The following publication W. Lei, M. Li, Y. -H. Kuo and G. Q. Huang, "Converged Address Resolution Protocol for Traceability and Visibility in Cyber–Physical Internet," in IEEE Internet of Things Journal, vol. 12, no. 14, pp. 27322-27338, 15 July15, 2025 is available at https://doi.org/10.1109/jiot.2025.3562803.

Converged Address Resolution Protocol for Traceability and Visibility in Cyber-Physical Internet

Wenqing Lei, Ming Li, *Senior Member, IEEE*, Yong-Hong Kuo, *Member, IEEE* and George Q. Huang, *Fellow, IEEE*

***Abstract*—The proposed Address Resolution Protocol (CPI-ARP) is designed for the Cyber-Physical Internet (CPI) environment, integrating physical and digital logistics networks. By extending traditional ARP mechanisms, CPI-ARP addresses the practical requirement for coordinated asset tracking and state awareness of vehicles and Physical Shipment Units (PSUs) within global logistics systems. The protocol enables coordinated interaction between digital and physical operations by linking logical network addresses (PIP) with physical addresses (PMAC) and physical locations. Key challenges related to synchronizing network and physical address systems, including scalability, mobility, and real-time data synchronization, are addressed in this protocol. Performance evaluations through simulation experiments under various network conditions demonstrate CPI-ARP's ability to reduce latency, improve packet delivery, and support logistics system responsiveness in intra-city and cross-border contexts. These findings pave the way for further development and standardization of CPI networks, contributing to the optimization of global logistics operations.**

***Index Terms*—Cyber-Physical Internet, Physical Internet, Address Resolution Protocol, Logistics Networks.**

NOMENCLATURE

Abbreviations	Full Term
ARP	Address Resolution Protocol
C-ARP	Converged Address Resolution Protocol
CPI	Cyber-Physical Internet
PI	Physical Internet
PIP	Physical Internet Protocol
PMAC	Physical Media Access Control
PSU	Physical Shipment Unit

I. INTRODUCTION

WITH the increasing demand for global logistics, traditional logistics systems are encountering unprecedented challenges. The CPI [1] emerges as an evolving paradigm that extends the Physical Internet (PI) [2], [3] by incorporating a cyber layer, thereby enhancing the operational efficiency of logistics networks. As illustrated in Fig.1, CPI bridges the digital and physical worlds, extending the internet's architecture while introducing architecture-level features tailored to specific logistics needs.

Inspired by the internet, the PI addresses inefficiencies and unsustainability in traditional logistics through standardization and modularization [4]. However, the traditional addressing systems in PI, which suffice for the transportation and management of physical goods between logistics nodes, are inadequate for the dual-layer demands of CPI. With the integration of a cyber layer, CPI requires addressing systems capable of aligning digital data flows and physical logistics operations.

CPI nodes serve dual roles as network-layer entities and physical logistics hubs. On the network layer, CPI nodes handle data packet routing and forwarding, while on the physical layer, they manage the transportation and operation of physical goods [5]. Consequently, CPI addresses must exhibit "dual attributes," enabling precise mapping between network addresses (e.g., IP addresses) and physical locations while ensuring synchronization between digital and physical operations. This integration introduces non-trivial challenges in multi-layered address resolution on address resolution mechanisms. Existing internet protocols, such as the ARP [6], which resolves IP addresses to MAC addresses, are insufficient to meet CPI's dynamic and multi-layered demands. Integrating

This paper is partially supported by two grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU15208824 and T32-707/22-N) *(Corresponding author: Ming Li)*

Wenqing Lei is with the Department of Data and Systems Engineering, The University of Hong Kong, Hong Kong, China (e-mail: u3009767@connect.hku.hk)

Ming Li is with the Department of Industrial and Systems Engineering, the Research Institute for Advanced Manufacturing, as well as the Research Centre for Digital Transformation of Tourism, The Hong Kong Polytechnic University, Hung Hom, Hong Kong, China (e-mail: ming.li@polyu.edu.hk)

Yong-Hong Kuo is with the Department of Data and Systems Engineering, The University of Hong Kong, Hong Kong, China (e-mail: yhkuo@hku.hk)

George Q. Huang is with the Department of Industrial and Systems Engineering and Research Institute for Advanced Manufacturing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong, China (e-mail: gq.huang@polyu.edu.hk)

identity modeling and addressing is crucial in bridging the physical and cyber spaces, supporting coordinated interaction across heterogeneous systems [7]. Thus, a dynamic and context-aware resolution mechanism is needed to dynamically adjust to changing network environments and technological advancements, ensuring efficient data and logistics operations in CPI.

The core research questions addressed in this paper are as follows:

1) How can existing internet addressing mechanisms be adapted to the logistics transport network under the CPI paradigm?
2) How can address resolution protocols in CPI be designed and optimized to ensure scalability and performance in dynamic network environments?
3) How can network and physical address migration and generalization be achieved within CPI?

Although the Physical Internet (PI) has garnered increasing attention in the academic community, with research progress—such as conceptual frameworks, protocol design, and simulation-based validation—its full-scale industrial deployment remains limited. Early-stage pilot projects have been initiated [8], notably by Europe's ALICE platform(www.etp-logistics.eu), but widespread adoption is yet to be realized.

This study, positioned as a theoretical and architectural contribution, conducts experiments and simulations to system-critically evaluate these questions and assess the proposed address resolution scheme's performance and feasibility in real-world applications. These findings provide critical technical support for CPI development, providing preliminary technical insights for future CPI standardization and the automation of global logistics networks.

The structure of this paper is as follows: Section 2 reviews the relevant literature; Section 3 presents the framework for address resolution protocols in CPI; Section 4 details the protocols used within the framework, including address resolution protocols; Section 5 discusses the implementation methods for these protocols and the migration and generalization of network and physical addresses within CPI; Section 6 evaluates the performance of the proposed protocols through simulations; and Section 7 concludes the paper with key findings and future research directions.

II. Literature Review

A. Physical Internet

As logistics and network technologies advanced, Montreuil [2], inspired by the concept of digital network interconnectivity, proposed an innovative logistics framework known as the PI. Montreuil et al. [11] further explained that the core principle of PI is the encapsulation of physical goods into standardized physical packets or containers, managed through unified intelligent interfaces and standardized protocols, enabling the free movement of goods within a digital, modular environment. Subsequently, Montreuil et al. [12] formalized PI as an open global logistics system based on the interconnection of physical, digital, and operational elements, operating similarly to the digital internet through encapsulation, interfaces,

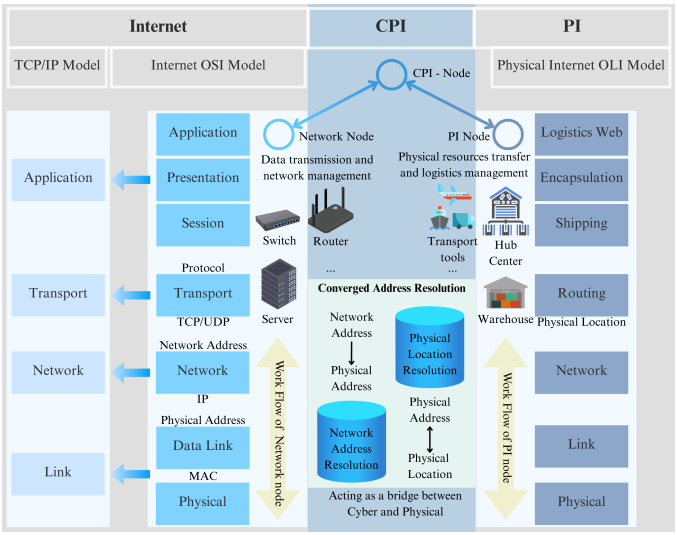

Fig. 1. Comparative Analysis of Address Resolution Across OSI, and OLI Models in the context of CPI [9]–[11]

and protocols. Then, it leverages IoT technologies such as real-time tracking, data sharing, and communication protocols to optimize logistics operations [13].

Pan et al. [8] provided a comprehensive exploration of the PI concept, reviewing research progress in the field, including conceptual studies, evaluation studies, solution design, and validation research. Regarding developing PI protocols, Gontara et al. [14] drew inspiration from the Border Gateway Protocol (BGP) in the digital internet and proposed the PI-BGP protocol as a new routing method. Utilizing the PI-BGP protocol, they established Physical Internet Autonomous Systems (PI-AS) among logistics service providers (LSPs), ensuring the rapid transmission and efficient flow of PI containers within logistics networks. By combining PI's standardization mechanisms with dynamic optimization approaches like Synchromodal Transport, a unified and more sustainable freight system can be achieved, addressing current inefficiencies in logistics networks [4]. Briand et al. [15] introduced an auction-based dynamic routing protocol, significantly improving the efficiency of logistics systems by reducing transportation costs and minimizing empty mileage.

While these studies have contributed significantly to the development of PI, introducing the Cyber layer in CPI presents new challenges, particularly concerning address resolution and network management, which require innovative solutions.

B. Standardizing Address and Protocol in Internet

In computer networks, addressing is fundamental to ensuring effective communication between devices. Two key types of addresses—MAC addresses and IP addresses [16]—operate at different network layers, playing distinct but complementary roles. A MAC address, often called a physical address, is a unique identifier for network adapters or network interface cards (NICs). The IEEE Registration Authority defines an extended unique identifier (EUI), comprising an Organizational Unique Identifier (OUI) and a Company Identifier (CID), ensuring the global uniqueness and traceability of device

identifiers. MAC addresses are primarily used for data frame transmission within local area networks (LANs) and switch forwarding decisions at the MAC layer, a sublayer of the data link layer [17].

Recent studies on MAC have focused on applications in specialized networks, such as vehicular ad-hoc networks (VANETs). For instance, the research, [18] employed fine-grained MAC protocol management and spatio-temporal coordination to achieve efficient data transmission at congested intersections and in complex environments, reducing vehicle communication conflicts and enhancing overall traffic safety and network reliability. Additionally, in [19] proposed a hybrid MAC protocol that integrates Physical Layer Network Coding (PNC) and Random Linear Network Coding (RLNC), effectively enhancing the reliability and efficiency of Basic Safety Message (BSM) transmission in vehicular networks while maintaining compatibility with IEEE 802.11p. Furthermore, [20] designed a self-adaptive TDMA-based MAC protocol that dynamically adjusts frame length and time slot allocation to meet the demands of reliable transmission in high-mobility, high-density environments, effectively mitigating collisions under heavy traffic conditions. Peterson and Davie [21] explain that MAC addresses are mainly utilized for LAN communication. However, when data needs to be transmitted through routers across different networks, such as wide area networks (WANs), IP addresses are used for network-layer communication.

