASM's operational phase (recall Fig. 2). In the preparation phase, where QRC is not yet mandatory, forging a classical signature on a hybrid SPDU can induce a legacy verifier to accept a bogus SPDU. Accordingly, CHASM's preparation-phase security reduces to the EUF-CMA unforgeability of its classical signature algorithm. Conversely, in the strictly quantum-resistant phase, ECDSA is disallowed, and CHASM's security reduces to the EUF-CMA unforgeability of its QRC signature algorithm. Before presenting these reductions, we first establish that certificate splitting preserves authentication.

Definition 2 (Certificate Authenticity Under Splitting). We define the certificate authenticity under splitting experiment, denoted by $\text{Expt}_{\text{CAUS}}(\mathcal{A})$, as follows. The challenger generates CA keypairs $(\text{pk}_{\text{CA}}^c, \text{sk}_{\text{CA}}^c)$ and $(\text{pk}_{\text{CA}}^q, \text{sk}_{\text{CA}}^q)$, and then supplies the public keys to a probabilistic polynomial-time (PPT) or quantum polynomial-time (QPT) adversary $\mathcal{A}$. We assume $\mathcal{A}$ has access to query two signing oracles $\mathcal{O}_{\text{sign}}^c(\mathcal{C})$ and $\mathcal{O}_{\text{sign}}^q(\mathcal{C})$ with an arbitrary certificate $\mathcal{C}$. Let $\mathcal{Q}$ be the set of all certificates queried to these oracles. The oracles operate in a black-box manner and respectively return signatures Sig^c and Sig^q over the queried $\mathcal{C}$ using the appropriate secret key (which $\mathcal{A}$ does not have access to). The experiment then proceeds:

- 1) $\mathcal{A}$ outputs the arbitrarily divided and/or encoded bytes of a hybrid certificate and supplies them to the verifier.
- 2) The verifier reassembles and/or decodes the bytes into a parsed certificate payload $\mathcal{C}^*$, and the accompanying signatures Sig^{c*} and Sig^{q*} that are claimed as produced by a CA over $\mathcal{C}^*$.

In the preparation phase, a classical (PPT) adversary $\mathcal{A}^c$ wins if $\mathcal{C}^* \notin \mathcal{Q}$ and $\text{V}^c(\text{pk}_{\text{CA}}^c, \mathcal{C}^*, \text{Sig}^{c*}) = 1$. Similarly, in the strictly quantum-resistant phase, a QPT adversary $\mathcal{A}^q$ wins if $\mathcal{C}^* \notin \mathcal{Q}$ and $\text{V}^q(\text{pk}_{\text{CA}}^q, \mathcal{C}^*, \text{Sig}^{q*}) = 1$. The advantage of either PPT or QPT $\mathcal{A}$ is: $\text{Adv}_{\text{CAUS}}^{\mathcal{A}}(\lambda) = \Pr[\text{Expt}_{\text{CAUS}}^{\mathcal{A}} = 1]$.

Theorem 1 (Certificate Splitting Preserves Unforgeability). If both the classical and quantum-resistant signature schemes used in CHASM are EUF-CMA unforgeable, then neither a PPT nor QPT adversary $\mathcal{A}$ can win $\text{Expt}_{\text{CAUS}}^{\mathcal{A}}$ with non-negligible advantage, i.e., $\text{Adv}_{\text{CAUS}}^{\mathcal{A}}(\lambda) = \text{negl}(\lambda)$.

Proof. Let CHASM be instantiated with classical and QRC signature algorithms that are EUF-CMA unforgeable. Then, assume for a contradiction that there exists a PPT or QPT adversary $\mathcal{A}$ who can win $\text{Expt}_{\text{CAUS}}$ with non-negligible advantage; i.e., where $\text{Adv}_{\text{CAUS}}^{\mathcal{A}}(\lambda) \geq \epsilon(\lambda)$ for a non-negligible function ϵ and sufficiently large security parameter λ . Depending on the threat model, we can construct either a classical forger $\mathcal{B}_1$ or quantum forger $\mathcal{B}_2$, each of which attacks one algorithm and honestly simulates the other.

For $\mathcal{B}_1$, we use $\mathcal{A}$ to perform the $\text{Expt}_{\text{CAUS}}$ experiment as per Definition 2 for the preparation phase. The classical public key pk_{CA}^* is obtained from the challenger, and we embed this as the CA's public key. The adversary $\mathcal{A}$ can query the classical signing oracle (which $\mathcal{B}_1$ routes to its challenger), while quantum-resistant queries are answered honestly by the simulator $\mathcal{B}_1$ (who knows the independently generated quantum-resistant secret key sk_{CA}^q). Eventually, because $\mathcal{A}$ is successful, $\mathcal{B}_1$ extracts and outputs the certificate payload C^* and classical signature Sig^{c*} , where neither the certificate payload nor signature appear in the signing oracle's transcript (i.e., a valid signature for a previously unseen certificate). Because that signature verifies with pk_{CA}^* , $\mathcal{B}_1$ violates the EUF-CMA unforgeability of the classical algorithm, a contradiction.

We can similarly construct the quantum forger $\mathcal{B}_2$ for the strictly quantum-resistant era by using $\mathcal{A}$ to conduct the experiment in the same manner as for $\mathcal{B}_1$, except using quantum-resistant keys and signatures obtained from a quantum challenger. Because $\mathcal{A}$ is successful, $\mathcal{B}_2$ produces the certificate payload C^* and a valid signature Sig^{q*} over the previously unseen certificate payload, which violates the EUF-CMA unforgeability of the QRC algorithm—a contradiction.

Success in the preparation phase requires the PPT adversary to violate the EUF-CMA unforgeability of the classical algorithm, while success in the strictly quantum-resistant phase requires the QPT adversary to break the EUF-CMA unforgeability of the QRC algorithm. In either case, this is a contradiction; therefore, no such adversaries can exist. $\square$

Next, we formally reduce the security of CHASM to the EUF-CMA unforgeability of its digital signature algorithms.

Definition 3 (Classical EUF-CMA Unforgeability). We define the *classical unforgeability experiment*, denoted by $\text{Expt}_{\text{eUF,c}}$, in which the challenger generates a classical keypair and a classical PPT adversary $\mathcal{A}$ is then provided with the public (but not secret) key, denoted by pk^c . We assume adversary $\mathcal{A}$ has access to query a signing oracle $\mathcal{O}_{\text{sign}}^c(m)$ with an arbitrary message m . The oracle operates in a black-box manner and returns the signature Sig^c over m using the classical secret key (which $\mathcal{A}$ cannot access). $\mathcal{A}$ wins if it outputs a pair (m^*, Sig^{c*}) , where m^* was never previously queried to the signing oracle, such that: $V^c(\text{pk}^c, m^*, \text{Sig}^{c*}) = 1$. We define the advantage of $\mathcal{A}$ as:

$$\text{Adv}_{\text{Expt}_{\text{eUF,c}}}^{\mathcal{A}}(\lambda) = \Pr[\text{Expt}_{\text{eUF,c}}^{\mathcal{A}} = 1].$$

Definition 4 (Quantum EUF-CMA Unforgeability). We define the *quantum unforgeability experiment*, denoted by $\text{Expt}_{\text{eUF,q}}$, in which the challenger generates a QRC keypair and a QPT adversary $\mathcal{A}$ is provided with the public (but not secret) key,

denoted by pk^q . We assume adversary $\mathcal{A}$ has access to query a signing oracle $\mathcal{O}_{\text{sign}}^q(m)$ with an arbitrary message m . The oracle operates in a black-box manner and returns the signature Sig^q over m using the QRC secret key (which $\mathcal{A}$ cannot access). $\mathcal{A}$ wins if it outputs a pair (m^*, Sig^{q*}) , where m^* was never previously queried to the signing oracle, such that: $V^q(\text{pk}^q, m^*, \text{Sig}^{q*}) = 1$. We define the advantage of $\mathcal{A}$ as:

$$\text{Adv}_{\text{Expt}_{\text{eUF,q}}}^{\mathcal{A}}(\lambda) = \Pr[\text{Expt}_{\text{eUF,q}}^{\mathcal{A}} = 1].$$

Theorem 2 (Message Unforgeability—Classical Adversary). If CHASM's classical and QRC signature algorithms are EUF-CMA unforgeable, then BSMs under CHASM are EUF-CMA unforgeable against a classical adversary $\mathcal{A}^c$. The advantage of $\mathcal{A}^c$ against CHASM is bounded by:

$$\text{Adv}_{\text{CHASM}}^{\mathcal{A}^c}(\lambda) \leq 2 \cdot \max\left(\text{Adv}_{\text{Expt}_{\text{eUF,c}}}^{\mathcal{A}^c}(\lambda), \text{Adv}_{\text{Expt}_{\text{eUF,q}}}^{\mathcal{A}^c}(\lambda)\right).$$

Proof. Let $\mathcal{A}^c$ be a PPT adversary with access to classical computing resources. We construct a simulator $\mathcal{B}$ that uses $\mathcal{A}^c$ to break the EUF-CMA unforgeability of one of the underlying schemes. $\mathcal{B}$ flips a fair coin $b \in \{0, 1\}$ to guess which signature $\mathcal{A}^c$ will successfully forge (0 for classical, 1 for quantum-resistant). $\mathcal{B}$ receives the challenge public key for scheme b from its own challenger, honestly generates the keypair for the other scheme, and provides both public keys to $\mathcal{A}^c$. When $\mathcal{A}^c$ makes a signing query for a message m , $\mathcal{B}$ forwards m to its challenger to get the b -scheme signature, uses its known secret key to generate the signature for the other scheme, and returns both to $\mathcal{A}^c$. Eventually, $\mathcal{A}^c$ outputs a forged message and hybrid signature pair. With probability at least $1/2$, $\mathcal{B}$ guessed correctly (i.e., $\mathcal{A}^c$'s valid forgery corresponds to scheme b). $\mathcal{B}$ extracts this signature and submits it to its own challenger. Because $\mathcal{A}^c$ produces a valid, fresh forgery with probability ϵ , $\mathcal{B}$ wins the underlying EUF-CMA game—either $\text{Expt}_{\text{eUF,c}}$ or $\text{Expt}_{\text{eUF,q}}$ —with non-negligible advantage $\epsilon/2$. This directly contradicts the assumption that both underlying schemes are EUF-CMA unforgeable. $\square$