An IP address, which functions as a logical address at the network layer, identifies devices across networks and enables data packet routing and forwarding, facilitating data transmission across multiple networks. The IP address architecture aims to universally identify all computers and terminal devices, allowing devices and services across all networks to follow the TCP/IP protocol. Igulu et al. [22] highlighted. However, IPv6 has been proposed as a solution; its widespread adoption has been slow due to the costs of upgrading hardware and software as well as compatibility issues with IPv4. Despite the gradual deployment of IPv6, IPv4 remains sufficient for the scenarios considered in this study. Feng et al. [23] noted that while this design was sufficient for communication between fixed and trusted hosts in the early days of the internet, it no longer meets the requirements of modern, dynamic internet environments. Internet services are built by combining different protocols at the application layer, such as SMTP [24], FTP [25], HTTP [26], and HTTPS [27], enabling network interconnection and data communication. In addition to addressing systems that assign basic addresses to devices, the ARP resolves addresses between layers.

C. Address Resolution Protocol in Internet

The ARP [6] is a protocol used in computer networks to map a network-layer IP address to the physical hardware address at the data link layer. It serves as a critical addressing mechanism and is widely used in IPv4 networks, playing a key role in ensuring the proper functioning of network communication. In practical applications, ARP faces several challenges, particularly concerning IP address conflicts and the accuracy of hardware address resolution. Arkko and Pignataro [28] focused on IP address conflict detection and guidelines for ARP parameter allocation, proposing specific ARP usage methods to avoid IP address conflicts and ensure global consistency of ARP-related numbers and codes.

Modern network architectures have introduced various extensions and alternative technologies to optimize ARP performance further. For example, to address security issues such as ARP poisoning and spoofing, several security-enhanced ARP schemes have been proposed, including TARP [29], Enhanced ARP [30] and ES-ARP [31], in addition to D-arp [32] to detect and prevent ARP spoofing.

While the studies mentioned above and their contributions have laid a solid foundation for the development of address standardization and resolution technologies, they remain insufficient to address the unique challenges posed by CPI. Specifically, in CPI logistics scenarios, the frequent changes in physical and network nodes, coupled with dynamic network topology adjustments, require a more flexible, real-time, and precise address resolution scheme. Therefore, this paper proposes an ARP scheme tailored to CPI logistics scenarios, combining the internet and PI characteristics to enhance traceable communication and location awareness in logistics network addressing.

III. System Framework

CPI extends the Physical Internet (PI) concept by integrating digital and physical logistics networks. It requires an advanced addressing mechanism that maps network addresses to physical addresses and further to physical locations. Traditional address resolution protocols, such as ARP and DNS, cannot support this multi-layered mapping, making them unsuitable for CPI's dynamic and large-scale environment.

Existing address resolution mechanisms have inherent limitations. ARP operates within a single subnet, resolving IP addresses to MAC addresses, but cannot function across networks. DNS provides hierarchical name resolution but does not incorporate real-time location data, making it unsuitable for logistics tracking. Meanwhile, GPS enables physical positioning but remains disconnected from network-layer identifiers, leading to fragmented logistics tracking and inefficient routing decisions.

A scalable and resilient address resolution mechanism must overcome several key challenges to support CPI's framework. First, network-to-physical address mapping is essential for resolving network-layer identifiers into physical addresses, facilitating communication between digital and physical infrastructures. Second, physical address-to-physical location mapping must be established to associate physical addresses with precise geographical locations, enabling real-time asset tracking. In large-scale logistics operations, scalability and performance are critical, requiring the mechanism to maintain minimal latency and support frequent address updates as assets move dynamically. Additionally, interoperability across networks is necessary to ensure that the resolution process functions seamlessly across different logistics providers and network architectures while maintaining reliability and consistency.

A. Address Resolution Framework in CPI

To address these challenges, CPI employs an integrated address resolution framework that enables interoperable mapping across multiple addressing layers.

This framework consists of two fundamental resolution mechanisms:

- **Network-to-Physical Address Resolution:**A mechanism that translates network-layer addresses into unique physical addresses, enabling inter-network resolution while ensuring basic reliability between digital nodes and physical assets.
- **Physical Address-to-Physical Location Resolution:** A mapping strategy that links physical addresses to precise geographical locations, supporting logistics asset identification and state synchronization.

By integrating these two resolution mechanisms, CPI enables communication between digital and physical infrastructures, allowing for efficient tracking and real-time awareness of logistics assets. Furthermore, this framework operates across multiple domains, facilitating interoperable coordination among logistics assets and service providers.

In the context of this study, for terminology clarification. We distinguish between "physical address (PMAC)" and "physical location" as follows:

- Physical address (PMAC) is the digital representation of an entity's identity in the physical logistics layer. It is similar to a MAC address in traditional networks but extended to cover logistics nodes and devices (e.g., trucks and docking ports). It supports both fixed and mobile formats.
- Physical location refers to an entity's geographical position, expressed through GPS coordinates or spatial references, which may be encoded into PMAC.

This distinction ensures clarity in subsequent discussions on address resolution in CPI, particularly in scenarios where identity and position are essential.

B. Transition to CPI-ARP

CPI harnesses the internet's full potential to enhance network functionality beyond the limitations of traditional systems. Unlike conventional network nodes that only perform digital routing, CPI nodes integrate network-level routing and physical logistics functions. These nodes, including CPI routers, facilitate interoperable data sharing and address conversion within logistics networks. When logistics transportation tools execute tasks in CPI, communication is established through network components at the originating logistics node. This node processes the request and dispatches a task data packet to the corresponding CPI logistics transportation tool. These packets contain network data and the next-hop network address but lack physical address information.

Resolving and converting the next-hop network address into a physical address is essential for completing logistics tasks at the physical layer. CPI-ARP facilitates this process by bridging the network and physical layers and providing readable physical locations. This dual-layer addressing scheme ensures supporting real-time alignment between digital processing and physical asset transitions, optimizing logistics management.

For example, consider a cross-border logistics scenario in the Greater Bay Area (GBA), where a Physical Shipment Unit (PSU) is transported from a manufacturing site in Guangzhou to a distribution warehouse in Hong Kong and subsequently to a construction site in Kowloon. The CPI-ARP system enables real-time synchronization between network and physical addresses, ensuring accurate routing across logistics nodes. At the Guangzhou manufacturing facility, a CPI node assigns a network address (PIP: 10.0.10.3) to a truck carrying the PSU. The routing table is then updated to indicate that the next-hop destination is the warehouse's network address in Hong Kong (PIP: 2.3.4.8). However, this network address alone does not provide sufficient information for physical localization. Therefore, the truck sends a request to the warehouse's CPI-ARP server and obtains the corresponding physical address (PMAC: 7PJP74MP6J:67D8:F899:5969:89F8:38B6:B786:86E6:6607), which is decoded to determine the warehouse's docking location. As the truck proceeds toward the border and into Hong Kong, it enters dynamic transportation zones where network-physical mappings must be continuously updated. Upon arriving at the warehouse, the truck is reassigned a new PIP (10.0.20.5) by the warehouse access point and is directed to a designated docking station. Throughout the journey, CPI routers and access points facilitate real-time resolution and routing updates, adapting to network topology changes, traffic conditions, and customs processing delays. Upon arrival, the truck queries the CPI-ARP table for final address resolution, confirms its docking slot, and updates the logistics record. When the PSU is transferred to a second vehicle for final delivery to the Kowloon construction site, CPI dynamically reassigns network and physical addresses to ensure uninterrupted traceability and coordination.

Additionally, CPI integrates digital twin technology to create a structured digital representation of road networks, parcels, and transportation tools. This integration "supports effective address allocation and management, enabling the unique identification and seamless communication of networked devices. The addressing mechanism prevents conflicts, ensuring that each entity within the CPI network is uniquely identifiable and accessible.

CPI-ARP is designed to serve as a unified standard for address resolution in logistics scenarios. It accommodates the dynamic and complex nature by providing real-time address resolution and management for network and physical locations. Fig.2 illustrates the Application Framework of ARP in CPI, detailing how CPI-ARP enables efficient address resolution across various logistical environments, including factories, transportation hubs, warehouses, and construction sites. This comprehensive approach optimizes network operations and physical logistics tasks, ensuring accurate tracking and seamless communication across the logistics ecosystem.

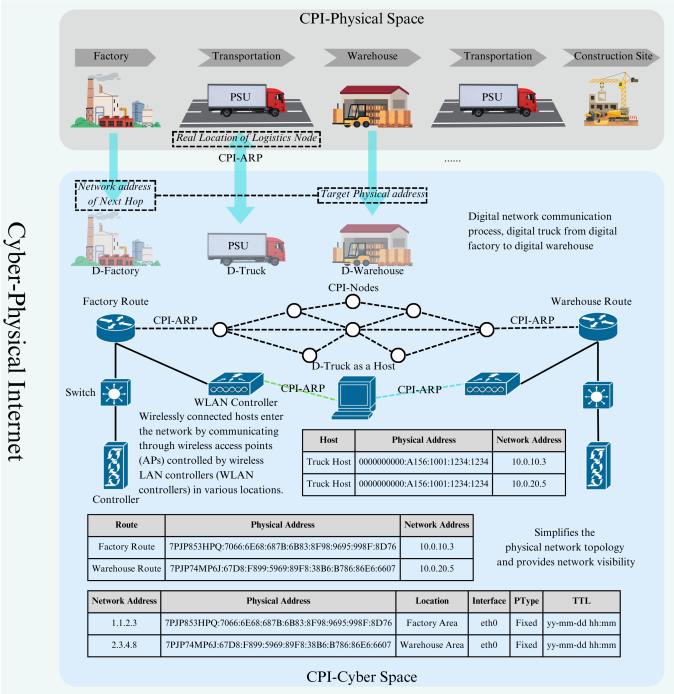

Fig. 2. Application Framework of ARP in CPI

IV. Cyber-Physical Internet Address Resolution Protocol

As the CPI framework integrates physical and network functionalities, an efficient communication and address resolution mechanism is critical to ensuring seamless operations across the cyber-physical network. Generalizing and mapping both network and physical addresses is necessary to meet the specific demands of CPI, enabling smooth transitions and interactions within diverse network environments. A new Address Resolution Protocol designed for the CPI environment was developed to address these unique requirements.

A. Address Transfer and Generalization

Building upon the previous discussion of CPI-ARP, in order to fully understand and implement ARP, this section explores the issues of address transfer and generalization in the CPI network. Address transfer and generalization are critical to ensuring the addressing system operates effectively in diverse network environments and meets the demands of emerging applications and business scenarios. The addressing system can facilitate smooth transitions across network environments by standardizing, ensuring compatibility, and mapping addresses. Furthermore, the system can enhance the network's flexibility, scalability, and interoperability by expanding address structures, application scenarios, and cross-network interoperability.