Corollary 2.1 (Message Unforgeability—Quantum Adversary). In the strictly quantum-resistant phase, we assume a quantum-capable adversary $\mathcal{A}^q$. Consequently, classical algorithms (e.g., ECDSA) provide no security (i.e., $\text{Adv}_{\text{eUF,c}}^{\mathcal{A}^q}(\lambda) \approx 1$) and are disallowed. Therefore, CHASM's security in the strictly quantum-resistant era reduces to the EUF-CMA unforgeability of its QRC algorithm:

$$\text{Adv}_{\text{CHASM}}^{\mathcal{A}^q}(\lambda) \leq \text{Adv}_{\text{eUF,q}}^{\mathcal{A}^q}(\lambda).$$

Proof. Follows directly from Theorem 2 where $b = 1$ is fixed. $\square$

Theorem 3 (CHASM is Secure Against Replay Attacks). In the sense of IEEE 1609.2, CHASM is protected against message replay attacks by classical and quantum-capable adversaries.

Proof. Under IEEE 1609.2, every BSM (denoted by M) must contain a timestamp that records when the BSM was generated (denoted by t_g), and a freshness check is performed on all received messages to guard against replays (see line 40 of

Algorithm 2). Recall that by Theorem 2 and Corollary 2.1, neither a classical nor quantum-capable adversary can modify M and retain or produce a valid pair of digital signatures. Therefore, since $t_g \in M$, any modification of t_g to evade the freshness check will result in signature verification failure and cause M to be discarded. Similarly, modifying the message while retaining a valid timestamp will cause verification failure. We thus conclude that as long as IEEE 1609.2 freshness checks are enforced, CHASM rejects replayed messages outside the permitted freshness interval. $\square$

Discussion and Limitations

Message Forgery—Reducing the security of CHASM to the EUF-CMA unforgeability of its underlying signature algorithms shows that CHASM inherits their resistance to message spoofing and forgery, fulfilling the first two design and security criteria from Section III. We note that while QRC algorithms such as ML-DSA are strongly EUF-CMA unforgeable [23], some algorithms (e.g., FN-DSA) may only prevent forging new signatures on previously unsigned messages [40]. This is often sufficient for V2V, since signing a previously signed message (prevented by EUF-CMA) does not change its semantics or utility for safety. Our proofs continue to hold for this assumption with only minor modifications, emphasizing that the security of CHASM reduces to the security of its signature algorithms.

Downgrade Attacks—Since CHASM prohibits using vulnerable algorithms—e.g., forbidding ECDSA after Q-Day—an adversary cannot induce a compliant device to accept a signature under a vulnerable algorithm. Before Q-Day, ECDSA will remain secure against classical attackers; after Q-Day, the use of QRC and the disallowance of ECDSA precludes downgrading from quantum-resistant authentication to vulnerable ECDSA in the post-quantum era.

Replay Attacks—CHASM resists message replay attacks outside of a defined freshness interval, as shown in Theorem 3.

V. EVALUATION

We experimentally evaluate CHASM to demonstrate its light computational load and safety and security advantages.

A. Experimental Setup

Hardware Experiments. We benchmark our implementation on state-of-the-art Cohda MK6 on-board units (OBUs) with i.MX8 ARM-based automotive processors [41], implementing CHASM and relevant IEEE and SAE standards (e.g., [9], [11]) using standard C++ libraries and proprietary APIs from Cohda Wireless. We use OpenSSL for ECDSA (with the P.256 elliptic curve specified by IEEE 1609.2), and `liboqs` [42] for NIST-compliant QRC.

Simulation Software. To evaluate CHASM at scale, we use NS-3 with the 5G C-V2X module from [43], which implements 3GPP C-V2X channel models to ensure our simulations reflect state-of-the-art V2V propagation and sidelink behavior. 3D building and vehicle geometry (required for predicting LOS in our fading model) is drawn from OpenStreetMap [44].

TABLE II
CHASM PERFORMANCE LOAD ON COHDA MK6 OBU

Configuration	Metric	Sending	vs. 1609.2	Receiving	vs. 1609.2
IEEE 1609.2	Peak CPU Usage	5.2%	—	4.0%	—
	Peak Memory Usage	3.3%	—	3.3%	—
	SPDU total sign time	$\mu = 0.518$ ms $\sigma = 0.018$ ms	—	—	—
	T_{verify}	—	—	1.2 ms	—
	Per-Message Delay	0.518 ms	—	1.2 ms	—
CHASM: ECDSA + FN-DSA-512	Peak CPU Usage	5.1%	−0.1%	4.3%	+0.3%
	Peak Memory Usage	8.9%	+5.6%	3.7%	+0.4%
	SPDU total sign time	$\mu = 3.518$ ms $\sigma = 0.213$ ms	+3.0 ms	—	—
	T_{verify}	—	—	$\mu = 0.446$ ms, $\sigma = 0.052$ ms	−0.754 ms
	Systematic Reed-Solomon	$\mu = 0.313$ ms $\sigma = 0.110$ ms	+0.313 ms	$\mu = 0.523$ ms $\sigma = 0.110$ ms	+0.523 ms
CHASM: ECDSA + ML-DSA-44	Per-Message Delay	3.831 ms	+3.313 ms	0.969 ms	−0.231 ms
	Peak CPU Usage	6.3%	+1.1%	8.9%	+4.9%
	Peak Memory Usage	3.9%	+0.6%	4.8%	+1.5%
	SPDU total sign time	$\mu = 3.152$ ms $\sigma = 1.01$ ms	+2.634 ms	—	—
	T_{verify}	—	—	$\mu = 0.254$ ms $\sigma = 0.011$ ms	−0.946 ms
CHASM: ECDSA + ML-DSA-44	Systematic Reed-Solomon	$\mu = 0.329$ ms $\sigma = 0.162$ ms	+0.329 ms	$\mu = 0.327$ ms $\sigma = 0.233$ ms	+0.327 ms
	Per-Message Delay	3.481 ms	+2.963 ms	0.581 ms	−0.619 ms