In the PI, integrating traditional network technologies with the physical world has provided modern logistics with a new perspective. With CPI, this integration has been further optimized to support the convergence of physical and digital networks, enabling dynamic address mapping for PSUs and logistics nodes. With the introduction of the CPI, this integration has been further deepened and optimized. In the CPI network, address transfer and generalization will drive the standardization and optimization of logistics systems, supporting diverse business applications and improving overall logistics efficiency. CPI not only aims to enhance logistics efficiency through intelligent and networked methods but also strives to achieve interoperable interaction between physical and network spaces for goods and equipment.

In this context, physical and logical addresses become two core standards within the CPI addressing system. In the CPI environment, addresses inherit the traditional functions of network addresses while introducing the Physical Media Access Control (PMAC) address. PMAC encodes textual address formats into transmission-friendly formats, ensuring the accurate identification and regional localization of goods and equipment during logistics operations. Additionally, PMAC helps construct CPI network maps and geofences, allowing for the realization of more advanced functions. PMAC addresses are divided into fixed and mobile formats to accommodate different application scenarios. Meanwhile, the Physical Internet Protocol (PIP) address inherits most of the functionality of the IP address, supporting both static and dynamic allocation, and enabling efficient management and path optimization for devices and goods in the network space. The combination of physical and logical addresses provides high synchronization and precision for data and information flows in the CPI network and lays a solid foundation for the global optimization and intelligent automation of logistics systems.

B. Definition of Physical Media Access Address

Within the CPI framework, the PMAC serves as a unique identifier for devices in the physical space and forms the foundation of physical addressing. PMAC addresses are categorized into two main types:

- **PMAC-Fixed:** This is used to identify building areas at logistics nodes, typically applied to main routes in logistics nodes. It is encoded based on geographic coordinates and provides precise localization for physical infrastructure within the logistics network. Format: 7PJP853HPQ:7066:6E68:687B:6B83:8F98:9695:998F:8D76.
- **PMAC-Mobile:** This is used to identify specific physical devices and is assigned based on the unique characteristics of each device. Technologies such as RFID Tags, Barcodes, or QR Codes assign these physical addresses based on device or cargo information. Format: 0000000000:A156:1001:1234:0001.

PMAC-Mobile provides a dynamic and flexible physical address identification mechanism in transportation and handover scenarios by integrating technologies such as RFID tags, barcodes, and QR codes, enabling precise identification and tracking of goods and transport vehicles. Specifically, RFID tags are typically attached to containers or vehicles, and their unique identifiers (UIDs) are mapped to PMAC-Mobile, creating a unique physical address identifier for the transportation process. During transit, RFID readers can dynamically read tag information and update PMAC-Mobile in real-time,

supports accurate alignment of goods' status and routing information. In handover stages at logistics nodes (e.g., distribution centers or transportation hubs), QR codes or barcodes serve as alternative technologies, allowing rapid retrieval of PMAC-Mobile information through scanning. This approach is particularly useful in scenarios where RFID signals are limited or manual operations are required. The dynamic generation and resolution mechanism of PMAC-Mobile not only enhances transparency and accuracy in goods transportation but also facilitates mapping between physical and digital identifiers in logistics networks, thereby establishing an efficient and reliable logistics tracking system.

This integration of physical addressing technologies ensures that goods and equipment can be uniquely identified and tracked throughout various stages of logistics operations. The PMAC-Mobile physical addressing mechanism not only inherits the functionality of traditional network addresses but also improves the accuracy and efficiency of logistics networks. It allows for high synchronization and optimization of data and information flows within the CPI network.

C. Definition of Physical Internet Protocol

Within the CPI framework, network addressing takes the form of Physical Internet Protocol (PIP) addresses, which act as logical identifiers within the network. PIP addresses serve as the foundation for data transmission, device identification, and network logic addressing. This system incorporates the traditional IP addressing methodology into the CPI framework, supporting both dynamic (e.g., through DHCP [33]) and static address allocation. These features enable efficient communication and device management for various nodes across the CPI network. PIP addresses follow traditional IP address structure and subnetting rules, offering a unified addressing scheme across physical and digital environments. This scheme ensures that devices within the CPI network are identified, tracked, and managed efficiently, supporting data routing between network nodes while maintaining compatibility with established IP-based network systems. Additionally, PIP addresses work alongside PMAC addresses to ensure accurate device localization and real-time tracking. The combination of these two addressing schemes enhances asset tracking and operational transparency in logistics networks, enabling synchronized data and physical operations.

CPI-ARP implementation relies heavily on interacting with PIP and PMAC addresses, encapsulating CPI-ARP messages, and enabling end-to-end cross-network data transmission. PIP addresses facilitate communication and routing between network nodes and enhance the system's ability to support consistent interaction between the physical and digital layers of the CPI network, enabling precise device localization and real-time tracking. These capabilities highlight the traceability and visualization features of the CPI network.

D. Address Resolution Protocol in CPI

In traditional internet frameworks, the ARP operates at the data link layer (Layer 2 of the OSI model) and primarily supports IP communication at the network layer (Layer 3 of the OSI model) within LANs. However, ARP in the CPI extends beyond the local network, offering a more structured solution that integrates network and physical layers. The core function of CPI-ARP is to ensure synchronization between network operations and physical logistics processes, bridging the gap between logical and physical addresses.

The CPI-ARP facilitates the resolution of logical (PIP) and physical (PMAC) addresses, ensuring that digital and physical layers function harmoniously. In CPI, physical transportation depends heavily on correctly decoding PMAC addresses, representing the physical location of goods or devices within the logistics network.

A CPI-ARP based on the User Datagram Protocol (UDP) has been designed to achieve layered integration of logical and physical addressing of address resolution in CPI. This protocol supports the mapping between PIP and PMAC addresses and the resolution of physical locations on devices within the CPI network. CPI-ARP builds upon the conceptual framework of traditional ARP but has been optimized to address the specific needs of the CPI environment. Using UDP as the transport medium allows CPI-ARP to achieve real-time logical-to-physical address resolution, enabling low-latency resolution and routing across the CPI network.CPI-ARP comprises two main message types:

- **CPI-ARP Request:** Sent by the requesting device to retrieve the PMAC address of the target device, ensuring communication between network nodes and physical locations.
- **CPI-ARP Response:** The target device responds with its PMAC address, allowing the requesting device to map the logical address to the corresponding physical location.

1) ARP Packet and Table: The design of the packet and table structures provides a solid foundation for address mapping and resolution across both the physical and network layers. Table I shows the structure of the CPI-ARP packet, which includes several key fields such as the Physical Address Type (PTYPE), Network Address Type (NTYPE), and Operation Code (OPER). These fields ensure that devices can communicate through the mapping between physical and network addresses in various network environments. The packet also includes the sender's physical and network addresses (SPA and SNA) as well as the target's physical and network addresses (TPA and TNA). These fields are crucial for identifying network nodes during the request and response processes. For instance, during an ARP request, the target's OPER is set to 1, indicating a request waiting for the target to reply with its physical address. This design allows the network to achieve rapid and efficient address resolution across different layers, supporting low-latency packet resolution.

Table II illustrates the ARP table structure, a key component of the ARP protocol's operational mechanism. Each ARP table entry dynamically maps and updates physical addresses with network addresses and relevant interface information. Fields in the ARP table, such as network address, physical address, interface, and Physical Address Type (PType), enable the system to look up and update node information in the network quickly. The location field (Loc) is combined with VLAN table information to determine the exact location. At

TABLE I
CPI-ARP PACKET STRUCTURE

Field Name	Size(Bytes)	Description
Physical Type (PTYPE)	2	Specifies the type of physical address, e.g., Fixed is 1.
Network Type (NTYPE)	2	Identifies the network address type, e.g., the network address is 0x0800.
Physical Address Length (PLEN)	1	Indicates the length of the physical address (in bytes), e.g., the PMAC address is 26 bytes.
Network Address Length (NLEN)	1	Indicates the length of the physical address (in bytes), e.g., the PIP address is 4 bytes.
Operation Code (OPER)	2	Specifies the type of operation, 1 for request, 2 for reply.
Sender Physical Address (SPA)	26	The sender's physical address.
Sender Network Address (SNA)	4	The sender's network address.
Target Physical Address (TPA)	26	The physical address of the target (empty in request).
Target Network Address (TNA)	4	The target network address.

TABLE II
CPI-ARP TABLE STRUCTURE

Field Name	Type	Description
Network Address	4 Bytes	The network address associated with the physical address.
Physical Address	26 Bytes	The physical address corresponding to the network address.
Interface	Variable	The network interface through which the ARP entry was learned.
PType	Variable	The type of Physical address: fixed or mobile.
Loc	Variable	The location is decoded from the Fixed physical address and combined with the PLoc field in the VLAN table.
TTL (Time to Live)	Variable	The period the entry is considered valid before removing it.

the same time, TTL (Time To Live) defines the validity period of table entries, ensuring the freshness of ARP data and preventing communication failures due to outdated entries. These table structures optimize communication efficiency between hosts, particularly in mobile or dynamic networks, where regular updates to the ARP table allow adaptation to changes in network topology.

2) Workflow: As shown in Fig.3, the CPI-ARP protocol operates in three key phases, as detailed below.

Request Phase: When Device A intends to communicate with Device B, identified by its PIP (PIP_B), but lacks the corresponding PMAC ($PMAC_B$), it constructs a CPI-ARP request message. It fills in its SPA and SNA, sets TPA to null,

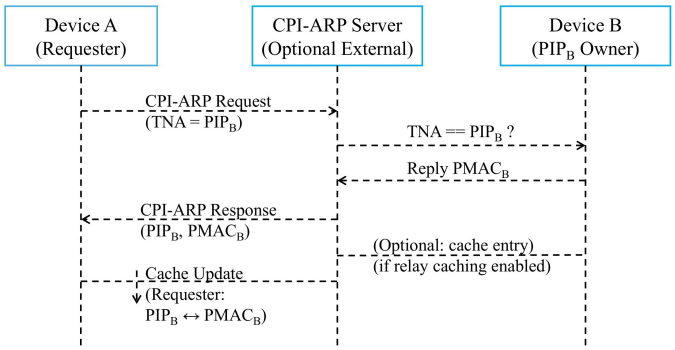

Fig. 3. Workflow of CPI-ARP with Optional Relay Server

and sets TNA to PIP_B. This message is then broadcast or multicast across the network via UDP.

Response Phase: Upon receiving the request, device B checks if the TNA matches its PIP address. If a match is found, device B constructs a CPI-ARP response message, filling in its SPA ($PMAC_B$) and SNA (PIP_B), setting TPA to the requester's SPA, and TNA to the requester's SNA. The response message is then unicast back to device A via UDP.