Vehicle traces are drawn from SUMO traffic simulations [45], geospatially reconciled with OpenStreetMap data. Parameters, including sources of error, are summarized in Table III.

B. COTS Compatibility—Computational Requirements

Table II gives our benchmarking results on commercial C-V2X hardware. We compare today's IEEE 1609.2 security against CHASM, instantiated with two QRC algorithms (FN-DSA-512 and ML-DSA-44), to demonstrate that CHASM imposes only modest computational requirements. In particular, peak CPU utilization increases by $\leq 6\%$ in both CHASM instances, while memory utilization increases by $< 9\%$. The added delay for signing and transmitting SPDUs is minimal (< 3.9 ms once every $T_b = 100$ ms), and verifying hybrid SPDUs increases processing time by an average of just 0.969 ms with FN-DSA and 0.581 ms with ML-DSA. We also note that erasure coding is very efficient on the OBUs, requiring < 0.6 ms per encoding or decoding operation (aided by the ARM NEON instruction set for vectorized arithmetic).

C. Certificate Distribution, BSM Verification, and Safety

We now compare various methods of integrating QRC into C-V2X. Our baseline is today's IEEE 1609.2, where ECDSA is universal, modulation is per [11], and a certificate is shared every 500 ms. We compare this against the approach of the closest related work [13], as well as against CHASM instantiated with various combinations of deterministic vs. safety-aware adaptive modulation, fragmentation vs. erasure coding, and different choices of QRC (see Table IV). For these studies, our results are based on 60-minute city-scale simulations of Buffalo, NY, a major U.S. metropolitan area with a variety of urban, semi-rural, and suburban roadways where vehicle speeds, densities, etc., vary widely, making for an excellent and diverse test environment. Our result set is based on intersection, rear-end, and lane-merging safety events recorded during simulations, with more than 65,000 events logged for each tested configuration. An “event” begins when a potential violation of the margins defined in our safety model is detected (Section III) and ends when the corresponding

TABLE III
PARAMETERS FOR CITY-SCALE SIMULATOR EVALUATION

	Parameter	Value
Simulation	Duration	60 minutes
	Transmit Power	20 dBm
	Noise floor	-95 dBm
	Center Frequency	5.92 GHz
	Tx/Rx Antenna Gains	4 dBi each
	Shadowing estimation error	$\mathcal{N}(\mu = 0, \sigma = 0.1)$, in dB
	Path loss model $PL(d)$	3GPP NR-V2X Model
Vehicle	Sensor LOS detection error	0.05
	GPS Error (σ_d)	3 m
	Speedometer Error (σ_v)	1.34 m/s
	Accelerometer Error (σ_a)	0.02 m/s ²
C-V2X	Generation	5G NR-V2X
	MCS options	As per [7, Table 5.1.3.2-1]
	Bandwidth / Subchannels / Resource Blocks	20 MHz / 10 / 96
	Rician K -Factor	Estimated per Table I
CHASM	$p_{out}^{fast}, p_{out}^{shadow}$	0.05 each
	P_{req}	0.9
	T_{verify}	As per Table II for different instantiations
Regional Parameters		Rural Suburban Urban
	Vehicle speed distribution (kph)	$\mathcal{N}(80, 5)$ $\mathcal{N}(50, 5)$ $\mathcal{N}(30, 8)$
	Median vehicle density (veh/km ²), selected per [46]	20 80 150

hazard no longer exists. For example, an intersection conflict event begins when two vehicles approach the same intersection from different directions and may collide in the intersection without appropriate speed and spacing, and the event ends once at least one vehicle has cleared the intersection. Due to the mutually exclusive nature of our defined safety scenarios (intersection crossing, rear-end following, and lane-merging), we assume that a distinct pair of vehicles can generate only one event at a time.

Metrics—We examine how often safety requirements are violated, based on our safety model from Section III, as a fraction of total safety events:

$$\text{Violation Frac.} = \frac{\text{Number of violations}}{\text{Number of events}}.$$

We also consider the rate of successfully completing certificate distribution to safety-critical receivers within the defined safety bound:

$$\text{Cert. Completion} = \frac{\text{Number of Successful Completions}}{\text{Number of Safety-Critical Receivers}}.$$

In Table IV, we report these metrics averaged over all events, with 95% confidence intervals. Further, we examine the percentage of safety-critical receivers for which the certificate delay bound from (1) is met (Fig. 6). An analysis of failure cases under each tested configuration is given in Fig. 7. Rather than evaluating components such as MCS selection or SNR estimation in isolation, we evaluate their end-to-end impact through safety-conditioned violation rates. This better reflects the cross-layer design and operational objectives of CHASM.