Cache Update: Upon receiving the CPI-ARP response, Device A extracts $PMAC_B$, optionally decodes the embedded physical location from the PMAC, and updates its local address resolution cache by storing the mapping:

$$PIP_B \leftrightarrow PMAC_B \leftrightarrow \text{Physical Location of Device B}$$

3) Protocol Features: Based on UDP, the CPI-ARP protocol leverages the connectionless nature of UDP to reduce the overhead of connection establishment and maintenance, which is ideal for the real-time and efficiency requirements of CPI networks. The protocol supports dynamic and static address allocation methods and is compatible with traditional mechanisms such as DHCP, making it flexible to adapt to various network environments. Its message format is flexible and extensible, allowing fields to be added or modified as needed to support additional types of physical and logical addresses. Overall, the CPI-ARP protocol is efficiency-oriented and extensible, meeting CPI networks' specific address resolution requirements.

4) Security Considerations: CPI-ARP incorporates a low-overhead and operationally effective security mechanism to ensure secure address resolution in the CPI environment. Given the inherent vulnerabilities of traditional ARP and UDP to man-in-the-middle (MITM) attacks, address spoofing, and replay attacks [30], [32], [34], CPI-ARP employs AES-GCM encryption, Pre-Shared Key (PSK) authentication, and Nonce-based timestamping to enhance the security while maintaining efficiency in dynamic logistics networks.

- **Data Integrity Protection:** CPI-ARP utilizes AES-GCM (Galois/Counter Mode Advanced Encryption Standard) to ensure data confidentiality and integrity.
  The GMAC (Galois Message Authentication Code) mechanism in AES-GCM guarantees that packets have not been altered during transmission.

- **MITM Attack Prevention:** Pre-Shared Key (PSK) authentication ensures that only authorized CPI nodes can participate in address resolution.
  Each CPI-ARP request and response is cryptographically verified, preventing unauthorized nodes from injecting false address mappings.
- **Replay Attack Mitigation:** CPI-ARP implements timestamping and Nonce to ensure each request is uniquely identifiable.
  Requests with expired timestamps or previously used nonces are rejected, preventing adversaries from replaying old packets to disrupt network operations.

These security measures ensure that CPI-ARP maintains high integrity, resilience against spoofing, and protection against unauthorized address resolution. This provides a secure and scalable addressing mechanism for the Cyber-Physical Internet.

V. Implementation of Address Resolution Protocol

As the CPI framework expands to manage both network and physical operations seamlessly, it is crucial to implement reliable address resolution protocols that handle the unique requirements of dynamic and mobile logistics environments. The following sections will detail the implementation of these protocols, focusing on managing both fixed and mobile addressing schemes, along with the integration of logical addressing mechanisms necessary for effective network communication and device management.

A. Implementation Process of CPI- Converged Address resolution protocol

The CPI environment, implementing the C-ARP, begins with a triggering event, where the routing table updates its next-hop address. After the routing table is updated, the system initiates the cross-CPI network resolution protocol. This phase involves communication between the sender and receiver, with the protocol resolving address mappings between different CPI nodes to ensure correct communication across various network segments. The cross-network address resolution mechanism helps the sender accurately locate the physical and network addresses of the receiver, enabling smooth data transmission.

Next, the system decodes the physical address and updates the Address Resolution Protocol table for the first time. The ARP table stores the mapping relationships between network addresses, physical addresses, and actual physical locations. During this process, the protocol confirms CPI-related address information through physical address resolution, allowing the host to identify the target location and initiate communication accurately. This step is critical in matching the physical address with the network identifier, providing essential support for subsequent network operations. Once the address resolution is complete, the host reaches the intended physical location based on the results of the decoding and resolution.

The system uses the previously updated routing table and ARP table to ensure the host reaches the correct destination within the CPI network. At this stage, the physical

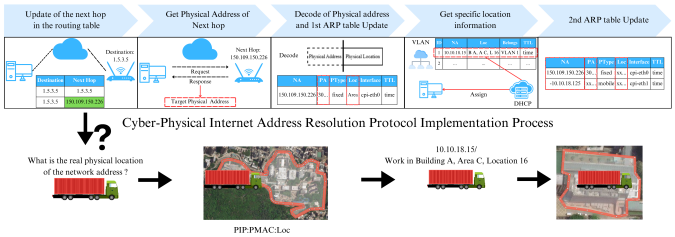

Fig. 4. Implementation Process of CPI Address Resolution Protocol

interaction between the host and the network environment formally begins, with operations depending on the resolved address mappings. The following steps involve internal network address resolution and LAN address allocation, which relies on VLAN configuration and Dynamic Host Configuration Protocol (DHCP) settings. The VLAN table associates each virtual network segment with its corresponding address (including network address, physical address, and specific internal location), ensuring proper segmentation and isolation within the LAN.

In parallel, the local address resolution module works with DHCP to dynamically allocate IP addresses within the LAN, ensuring accurate local address management. The final stage involves updating the complete CPI-ARP table, which integrates all relevant network and physical addresses collected in the previous stages. This table includes fixed and mobile address types and their respective Time to Live (TTL) values. The CPI-ARP table is the core reference for subsequent network operations, providing a comprehensive foundation for address resolution and routing.

As shown in Fig.4, the protocol is executed in phases, covering the management of network and physical addresses within the CPI framework. By combining cross-network address resolution and decoding with VLAN and DHCP configurations, the protocol ensures that the host can accurately identify, locate, and reach the target destination. This multi-stage process not only references traditional protocols like ARP but also extends and redefines their functionality to meet the unique demands of CPI, ensuring robust and dynamic address management in heterogeneous network environments.

B. PMAC - Fixed

The format of a physical address can be represented in various ways, with one of the most intuitive being latitude and longitude coordinates. Latitude and longitude, as used by the Global Positioning System (GPS), provide an accurate method to pinpoint any location on the Earth's surface. In PMAC encoding, actual physical addresses can be converted and represented through latitude and longitude. Although postal addresses (street names, house numbers, cities, and countries) are common, they are not as precise as latitude and longitude. Moreover, differences in address formats and languages across countries and regions limit their global application. Therefore, latitude and longitude are the preferred choices for CPI physical addressing.

PMAC-Fixed encoding is primarily used as a fixed physical identifier for logistics node areas and is typically applied to the

TABLE III
PMAC-FIXED STANDARD

Field Name	Size (Bytes)	Description
Centroid	10	The reference point is calculated based on the selected point and encoded using Plus code (Fig.7).
Coordinate point	16	According to the selected point coding.

main routers of the area network. Traditional MAC addresses consist of an Organizational Unique Identifier (OUI) and a Company Identifier (CID), usually displayed as 12 hexadecimal digits, such as AB:CD:01:02:03:04. To adapt latitude and longitude coordinates to the PMAC-Fixed encoding format, a series of conversions are required. First, precise latitude and longitude coordinates are obtained via GPS, map services, or Geographic Information Systems (GIS). This paper uses a latitude and longitude coordinate array of building outlines obtained from OpenStreetMap's RESTful API and Overpass API, and the encoding process is implemented in Algorithm 1.

The point selection Algorithm 1 consists of the following steps:

- **Input:** A set of building outline coordinates obtained through the Overpass API.
- **Output:** A sorted list of eight points, List<Point>SelectedPoints, which represents the final selection result and completes the filtering of the point set.

After completing the point set selection, the following Algorithm 2 operations are performed on the selected points:

1) **Normalization:** The coordinates of the points in SelectedPoints and the centroid are rounded to six decimal places (precision can be adjusted based on specific requirements). Since latitude and longitude can include negative values, 90 is added to all latitudes, and 180 is added to all longitudes. Then, the difference between each coordinate in SelectedPoints and the centroid's latitude and longitude is calculated. After eight iterations, all differences are combined into a 16-bit binary number, which is converted into a hexadecimal number to form part of the code.
2) **Reference Point:** The reference point is the centroid. There is a concept called Plus Codes on Google Maps, which constructs block units based on the centroid's Plus Codes. These blocks quickly identify places and can restore the differences, helping to build the network map of the building outline.
3) **Integrated:** After completing the above steps, combine the results according to the format of Table III.

C. PMAC - Mobile

In addition to PMAC-Fixed, which identifies specific building areas, the CPI network requires a corresponding PMAC encoding for devices that dynamically enter and exit the network, known as PMAC-Mobile. In CPI, devices such as vehicles frequently need to connect and disconnect from different building areas and establish connections with various networks. To address this need, PMAC-Mobile has been introduced as the unique identifier for registered devices in the CPI network, excluding the regional main routing.

Given CPI's nature as a global logistics network, a unified standard is necessary to ensure that PMAC-Fixed address allocation and format are consistent. However, the identifiable unique identifiers for various logistics devices, including IoT devices, have yet to be standardized. Therefore, it is essential to design a unique device identifier applicable to the CPI environment so that these devices can seamlessly integrate into the CPI network. The design of the device's unique identifier is shown in Table IV.

Algorithm 1 Selection of Points

Input: List of points $points$, integer $n = \texttt{size}(points)$
Output: List of selected points

if $n < 3$ **then**
  return [] {Return empty list if less than 3 points}
end if
$p_0 \leftarrow \texttt{lowestPoint}(points)$ {Find lowest y-coordinate (smallest x if tie)}
`sort`(points) by polar angle with p_0
$m \leftarrow 1$
for $i \leftarrow 1$ to $n-1$ **do**
  while $i < n-1$ **and** `orientation`$(p_0, points[i], points[i+1]) == 0$ **do**
    $i \leftarrow i+1$
  end while
  $points[m] \leftarrow points[i]$
  $m \leftarrow m+1$
end for
if $m < 3$ **then**
  return []
end if
`stack` $\leftarrow$ $[points[0], points[1], points[2]]$ {Initialize stack}
for $i \leftarrow 3$ to $m-1$ **do**
  while `stack.size()` > 1 **and** `orientation(nextToTop(stack), stack.peek()`, $points[i]) \neq 2$ **do**
    `stack.pop()`
  end while
  `stack.push(`$points[i]$`)`
end for
return `list(stack)`

{**Helper Functions:**}
`orientation`(p, q, r): Returns 0 if collinear, 1 if clockwise, 2 if counterclockwise.
`nextToTop(stack)`:
  $t \leftarrow$ `stack.pop()`
  $result \leftarrow$ `stack.peek()`
  `stack.push(t)` {Restore the popped element}
return $result$

Algorithm 2 PMAC Generation

Input: Coordinates list $points$, centroid $centroid$, integer $bits$, real $boundary$
Output: PMAC string $pmac$

$(lat_c, lng_c) \leftarrow (centroid.lat + 90, centroid.lng + 180)$
`result` $\leftarrow$ [] {Initialize list to store binary values}
for $point \in points$ **do**
  $(lat_p, lng_p) \leftarrow (point.lat + 90, point.lng + 180)$
  $latDiff \leftarrow lat_p - lat_c$, $lngDiff \leftarrow lng_p - lng_c$
  $binary \leftarrow 0$, $minVal \leftarrow -boundary$, $maxVal \leftarrow boundary$
  for $i \leftarrow 0$ to $bits - 1$ **do**
    $midVal \leftarrow (minVal + maxVal)/2$
    if $latDiff \geq midVal$ **then**
      $binary \leftarrow (binary << 1)|1$
      $minVal \leftarrow midVal$
    else
      $binary \leftarrow (binary << 1)$
      $maxVal \leftarrow midVal$
    end if
  end for
  `result.add`($binary$)
end for
$hexString \leftarrow$ `joinWithColon`([`hex`(b) | $b \in$ `result`]) {Convert binaries to hex}
$plusCode \leftarrow$ `encodePlusCode`($centroid$)
return `concat`(plusCode, ":", hexString)

{**Helper Functions:**}
`encodePlusCode`($centroid$): Encodes the centroid's coordinates into a PlusCode.
`joinWithColon`($list$): Joins elements of the List with ":" as the delimiter.

IEEE manually manages traditional MAC addresses and OUIs to guarantee uniqueness. In contrast, CPI's PMAC-Mobile ensures uniqueness by truncating the device's serial number, which is automatically generated and managed by a database management system to prevent duplication. The extension field is reserved for future development and can also be used to ensure uniqueness. Finally, the comprehensive combination of these elements guarantees the uniqueness and universality of the overall identifier.

D. Logic Addressing Allocation Scheme

In CPI, designing a dedicated Dynamic Host Configuration Protocol service for the CPI local area network is crucial for the implementation of CPI-ARP. This service dynamically allocates and manages Physical Internet Protocol addresses and meets specific CPI network requirements, such as efficient address allocation, precise device tracking, mobility support, and real-time data synchronization. Crucially, it guides the execution of logistics transportation tasks within the internal network by assigning addresses to hosts for specific operations. This objective is achieved by designing the DHCP service to include several modules: the address allocation module, mobility support module, real-time synchronization module, and VLAN management module.

TABLE IV
PMAC-MOBILE STANDARD

Field Name	Size (Bytes)	Description
Access Affiliation	1	Single digit or letter representing device affiliation, e.g., A, B, C or 1, 2, 3.
Country/Region Code	3	ISO 3166-1 numeric code representing the country or region, e.g., China is 156, USA is 840.
Device Type	1	One-digit code representing device type, e.g., 1 for vehicles, 2 for sensors.
Device Model	3	Three-digit code representing the specific device model, e.g., 001, 002.
Device Serial Number (Extract)	4	Extract the last four digits of the full device serial number, e.g., 1234.
Auto-Increment Serial Number	4	Four-digit auto-increment serial number ensuring uniqueness, e.g., 0001.
Extension Field	10	Combination of numbers and letters reserved for future use. Left blank by default or filled with 10 zeros (0000000000).

Address Allocation Module: This module ensures that each device can dynamically obtain a unique PIP address, offering high adaptability and scalability. It maintains a pool of PIP addresses, dynamically allocating addresses based on device requests and ensuring uniqueness and validity.

Mobility Support Module: This module supports the seamless transition of mobile devices (such as transport vehicles and mobile sensors) between different network nodes, ensuring that their PIP addresses remain valid throughout the movement. It updates the PIP address information in real time, ensuring timely adjustment and optimization of data transmission paths.

Real-Time Synchronization Module: Integrated with GPS and GIS technologies, this module tracks the geographic location of devices in real-time and dynamically updates their PIP address information. It monitors the status and network connection of devices, ensuring the DHCP service's efficient operation and timely response.

VLAN Management Module: This module divides VLANs based on different physical locations (such as buildings and regions), ensuring that each VLAN segment corresponds to a specific geographic location and retains its configuration. It assigns specific address ranges to each VLAN, clearly identifying the precise location of devices.

This design supports the address resolution of CPI-ARP within the LAN, ensuring efficient address management in the network. It also meets the needs of mobile devices and real-time data synchronization while enabling finer network control and location identification through VLAN management.

E. Logic Addressing Application

In the CPI network, the logical addressing mechanism drives the intelligence and real-time responsiveness of network components through dynamic updates, helping optimize network layout and improve the operational efficiency of logistics transportation. PIP addresses serve as unique digital identifiers for devices within the network, ensuring that devices can communicate, locate other devices, and be recognized by other devices in the network. PIP addresses demonstrate logical management capabilities through hierarchical addressing, subnetting, and dynamic allocation. However, the migration and generalization of PIP addresses, while preserving the standardization of traditional IP addresses, remains challenging, as it introduces unique functionality within the CPI environment. For instance, 192.168.0.1/24 represents a specific network, where 192.168.0 identifies the network, and 1 identifies the specific device within that network. The subnet mask 255.255.255.0 determines which bits represent the network portion and which represent the host portion. This structure applies in the CPI network, though its meaning extends within logistics management.

In the CPI network, PIP addresses are not merely digital combinations but also key to device positioning and operation. In WAN, PIP addresses simplified logistics operations as the foundation of interconnected communication. In LAN, PIP addresses are pre-allocated through VLAN segmentation by network management to locate devices within the LAN precisely. For example, in a warehouse environment, PIP addresses are reserved for logistics transport tools' loading and unloading areas through VLAN segmentation. When a CPI logistics vehicle arrives at the Warehouse, it connects to the warehouse network and is allocated a PIP within the LAN. The vehicle then sends an ARP message to the Warehouse, which assigns the reserved PIP to the logistics tool, ensuring it accurately reaches the designated loading/unloading location. CPI enables support for synchronized asset tracking and network management, and devices by combining physical and logical addresses. With a unified coding system, location-based addressing, and real-time data updates, this system supports logistics networks' dynamic optimization and intelligent operations, driving innovation and development in the logistics industry.

VI. A Demonstrative Case Study

A demonstration case study was conducted to validate the proposed CPI-ARP and demonstrate protocol functionality and implementation feasibility to facilitate implementing a new address system within CPI logistics. First, an experimental scenario was described and established based on the context of CPI logistics. Then, the experimental results for this scenario were presented to assess and validate the PMAC-Fixed encoding mechanism and CPI-ARP itself. Finally, a sensitivity analysis was conducted from different perspectives, and the experimental results were compared and discussed.

A. Demonstration of CPI-ARP Functional Capabilities

1) Scenario Description of CPI-ARP: To effectively validate CPI-ARP within the context of CPI logistics, two scenarios were considered:

Scenario 1: This scenario examines CPI-ARP between logistics nodes within an Intra-city network, as illustrated in Fig.5 (a). The servers of the two logistics nodes are located on Hong Kong Island and in Kowloon, Hong Kong, respectively. Both servers possess unique PIP and PMAC addresses, encoded based on their geographic coordinates and contour data, as discussed in Sections 4 and 5.

Scenario 2: This scenario explores CPI-ARP between logistics nodes in a Cross-border network, as illustrated in Fig.5 (b). The servers in this scenario are located in Hong Kong and Guangzhou, China. Similarly, each server has a unique PIP address and PMAC address, both encoded according to their respective locations' geographic and contour data.

In both scenarios, the transportation of the PSUs between nodes is divided into two layers: the cyber layer and the physical layer. In the cyber layer, the feasibility of transportation is determined through PIP addresses and routing mechanisms. Once the routing table is updated, the PMAC address corresponding to the next-hop PIP address is retrieved via CPI-ARP. This decoded PMAC address provides the exact physical location, allowing the transport vehicle in the physical layer to determine the next-hop destination. Fig.6 presents a sample illustration of this process. The CPI-ARP request based on the UDP protocol is shown in Fig.7, and Fig.8 is the mapping table of the centroid. Given that these processes span different network environments (intra-city and cross-border networks), the following assumptions were made for the experimental scenario:

1) The study focuses solely on land transportation via direct routes, excluding intermediate transshipment points.
2) This analysis does not consider Specific conditions of the PSU (such as cargo type, weight, etc.).
3) The routing is set to default, with the next-hop PIP address being the destination.
4) Only PMAC-Fixed addresses within the primary area are considered without addressing auxiliary situations.

2) Experimental Description of PMAC - Fixed Encoding: First, we consider the sample locations. Given the varying sizes and coordinate points of the building outlines in each location, we do not require an overly large sample size. Therefore, we collected 30 location samples, including 18 from Hong Kong, 9 from Guangdong Province, and 3 from Macau. The inconsistency in the outlines and sizes of these samples increases the representativeness and generalizability of the analysis. Next, we consider the point selection issue. In previous assumptions, eight points were considered optimal, as more than eight points would result in excessively long encoding, while fewer than eight points might lead to significant errors. To ensure rigorous analysis, we selected between 4 and 12 even-numbered points. The reasons for selecting even-numbered points are as follows:

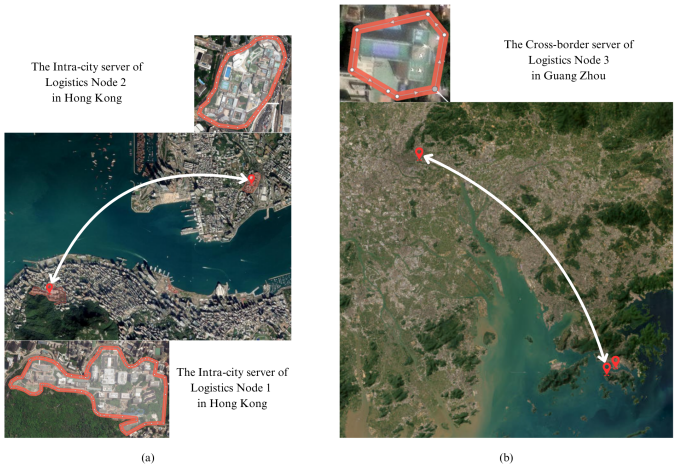

Fig. 5. Use of CPI-ARP between logistics nodes in the Intra-city network and cross-border network

Fig. 6. Use Sample of CPI-ARP between logistics nodes in the Intra-city network

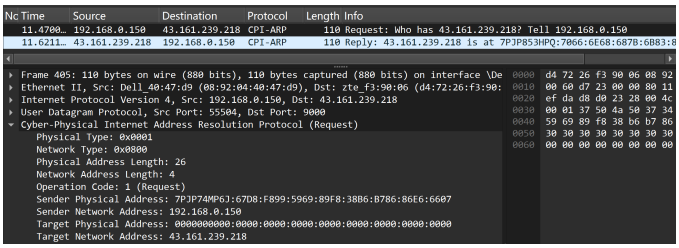

Fig. 7. Sample of CPI-ARP Request

Mapping of Open Location Codes

Base 20 digit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Code digit	2	3	4	5	6	7	8	9	C	F	G	H	J	M	P	Q	R	V	W	X

Block sizes of Open Location Codes

Code length	2	4	6	8	+	10	11
Block size	20°	1°	0.05° (3′)	0.0025° (9′)		0.000125° (0.45″)	
Approximately	2,200 km	110 km	5.5 km	275 m		14 m	3.5m

Fig. 8. Plus Codes (Open Location Codes)

1) **Geometric Uniformity:** Even-numbered points help achieve more uniform distribution in geometric shapes. For example, choosing four points can form a rectangle or polygon, ensuring even coverage in all directions. Odd-numbered points may lead to asymmetric distributions, affecting the stability and consistency of the encoding.
2) **Boundary Testing:** Choosing 10 and 12 points allows boundary testing to evaluate the performance of the encoding algorithm under extreme conditions.