Results—For our baseline IEEE 1609.2 configuration, we find that safety margins are violated 24.13% of the time and certificate distribution is only completed quickly enough to avoid a conflict 80.58% of the time. This is slightly unexpected, but makes sense when considering the deterministic and heuristic nature of current standards—e.g., Fig. 7 shows that most failures of the baseline case come from transmitting a certificate over a bad channel, since these attempts are fixed and periodic as well as blind to channel quality. The approach

TABLE IV
EXPERIMENTAL CONFIGURATIONS AND RESULTS BY CHOICE OF QRC, CERTIFICATE SPLIT (FR = FRAGMENTATION, EC = ERASURE CODE), AND MCS (D = DETERMINISTIC, A = ADAPTIVE).

Config	Protocol	QRC	Certificate Split	MCS	Violation Frac. (95% CI)	Cert. Completion (95% CI)
1	Baseline	N/A	N/A	D	24.13±0.33%	80.58±0.29%
2	[13]	FN-DSA	FR	D	12.06±0.25%	89.62±0.23%
3	[13]	ML-DSA	FR	D	63.49±0.38%	38.02±0.38%
4	CHASM	FN-DSA	FR	D	5.50±0.18%	95.63±0.15%
5	CHASM	FN-DSA	EC	D	3.98±0.15%	97.54±0.11%
6	CHASM	FN-DSA	FR	A	3.50±0.14%	97.80±0.10%
7	CHASM	FN-DSA	EC	A	1.94±0.11%	98.86±0.07%
8	CHASM	ML-DSA	FR	D	12.39±0.26%	90.22±0.22%
9	CHASM	ML-DSA	EC	D	9.19±0.23%	93.52±0.18%
10	CHASM	ML-DSA	FR	A	14.69±0.28%	87.35±0.25%
11	CHASM	ML-DSA	EC	A	11.68±0.25%	88.68±0.25%

from [13] performs slightly better than the baseline when using FN-DSA, but still fails to meet safety requirements 12.06% of the time. Moreover, it performs very poorly under the overhead of ML-DSA, violating safety bounds in 63.49% of instances.

In contrast to the baseline and related work, CHASM has a safety violation rate $\leq 14.7\%$ regardless of instantiation, and $\leq 5.5\%$ when using FN-DSA. Configuration 9 performs slightly better than other ML-DSA configurations—we attribute this to its use of erasure coding, which accommodates the overhead of ML-DSA better than fragmentation. It is also notable that ML-DSA with adaptive modulation performs slightly worse than with deterministic MCS selection. This is an expected consequence of attempting to accelerate certificate sharing with especially high-overhead ML-DSA, which forces attempting more aggressive MCS more often (sometimes when the channel is poor) to meet the safety delay requirements. However, all ML-DSA instantiations of CHASM outperform IEEE 1609.2, and they are comparable to or better than [13], validating our advantage. Most importantly, CHASM's best performance is under configuration 7, which violates safety margins in just 1.94% of instances and delivers certificates quickly enough in 98.86% of attempts. This shows that safety and security are jointly maximized only when all aspects of our cross-layer solution—erasure coding, adaptive modulation, and quantum-resistant authentication—are jointly employed.

For coverage of safety-critical receivers, we extrapolate from current standards to set a target that certificates should be successfully delivered to high-risk neighbors at least 90% of the time (see [11, Clause 6.3.8.1]). Fig. 6 shows that with a threshold $\rho = 0.9$ set accordingly, only CHASM with adaptive modulation meets this requirement. This demonstrates CHASM's advantage from a safety perspective and its crypto-agility, meeting the target with both FN-DSA and ML-DSA.

D. Analysis of Failure Cases

While CHASM reduces safety violations to a rate of just 1.94%, it is important to consider why some violations still occur (Fig. 7). As noted previously, most baseline failures result from the fixed 500 ms certificate interval under [11], which often causes the certificate to be sent too slowly or forces scheduled transmission over a channel that is too poor for successful delivery. Most importantly, we can see that for configurations 6 and 7 (the two best performers in Table IV), nearly all failures are the result of an extremely

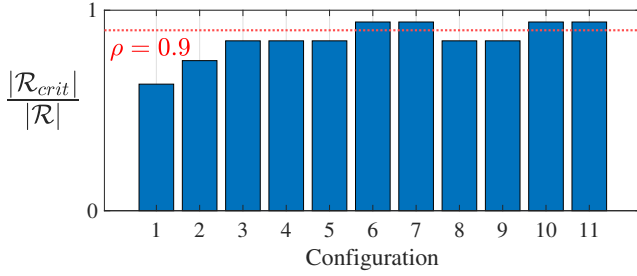


Fig. 6. Critical receiver coverage under each configuration (mean values shown, standard error is always $\leq 0.5\%$). CHASM with adaptive modulation (configurations 6, 7, 10, 11) meets the target of 90% coverage.