Then, GET requests with attached parameters were sent from Python to Java to retrieve the coordinate point sets for the specified locations from the Overpass API. To minimize the impact of outliers, we selected 4 to 12 even-numbered points for each location and iterated 100 times. In total, each location was sampled 500 times.

3) Experimental Description of CPI-ARP: The primary objective of this benchmark is to evaluate the performance of CPI-ARP under different network conditions and loads. A multi-threaded concurrent request method was employed to simulate high concurrency in real network environments. Each thread represented an independent client that sent multiple CPI-ARP requests to the server, testing the protocol's responsiveness and stability under high-load conditions.

Table V presents the experimental setup covering hardware, network, software environment, and test configurations to ensure experimental rigor while evaluating CPI-ARP performance. By conducting experiments between servers and clients at different physical locations and simulating real transmission in a virtual network environment, we ensured the accuracy and repeatability of the test results. Experiments were conducted in cross-border (CB) and Intra-city (HK) networks, simulating different network environments to test CPI-ARP performance under varying load conditions. Scripts were used to generate requests and record response times, ensuring the efficiency and standardization of the experiments.

4) Experimental Result: This section presents and analyzes key data obtained during the experiment, evaluating the performance of the PMAC-Fixed encoding mechanism and the CPI-ARP protocol in different network environments. First, for the PMAC-Fixed encoding mechanism, the experimental data focuses on encoding precision, computational efficiency, encoding length, and performance under varying geographic locations and building contour conditions. Specifically, the evaluation examines how different numbers of contour points influence PMAC-Fixed encoding's adaptability to varying building structure complexities and how encoding length can be optimized while maintaining precision to improve overall efficiency. Sensitivity analysis of these factors reveals performance differences and optimization potential of PMAC-Fixed encoding across various scenarios.

Next, the experiment evaluates the network performance of the CPI-ARP protocol under high-load conditions. Four key indicators, including latency, jitter, success rate, and packet loss rate, are typical Quality of Service (QoS) metrics used to measure the protocol's responsiveness, stability, and reliability, as shown in Table VI. By simulating high-concurrency requests in Intra-city and cross-border network scenarios, the

TABLE V
FUNCTIONAL EXPERIMENTAL SETUP

Experimental Setup	Description
Hardware Environment	**Client Specifications**: 4-core CPU, 8GB RAM, 80GB SSD, 1000 Mbps bandwidth. The client is located in the Hong Kong region.
	Server Specifications: Two servers located in Guangzhou and Hong Kong, each with a 2-core CPU, 2GB RAM, 40GB SSD, and a peak bandwidth of 20Mbps.
Network Conditions	**Physical Location**: Server A is in Guangzhou, and Server B is in Hong Kong. The client is also in Hong Kong but in a different network range and physical area than Server B.
	Network Environment: The tests are conducted in an isolated virtual network environment to ensure the accuracy and repeatability of the results and to avoid interference from external network traffic.
Software Environment	**Operating System**: Ubuntu 20.04 LTS
	Programming Language: Python 3.8
	Testing Tools: Python scripts generate CPI-ARP requests and record response times.
Test Configuration	Each thread independently sends 100 CPI-ARP requests, with the number of threads set to 10 or 50 to simulate different concurrent load conditions.

TABLE VI
CPI-ARP QOS TEST METRICS

QoS Test Metric	Description
Latency	The round-trip time from sending a CPI-ARP request to receiving a response. This metric is used to measure the response speed of the protocol.
Success Rate	The ratio of successfully received CPI-ARP responses to the total requests within a certain period. This metric is used to evaluate the reliability of the protocol.
Packet Loss Rate	The ratio of requests that did not receive a response to the total number of requests. This metric is used to detect network packet loss under high load conditions.
Jitter	The variability of latency is expressed as a standard deviation. This metric is used to assess the consistency and stability of response times.

experiment aims to verify the effectiveness of CPI-ARP and explore its applicability and areas for improvement in logistics networks. A comparative discussion based on empirical data of the experimental results for both the PMAC-Fixed encoding and CPI-ARP protocol further reveals their potential applications in different logistics scenarios and provides suggestions for future optimizations.

B. Security Enhancement and Evaluation of CPI-ARP

This section presents security enhancements for CPI-ARP and evaluates its resistance to potential attacks, particularly in defending against MITM attacks and packet manipulation attempts. The evaluation includes both security mechanism implementation and experimental validation, supplemented by theoretical security analysis.

1) Threat Model and Attack Scenarios: The security evaluation is conducted under a realistic adversarial model, where an attacker MITM attempts to intercept, manipulate, and redirect CPI-ARP resolution messages. The following attack scenarios are considered:

- **Spoofing Attack:** The attacker forges UDP source addresses to impersonate the CPI-ARP server and inject fraudulent address mappings.
- **MITM Attack via Packet Injection:** The attacker (Node A) intercepts UDP-based CPI-ARP requests, modifies address resolution responses, and attempts to redirect communication.
- **Replay Attack:** The attacker replays an intercepted CPI-ARP response packet to impersonate a legitimate server.

2) Experimental Setup: The experimental validation is conducted in a controlled testbed consisting of:

- **Client(CPI-ARP requester):** A node initiating CPI-ARP queries for address resolution.
- **Server (CPI-ARP resolver):** A legitimate resolver that maps PIP (Physical Internet Protocol) addresses to PMAC (Physical MAC addresses).
- **Attacker (MITM node):** A node capable of sniffing and injecting UDP-based CPI-ARP packets.
- **Network Configuration:** Client (WAN IP: Has) is not in the same subnet as Server or MITM.
  Server IP and MITM IP: On the same subnet, but not on the same subnet as the client
  MITM IP. (WAN IP: Both have)
- **UDP Communication Ports:** 9000 (server), 9999 (MITM)
- **Packet Sniffing and Injection:** tcpdump, iptables, and netcat (nc) for traffic manipulation
- **Attack Prevention Mechanisms:** To mitigate the identified threats, the CPI-ARP framework implements the following countermeasures in Table VII.

A dedicated packet logging system is deployed to capture all transmitted, modified, and received packets for analysis.

3) Experimental Result: This section presents an empirical evaluation of the CPI-ARP system under different operational conditions, focusing on the trade-off between security, performance, and resilience to attacks. The comparison between secure and non-secure configurations provides insights into how cryptographic protections affect system efficiency and robustness.

This study's experimental results were obtained in a controlled and isolated test environment to ensure accuracy and reproducibility. This closed network setup eliminates external interference, such as unexpected network congestion, variable latencies, and security threats from public networks. While this provides a reliable baseline for evaluating the CPI-ARP protocol, further validation in real-world logistics environments

TABLE VII
CPI-ARP COUNTERMEASURES TABLE

Threat	Mitigation Strategy
UDP Spoofing	Pre-Shared Key (PSK) Authentication ensures that only authorized entities can generate valid CPI-ARP responses.
MITM Attack	AES-GCM Encryption ensures that address resolution responses are cryptographically authenticated, preventing unauthorized modification.
Replay Attack	Nonce-based Challenge-Response Protocol prevents the reuse of previous responses, ensuring the freshness of CPI-ARP replies.

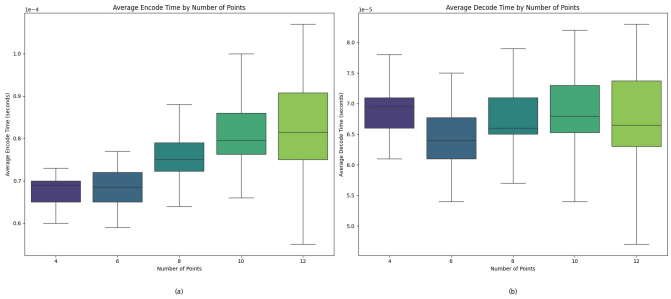

Fig. 9. Box plot of the average encoding time versus the number of points

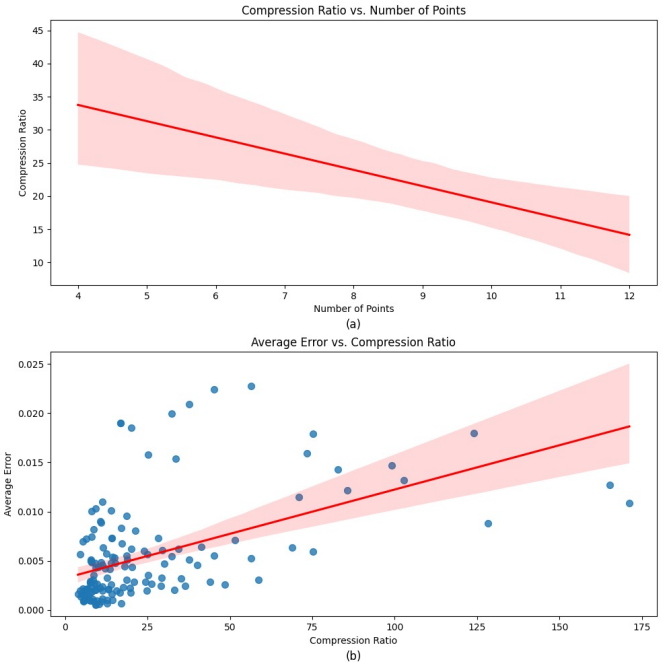

Fig. 10. a) Regression of compression ratio versus number of points
b) Scatter plot and regression line of average error versus compression ratio

is required to assess its full performance under dynamic and large-scale deployment scenarios.