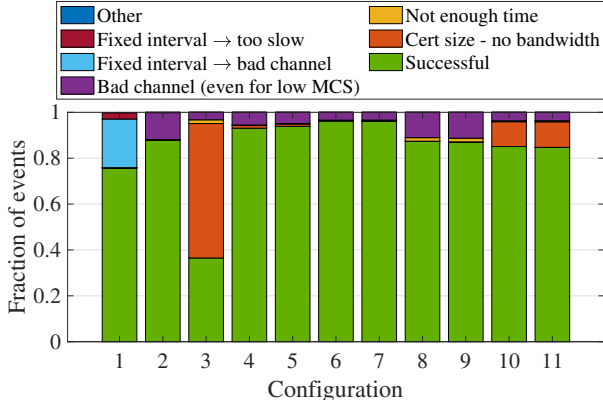


Fig. 7. Distribution of failure case root causes, revealing that for the highest-performing configurations (CHASM with FN-DSA), safety violations usually result from an extremely poor channel rather than a protocol-level failure.

poor channel that could not deliver certificate data even using a very robust, low-order MCS. This reflects the nature of V2V rather than a protocol-level failure, as the majority of failure cases for the baseline IEEE 1609.2 configuration are also due to a channel that is too poor to reliably communicate even unfragmented certificates. We therefore conclude that CHASM avoids the causes of failure that might emerge from fragmenting certificates, such as having too little time or too little bandwidth, sufficiently that its remaining failure cases are dominated by the inextricable aspects of high-mobility V2V that are shared with current systems.

VI. RELATED WORK

To the best of our knowledge, we are the first to develop a standards-aligned, more practical technique for quantum-resistant BSM authentication in C-V2X. Several prior works propose wholesale changes to the C-V2X security ecosystem at the application layer to address the challenges of certificates, e.g., by adopting blockchain [16] or certificateless schemes [47], [48]. However, these and similar schemes differ from today’s public-key design and already-deployed security infrastructure such as the Security Credential Management System (SCMS) [49]; therefore, their adoption would require an expensive overhaul of the entire C-V2X ecosystem. One prior work attempts to solve this problem within the current public-key paradigm [13]—in that work, the authors integrated QRC *only for certificates*, leaving BSMs protected only by

ECDSA signatures. This explicitly transitional approach becomes insecure once quantum computers can break ECDSA in less than 5 minutes, whereas CHASM provides robust quantum resistance to near- and long-term threats. Also, [13] focuses on a different (and now-outdated) V2V protocol with less severe constraints than C-V2X. The few other works on quantum resistance for CVs are surveys that do not offer novel solutions (e.g., [50]), are not backwards-compatible [17], or make assumptions that do not match the current constraints of C-V2X sidelink (e.g., [15] assumes a payload limit of 12,000 bytes vs. a realistic ~ 800).

Outside of CVs, many works have proposed hybrid authentication protocols to support QRC, including in embedded devices [3] and other constrained environments. However, these works assume very different communication settings, for example, where wired connections are available [51] or where bidirectional communication enables key negotiation [3], [52]. These do not readily apply to the wireless, broadcast-only V2V environment. Moreover, many of these works (e.g., [53]) focus on confidentiality or access control rather than authentication. Other related work that aims to reduce the overhead of security or privacy-preservation mechanisms in V2V, such as the cloud-assisted reputation management scheme in [54], addresses problems other than quantum-resistant security and is thus not directly comparable to CHASM.

VII. CONCLUSION

We introduced CHASM to enable a standards-aligned transition to quantum-resistant C-V2X authentication. We proved that under the stated protocol assumptions, CHASM is secure against message forgery and replay attacks by classical and quantum-capable adversaries, and our experimental evidence validates CHASM can run comfortably on commercial C-V2X hardware with a significant safety improvement over both current IEEE 1609.2 and prior work. Most significantly, CHASM with all of our contributions—erasure-coded FN-DSA hybrid certificates and safety-aware adaptive modulation—reduces the safety violation rate in a city-scale evaluation to just 1.94%, suggesting promise for real-world deployment.

REFERENCES

- [1] “ITS America national V2X deployment plan,” Apr. 2023. [Online]. Available: <https://bit.ly/3LP3anB>
- [2] M. Mosca and M. Piani, “Quantum threat timeline report 2025,” Global Risk Institute, Tech. Rep., Mar. 2026. [Online]. Available: <https://tinyurl.com/2s42wrtb>
- [3] G. Tasopoulos, J. Li, A. P. Fournaris, R. K. Zhao, A. Sakzad, and R. Steinfeld, “Performance evaluation of post-quantum TLS 1.3 on resource-constrained embedded systems,” in *Proc. Int. Conf. Info. Security Practice and Exp.*, Taipei, Taiwan, Nov. 2022, pp. 432–451.
- [4] 3GPP, *Release 14: Release Description*, 3GPP TR 21.914, 2018.
- [5] —, *Summary of Rel-16 Work Items*, 3GPP TR 21.916, Jul. 2020.
- [6] —, “Releases,” 2025, <https://bit.ly/4mHSdoM>.
- [7] *Physical layer procedures for data*, 3GPP TS 38.214, Jul. 2022.
- [8] K. N. Khan and F. Ahmad, “ARC: Accurate, real-time, and scalable multi-vehicle cooperative perception,” in *Proc. Int. Conf. Embedded Artificial Intell. Sensing Syst. (SenSys)*, Saint-Malo, France, May 2026.
- [9] *Wireless Access in Vehicular Environments—Security Services for Application and Management Messages*, IEEE Std. 1609.2, 2022.
- [10] *On-Board System Requirements for V2V Safety Communications*, SAE International Surface Vehicle Standard J2945/1, Apr. 2020.
- [11] *On-Board System Requirements for LTE-V2X V2V Safety Communications*, SAE Std. J3161/1, Mar. 2022.

- [12] J. Sloop, E. Allen, E. Jones, C. E. Thornton, L. Liu, and F. Templin, "Federated learning for MCS selection in 5G sidelink MANETs," in *Proc. Military Commun. Conf. (MILCOM)*, Los Angeles, CA, Oct. 2025.
- [13] G. Twardokus, N. Bindel, H. Rahbari, and S. McCarthy, "When cryptography needs a hand: Practical post-quantum authentication for V2V communications," in *Proc. NDSS Symposium*, San Diego, CA, Feb. 2024.
- [14] T. Yoshizawa and B. Preneel, "Post-quantum impacts on V2X certificates - already at the end of the road," in *Proc. IEEE Vehicular Technol. Conf. (VTC)*, Florence, Italy, Jun. 2023.
- [15] Y. Kim and S. Seo, "Signature split method for a PQC-DSA compliant with V2V communication standards," *Appl. Sci.*, vol. 13, May 2023.
- [16] M. U. F. Qaisar, W. Yuan, L. Zhang, S. A. Chaudhry, G. Han, and Y. Zhang, "A robust trust management system for V2X networks integrating ISAC with blockchain smart contracts," *IEEE Trans. Netw. Service Manag.*, vol. 23, pp. 2428–2439, Mar. 2026.
- [17] Y. Gong and B.-J. Hu, "A quantum-resistant key management scheme using blockchain in c-v2x," *IEEE Trans. on Intelligent Transportation Systems*, vol. 25, no. 11, pp. 16 831–16 844, Nov. 2024.
- [18] D. Moody, R. Perlner, A. Regenscheid, A. Robinson, and D. Cooper, "Transition to post-quantum cryptography standards," National Institute of Standards and Technology (NIST), Internal Report 8547, Nov. 2024.
- [19] 5GAA, "List of C-V2X devices," Apr. 2024, Accessed: Apr. 8, 2025. [Online]. Available: <https://bit.ly/45HIEKH>
- [20] National Automobile Dealers Association, "US consumers keep vehicles for a record 12.5 years on average - S&P (Reuters)," 2023, Available: <https://tinyurl.com/3jc224u8>. Accessed: Jan. 21, 2025.
- [21] U.S. Dept. of Commerce, "Fast-Fourier transform over NTRU-lattice-based digital signature standard," National Institute of Standards and Technology (NIST), Draft Federal Information Processing Standard (FIPS) 206, 2026.
- [22] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. Annual Symposium on Foundations of Computer Science*, Santa Fe, NM, Nov. 1994, pp. 124–134.
- [23] U.S. Dept. of Commerce, "Module-lattice-based digital signature standard," Federal Information Processing Standard (FIPS) 204, 2024.
- [24] —, "Stateless hash-based digital signature standard," National Institute of Standards and Technology (NIST), Federal Information Processing Standard (FIPS) 205, 2024.
- [25] R. Perlner, "FIPS 206 status update," NIST, 2025, Accessed: Apr. 7, 2026. [Online]. Available: <https://tinyurl.com/3h9n5cwj>
- [26] *Status Report on the First Round of the Additional Digital Signature Schemes for the NIST Post-Quantum Cryptography Standardization Process*, NIST IR 8528, Oct. 2024.
- [27] P. Li and J. Ding, "Cryptanalysis of the SNOVA signature scheme," in *Proc. Int. Conf. Post-Quantum Crypto.*, Oxford, UK, Jun. 2024.
- [28] G. Twardokus and H. Rahbari, "Towards protecting 5G sidelink scheduling in C-V2X against intelligent DoS attacks," *IEEE Transactions on Wireless Communications*, vol. 22, no. 11, pp. 7273–7286, Nov. 2023.
- [29] 5G; NR: User Equipment (UE) radio transmission and reception; Part 1: Range 1 Standalone, 3GPP TS 38.101 V16.1.1, Jul. 2020.
- [30] *Architecture enhancements for 5G System to support V2X services*, 3GPP Technical Specification 23.287, 2021.
- [31] Bureau of Transportation Statistics, "Average age of automobiles and trucks in operation in the United States," 2022, Accessed: June 21, 2023. [Online]. Available: <https://bit.ly/3PosWR3>
- [32] Federal Communications Commission, 2024, 89 FR 100838.
- [33] S. Balumuri, E. Eaton, and P. Lamontagne, "Quantum-safe public key blinding from MPC-in-the-head signature schemes," in *Proc. Int. Conf. Appl. Crypto. Netw. Security*, Munich, Germany, Jun. 2025, pp. 369–397.
- [34] K.-W. Chiang, T. T. Duong, and J.-K. Liao, "The performance analysis of a real-time integrated INS/GPS vehicle navigation system with abnormal GPS measurement elimination," *Sensors*, vol. 13, no. 8, pp. 10 599–10 622, Aug. 2013.
- [35] A. Goldsmith, *Wireless Commun.* Cambridge University Press, 2005.
- [36] A. F. Molisch, *Wireless Communications*, 3rd ed. Wiley-IEEE Press, 2022.
- [37] T. Abbas, K. Sjöberg, J. Karedal, and F. Tufvesson, "A measurement based shadow fading model for vehicle-to-vehicle network simulations," *Int. Journal of Antennas and Propagation*, vol. 2015, no. 1, Jun. 2015.