C. Analysis and Discussion

Based on the experimental data presented above, this section evaluates the PMAC-Fixed encoding mechanism's efficiency and the CPI-ARP protocol's performance in different network environments. First, a sensitivity analysis was conducted on the PMAC-Fixed encoding mechanism, focusing on the relationship between the number of encoding points, compression ratio, and average error. These three factors were chosen because the number of encoding points directly affects the resolution and accuracy of encoded data, the compression ratio determines the trade-off between data size and information loss, and the average error provides a measure of encoding efficiency and precision. Fig.9 shows that the encoding and decoding time of PMAC-Fixed increases with the number of encoding points, especially when the number of points exceeds 8, where a significant rise in time is observed. Additionally, the compression ratio positively correlates with the average error, as shown in Fig.10 (a), (b). These findings indicate that while increasing the compression ratio can effectively shorten the data length, it also leads to a certain degree of accuracy loss. In practical applications, achieving an optimal balance between encoding efficiency and data accuracy requires dynamically adjusting the number of encoding points based on actual needs. Implementing this dynamic adjustment ensures data compression while keeping the error within a reasonable range, allowing the system to maintain high encoding efficiency while minimizing the error caused by compression.

Subsequently, the experimental data from cross-border (CB) and intra-city (HK) networks was analyzed to evaluate the performance of the CPI-ARP protocol under different network environments. Despite identical server configurations, CB and HK networks exhibited observable performance differences due to varying network conditions. As shown in Fig.11, 12, and 13, the HK network demonstrated more stable latency and higher throughput compared to the CB network. Even under high load (50 threads), the latency increases in the HK network remained gradual, exhibiting better scalability. In contrast, the CB network experienced a sharp increase in latency as the load increased, particularly under a 50-thread load, where the system's processing capacity reached its limit, hindering further throughput improvements. Moreover, cross-border networks' long-distance transmission and complex routing exacerbated this performance bottleneck.

Additionally, comparing packet loss rates and network stability revealed significant differences. In high-load conditions, the packet loss rate in the CB network exceeded 6%, constraining system responsiveness. In contrast, the HK network maintained a lower packet loss rate, peaking at only 0.5%, and showed stronger network stability under most load conditions. Jitter analysis further indicated a observable jitter variations in the CB network under high load, reflecting increased instability during cross-border transmission. On the other hand, the HK network exhibited consistently low jitter under different load conditions, indicating more uniform data transmission. Based on the above analysis, several optimization directions for CPI-ARP are proposed. To address the high latency and instability in cross-border networks, where QoS can be improved through stronger traffic control and congestion management mechanisms can be introduced into

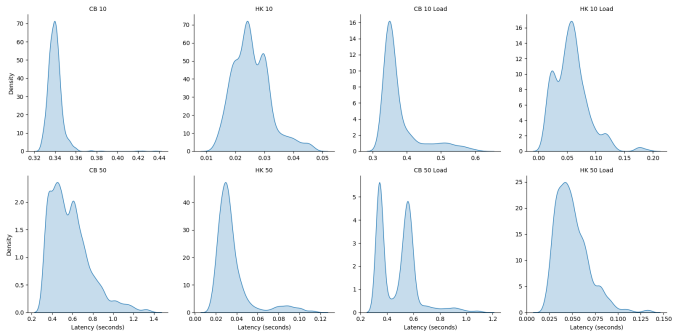

Fig. 11. KDE plot of Latency Distribution

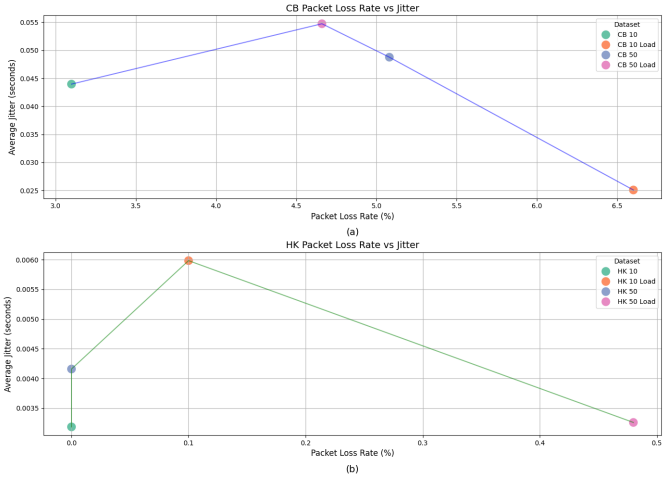

Fig. 12. Plot of packet loss rate versus jitter

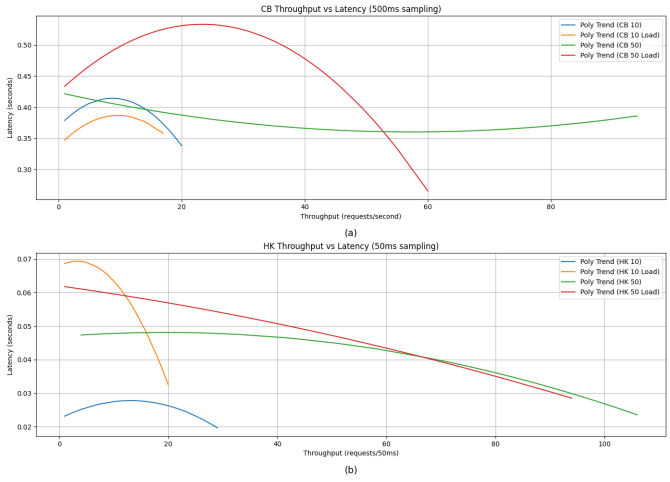

Fig. 13. Plot of Throughput versus latency (polynomial fitting under different thread numbers and load conditions)

the protocol, particularly in terms of routing optimization and error control during cross-border transmission. For intra-city networks, due to their low latency and high stability, protocol performance can be further enhanced to support the processing of larger-scale data requests in high-concurrency scenarios.

The relationship between throughput (requests per 50ms) and latency (response time in seconds) is illustrated in Fig.14.

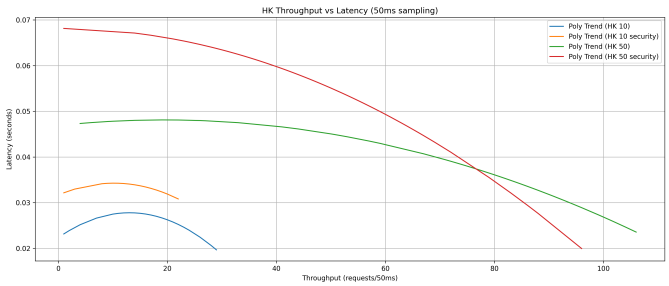

Fig. 14. Plot of Throughput versus latency (polynomial fitting under different thread numbers and security)

TABLE VIII
CPI-ARP SECURITY (WITHOUT OR WITH) AVERAGE LATENCY

Scenario	Without Security (avg/s)	With Security (avg/s)	Overhead (%)
HK 10	0.025580	0.033100	+29.4%
HK 50	0.035811	0.053067	+48.2%

The baseline configurations (HK 10, HK 50) show a decreasing latency trend as throughput increases, reflecting efficient resource utilization in normal conditions. However, with security mechanisms enabled (HK 10 Security, HK 50 Security), the system experiences an increase in latency, attributed to the additional computational overhead introduced by AES-GCM encryption, PSK authentication, timestamp validation, and nonce verification.

Despite the increased latency, implementing security mechanisms is necessary to ensure data integrity and authentication, preventing unauthorized access and malicious modifications. In addition to latency, security measures may also impact resource consumption, requiring additional memory and processing power, which could limit scalability in resource-constrained environments. The impact of these security measures is more evident in high-throughput scenarios (HK 50 Security), where the computational cost of cryptographic operations becomes more significant. Nonetheless, despite high request rates, the system maintains operational stability under high load, reflects the trade-off between performance and security.

Table VIII summarizes the average latency for different configurations, with and without security mechanisms, to quantify the performance impact. The results show that enabling security increases latency by approximately 29.4% for HK 10 and 48.2% for HK 50. This increase is expected due to the added cryptographic computations required for each request-response exchange.

Security mechanisms, while adding computational overhead, are essential for protecting the CPI-ARP system against various attacks. The effectiveness of these protections was validated through attack simulations, as shown in Table IX. Without security mechanisms, spoofing, MITM, and replay attacks achieved a 100% success rate, indicating a complete lack of protection. However, when AES-GCM encryption, PSK authentication, and nonce-based freshness validation were em-

TABLE IX
CPI-ARP SECURITY TEST RESULT

Attack Types	Success rate (unencrypted)	Success rate (AES-GCM + PSK + Timestamp + Nonce)
Spoofing	100%	0%
MITM Attack	100%	0%
Replay Attack	100%	0%

ployed, all attacks were completely mitigated, reducing their success rate to 0%.

Without encryption and authentication, attackers could intercept, modify, and inject CPI-ARP responses, compromising the integrity of address resolution. AES-GCM encryption ensured cryptographic integrity, PSK authentication prevented unauthorized responses, and nonce-based timestamp validation eliminated replay attacks. These findings confirm that security mechanisms are indispensable for supporting CPI-ARP resilience under adversarial scenarios despite the minor performance overhead. In conclusion, the results highlight that while security measures introduce moderate processing overhead, they effectively prevent network-level attacks, ensuring the CPI-ARP system remains both secure and operational in high-throughput environments. The trade-off between performance and security is justified, reinforcing the necessity of cryptographic protections in next-generation CPI-ARP.

VII. CONCLUDING DISCUSSION

This paper introduces a concept for address resolution in the CPI environment, presenting it as an integrated solution that extends standard internet protocols to support operational coordination in logistics networks. The protocol facilitates the migration and generalization of physical and network addressing by encoding real physical addresses and setting up VLAN configurations. A protocol was proposed to organize the resolution of addressing issues, and its effectiveness was validated through experiments.

The test environment designed in this study combined different concurrent load conditions and diverse network topologies. The experimental results demonstrated that CPI-ARP maintained low latency and success rates even under high concurrency, verifying the solution's scalability and effectiveness. Additionally, the flexible migration mechanism of physical and network addresses in CPI ensures the continuity and accuracy of logistics nodes and devices when the network topology changes dynamically, laying the technical foundation for future logistics network expansion.

This paper contributes to the field in three main areas. First, the scope and application framework of the addressing mechanism proposed for CPI integrates existing internet addressing mechanisms with CPI concepts, applying them to logistics transport networks. The second contribution is the design and implementation of the integrated CPI-ARP, which provides efficient network and physical address resolution mechanisms for logistics nodes within the CPI network. This protocol addresses the issue of asynchronous resolution between physical and network addresses in current logistics networks and exhibits performance improvements in node-level traceability and operational responsiveness, as supported by experimental validation. The third contribution is introducing and validating the PMAC-Fixed and PMAC-Mobile physical address encoding systems within the CPI environment, ensuring that this encoding design supports structured data transmission and identification across heterogeneous logistics units, devices, and goods. This structured encoding mechanism enables logistics data exchange and coordination and data synchronization across nodes, providing foundational technical mechanisms to support coordinated logistics operations in distributed environments.

However, since the application of CPI in the logistics field is still in its early stages, there are certain limitations to the proposed CPI-ARP protocol and its address migration and generalization mechanisms. First, although this study integrates AES-GCM encryption for data integrity, PSK-based authentication for MITM attack prevention, and timestamp-based nonce mechanisms for replay attack mitigation, further research is required to enhance security in large-scale deployments. Future work should explore dynamic key exchange mechanisms, resilience against DoS attacks, and mitigation strategies for Sybil attacks to enhance secure data transmission in logistics networks. Second, the study only tested certain logistics scenarios and did not cover more complex dynamic environments, such as high-frequency interactions between multiple nodes. Future research should expand the scope of scenario testing, especially in large-scale applications within global logistics networks. Finally, this study lacks real-world application and validation. Although the experimental results suggest the feasibility of CPI-ARP, its performance in actual logistics operations still needs further evaluation.

As CPI and logistics networks continue to evolve, we need to explore the optimization and expansion of address management in the CPI network environment further. First, regarding protocol security, although CPI-ARP addresses basic functional requirements for address resolution, researchers must further strengthen its ability to counter security threats as network size expands and device types diversify. Second, the PMAC and PIP address systems proposed in this study have demonstrated significant flexibility and adaptability in dynamic addressing optimization. However, future research should focus on developing intelligent methods to allocate and manage these addresses in more complex logistics scenarios. Researchers can integrate artificial intelligence (AI) and machine learning (ML) technologies with dynamic optimization algorithms to enhance address allocation and routing efficiency. These algorithms can optimize address allocation and routing based on device usage frequency, location changes, and task priorities, thereby improving the CPI network's overall efficiency and resource utilization. Finally, although this paper focuses on experimentally validating and evaluating the feasibility and effectiveness of CPI-ARP and physical address encoding, researchers must address potential practical challenges, such as standardization and policy issues, before large-scale deployment.

REFERENCES

[1] H. Wu, L. Huang, M. Li, and G. Q. Huang, "Cyber-physical internet (cpi)-enabled logistics infrastructure integration framework in the greater bay area," *Advanced Engineering Informatics*, vol. 60, p. 102551, 2024.

[2] B. Montreuil, "Toward a physical internet: meeting the global logistics sustainability grand challenge," *Logistics Research*, vol. 3, pp. 71–87, 2011.

[3] E. Ballot, *The physical internet*. Springer, 2019.

[4] H. Tran-Dang, N. Krommenacker, P. Charpentier, and D.-S. Kim, "Toward the internet of things for physical internet: Perspectives and challenges," *IEEE internet of things journal*, vol. 7, no. 6, pp. 4711–4736, 2020.

[5] X. Qu, M. Li, Z. Ouyang, C.-l. Ng, and G. Q. Huang, "Routing protocols for b2b e-commerce logistics in cyber-physical internet (cpi)," *Computers & Industrial Engineering*, p. 110293, 2024.

[6] D. Hercog and D. Hercog, "Arp protocol," *Communication Protocols: Principles, Methods and Specifications*, pp. 321–322, 2020.

[7] H. Ning, Z. Zhen, F. Shi, and M. Daneshmand, "A survey of identity modeling and identity addressing in internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4697–4710, 2020.

[8] S. Pan, E. Ballot, G. Q. Huang, and B. Montreuil, "Physical internet and interconnected logistics services: research and applications," pp. 2603–2609, 2017.

[9] M. M. Alani, "Guide to osi and tcp/ip models," 2014.

[10] H. Zimmermann, "Osi reference model-the iso model of architecture for open systems interconnection," *IEEE Transactions on communications*, vol. 28, no. 4, pp. 425–432, 2003.

[11] B. Montreuil, E. Ballot, and F. Fontane, "An open logistics interconnection model for the physical internet," *IFAC Proceedings Volumes*, vol. 45, no. 6, pp. 327–332, 2012.

[12] B. Montreuil, R. D. Meller, and E. Ballot, *Physical internet foundations*. Springer, 2013.

[13] Y. Song, F. R. Yu, L. Zhou, X. Yang, and Z. He, "Applications of the internet of things (iot) in smart logistics: A comprehensive survey," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4250–4274, 2020.

[14] S. Gontara, A. Boufaied, and O. Korbaa, "Routing the pi-containers in the physical internet using the pi-bgp protocol," in *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*. IEEE, 2018, pp. 1–8.

[15] M. Briand, R. Franklin, and M. Lafkihi, "A dynamic routing protocol with payments for the physical internet: A simulation with learning agents," *Transportation Research Part E: Logistics and Transportation Review*, vol. 166, p. 102905, 2022.

[16] U. Garg, P. Verma, Y. S. Moudgil, and S. Sharma, "Mac and logical addressing (a review study)," *Int. J. Eng. Res. Appl.(IJERA)*, 2012.

[17] D. Skordoulis, Q. Ni, H.-H. Chen, A. P. Stephens, C. Liu, and A. Jamalipour, "Ieee 802.11 n mac frame aggregation mechanisms for next-generation high-throughput wlans," *IEEE Wireless Communications*, vol. 15, no. 1, pp. 40–47, 2008.

[18] J. Jeong, Y. Shen, S. Jeong, S. Lee, H. Jeong, T. Oh, T. Park, M. U. Ilyas, S. H. Son, and D. H. Du, "Stmac: Spatio-temporal coordination-based mac protocol for driving safety in urban vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1520–1536, 2017.

[19] M. Zhang, G. M. N. Ali, P. H. J. Chong, B.-C. Seet, and A. Kumar, "A novel hybrid mac protocol for basic safety message broadcasting in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4269–4282, 2019.

[20] J. Wu, H. Lu, Y. Xiang, F. Wang, and H. Li, "Satmac: Self-adaptive tdma-based mac protocol for vanets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 21 712–21 728, 2022.

[21] L. L. Peterson and B. S. Davie, *Computer networks: a systems approach*. Morgan Kaufmann, 2007.

[22] K. Igulu, F. Onuodu, and T. P. Singh, "Ipv6: Strengths and limitations," in *Communication Technologies and Security Challenges in IoT: Present and Future*. Springer, 2024, pp. 147–172.

[23] B. Feng, H. Zhang, H. Zhou, and S. Yu, "Locator/identifier split networking: A promising future internet architecture," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2927–2948, 2017.

[24] J. Klensin, "Simple mail transfer protocol," Tech. Rep., 2008.

[25] J. Postel and J. Reynolds, "Rfc0959: File transfer protocol," 1985.

[26] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, and P. Tran-Gia, "A survey on quality of experience of http adaptive streaming," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 469–492, 2014.

[27] D. Naylor, A. Finamore, I. Leontiadis, Y. Grunenberger, M. Mellia, M. Munafò, K. Papagiannaki, and P. Steenkiste, "The cost of the" s" in https," in *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*, 2014, pp. 133–140.

[28] J. Arkko and C. Pignataro, "Iana allocation guidelines for the address resolution protocol (arp)," Tech. Rep., 2009.

[29] W. Lootah, W. Enck, and P. McDaniel, "Tarp: Ticket-based address resolution protocol," *Computer networks*, vol. 51, no. 15, pp. 4322–4337, 2007.

[30] S. Y. Nam, D. Kim, and J. Kim, "Enhanced arp: preventing arp poisoning-based man-in-the-middle attacks," *IEEE Communications Letters*, vol. 14, no. 2, pp. 187–189, 2010.

[31] M. Ataullah and N. Chauhan, "Es-arp: an efficient and secure address resolution protocol," in *2012 IEEE Students' Conference on Electrical, Electronics and Computer Science*. IEEE, 2012, pp. 1–5.

[32] S. M. Morsy and D. Nashat, "D-arp: An efficient scheme to detect and prevent arp spoofing," *IEEE Access*, vol. 10, pp. 49 142–49 153, 2022.

[33] R. Droms, "Automated configuration of tcp/ip with dhcp," *IEEE Internet Computing*, vol. 3, no. 4, pp. 45–53, 1999.

[34] C. Adams, "Replay attack," in *Encyclopedia of Cryptography, Security and Privacy*. Springer, 2021, pp. 1–2.

Wenqing Lei received the M.S. degree in Information Systems from Northeastern University, USA. He is currently pursuing the Ph.D. degree in Industrial and Manufacturing Systems Engineering with the University of Hong Kong. His research interests include kernel services, digital twin, and asset management of Cyber-Physical Internet.

Ming Li (Senior Member, IEEE) received his bachelor's degree in Computer Science from South China University of Technology in 2012 and master's and Ph.D. degrees in Industrial and Manufacturing Systems Engineering from the Department of Data and Systems Engineering, The University of Hong Kong, in 2013 and 2018, respectively. He is currently a Research Assistant Professor with the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University. His research interests are blockchain and cyber-physical systems in smart manufacturing and logistics. He has authored or co-authored more than 70 papers in international journals and top conferences, including IEEE T-II, IJPE, JMS, RCIM, etc. Dr. Li is the Core Member of the 2019 Guangdong Special Support Talent Program – Innovation and Entrepreneurship Leading Team. He is also a Senior Member of CCF and a Member of ASME and IISE, as well as Co-found of CommaTech.

Yong-Hong Kuo (Member, IEEE) received B.Sc. in Mathematics and M.Phil. and Ph.D. in Systems Engineering and Engineering Management from the Chinese University of Hong Kong. During the period, he also worked at the University of California at Berkeley as Visiting Researcher and Oak Ridge National Laboratory as Visiting Student. Prior to joining HKU, he was Research Assistant Professor at Stanley Ho Big Data Decision Analytics Research Centre, the Chinese University of Hong Kong. His research spans theory and application of mathematical modeling and optimization techniques for decision problems encompassing the field of management science. He is interested in modeling problems, devising effective and efficient solution methodologies and by using these tools to improve operations, help system design and derive managerial insights. His research, with his role as Principal Investigator, has been supported by a number of funding agencies, including Hong Kong Research Grants Council, Health and Medical Research Fund, Microsoft Research Asia, and Macao Science and Technology Development Fund.

George Q. Huang (Fellow, IEEE) received the B.E. degree from Southeast University, Nanjing, China, in 1983, and the Ph.D. degree from Cardiff University, Cardiff, U.K., in 1991, both in mechanical engineering. He joined the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong, in December 2022 as Chair Professor of Smart Manufacturing. Prior to this appointment, he was Chair Professor of Industrial and Systems Engineering and Head of Department wih the Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong. He has published extensively, and his works have been highly cited by research communities. He has conducted research projects in areas of Smart Manufacturing, Logistics, and Construction through IoT-enabled Cyber-Physical Internet and Systems Analytics. His research has been supported with substantial government and industrial grants exceeding HK$100M. Mr. Huang is the Associate Editor and Editorial Member for several international journals. He is a Chartered Engineer (CEng) and a fellow of ASME, CILT, HKIE, IET, and IISE.