- [38] H. Ngo, H. Fang, and H. Wang, "Cooperative perception with V2V communication for autonomous vehicles," *IEEE Trans. Veh. Technol.*, vol. 72, no. 9, pp. 11 122–11 131, Sep. 2023.
- [39] *Physical layer procedures for control (Release 16)*, 3GPP Technical Report 38.213, 2021.
- [40] P. Gajland, J. Jannec, and E. Kiltz, "A closer look at falcon," Feb. 2025. [Online]. Available: <https://eprint.iacr.org/2024/1769>
- [41] Cohda Wireless, "MK6," 2022, Accessed: Mar. 24, 2025. [Online]. Available: <https://www.cohdawireless.com/solutions/hardware/mk6-obu/>
- [42] N. Allen *et al.*, "liboqs," 2025, <https://tinyurl.com/muv8x3ej>.
- [43] Z. Ali, S. Lagén, L. Giupponi, and R. Rouil, "3GPP NR V2X Mode 2: Overview, models and system-level evaluation," *IEEE Access*, vol. 9, pp. 89 551–89 579, Jun. 2021.
- [44] OpenStreetMap, "OpenStreetMap," 2026. [Online]. Available: <https://openstreetmap.org>
- [45] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, and R. Hilbrich, "Microscopic traffic simulation using SUMO," in *Proc. IEEE Intelligent Transp. Syst. Conf.*, Maui, HI, USA, Nov. 2018.
- [46] Highway Data Services Bureau, "Traffic Data Viewer," New York State Department of Transportation, 2024, Accessed: Jul. 29, 2024. [Online]. Available: <https://www.dot.ny.gov/tdv>
- [47] Z. Y. A. Tmari *et al.*, "A post-quantum certificateless aggregate signature scheme for VANETs resilient to rogue-key attacks," *Journal of Communications*, vol. 21, no. 2, pp. 233–244, Mar. 2026.
- [48] Y. Ma, Q. Cheng, X. Li, and Z. Liu, "PQACLA: Post-quantum anonymous and robust certificateless identity authentication for the internet of vehicles," *IEEE Trans. Veh. Technol.*, Nov. 2025, Early access.
- [49] U.S. Dept. of Transportation, "Security credential management system (SCMS)," 2024.
- [50] N. Lohmiller, S. Kaniewski, M. Menth, and T. Heer, "A survey of post-quantum cryptography migration in vehicles," *IEEE Access*, vol. 13, pp. 10 160–10 176, Jan. 2025.
- [51] J. Yao, K. Matusiewicz, and V. Zimmer, "Post quantum design in SPDM for device authentication and key establishment," *Cryptography*, vol. 6, no. 4, p. 48, 2022.
- [52] D. D. Tran, K. Ogata, S. Escobar, S. Akleylek, and A. Otmani, "Formal analysis of post-quantum hybrid key exchange SSH transport layer protocol," *IEEE Access*, vol. 12, pp. 1672–1687, Dec. 2023.
- [53] V. Q. Ulitzsch, S. Park, S. Marzougui, and J.-P. Seifert, "A post-quantum secure subscription concealed identifier for 6G," in *Proc. ACM Conf. Security and Privacy in Wireless and Mobile Netw. (WiSec)*, San Antonio, TX, 2022, pp. 157–168.
- [54] Z. Liu, L. Wan, J. Guo, F. Huang, X. Feng, and L. Wang, "PPRU: A privacy-preserving reputation updating scheme for cloud-assisted vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 74, no. 2, pp. 1877–1892, Feb. 2023.



Geoff Twardokus (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the Rochester Institute of Technology (NY, USA) in 2026. He is currently an assistant professor with the College of Computing at Grand Valley State University in Grand Rapids, Michigan. His broad research interests include security in emerging and next-generation connected technologies, particularly connected vehicles and intelligent transportation systems, as well as quantum resistance in cyber-physical systems.



Hanif Rahbari (Member, IEEE) received the B.Sc. degree from Sharif University of Technology, in 2007, the M.Sc. degree from Amirkabir University of Technology, in 2010, and the PhD degree in electrical and computer engineering from the University of Arizona in 2016. He joined Rochester Institute of Technology (RIT) in 2018 following a postdoctoral associate experience with Virginia Tech. He is currently an associate professor with the Golisano College of Computing and Information Sciences and an affiliated member of ESL Global

Cybersecurity Institute and the College of Engineering at RIT. His research interests include security and reliability in wireless ecosystem, with an emphasis on authentication and leakage of transmission attributes at the physical layer, connected vehicle security, quantum-resistant security, security of spectrum coexistence, digital twins, and education. He received the NSF CAREER Award in 2023, a co-author on over 40 peer-reviewed papers, and is a co-inventor on three US patents.