Requirements-Driven Automotive Electrical/Electronic Architecture: A Survey and Prospective Trends

HAILONG ZHU[1,2], **WEI ZHOU**[3,4], **ZHIHENG LI**[3,4], **(Member, IEEE), LI LI**[3], **(Fellow, IEEE), AND TAO HUANG**[1,2], **(Senior Member, IEEE)**

[1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunication, Beijing 100876, China
[2]Purple Mountain Laboratories, Nanjing 211111, China
[3]Tsinghua Shenzhen International Graduate School, Shenzhen 518055, China
[4]Department of Automation, Tsinghua University, Beijing 100084, China

Corresponding author: Zhiheng Li (zhhli@mail.tsinghua.edu.cn)

ABSTRACT The automotive E/E architecture has undergone a paradigm shift in the past century. Particularly, the new requirements of automated driving are severely challenging the existing architecture, which has led to ongoing revolutionary innovation in E/E architecture. In this paper, we reviewed the evolution of E/E architecture and outlined the requirements-driven developments. We illustrated the state-of-the-art E/E architecture, including network topology, standards, simulator and software platform. We also discussed the next generation of E/E architecture from the perspective of different OEMs and suppliers. The analysis shows that software-defined, hierarchical, reconfigurable and customized E/E architecture is universally accepted. Furthermore, the automotive industry has been experiencing several transitions related to OEMs and suppliers. With the emergence and maturation of automated driving, we analyzed the new requirements on E/E architecture and proposed prospective development trends.

INDEX TERMS Automotive E/E architecture, automated driving, TSN, AUTOSAR, future trends.

LIST OF ABBREVIATIONS

ADAS	Advanced Driving Assistance System
BMCA	Best Master Clock Algorithm
CAN	Controller Area Network
DCU	Domain Control Unit
ECU	Electronic Control Unit
E/E	Electrical/Electronic Architecture
gPTP	generic Precision Time Protocol
HMI	Human Mechine Interface
IVN	In-Vehicle Network
LIN	Local Interconnect Network
MOST	Media Oriented System Transport
OEM	Original Equipment Manufacturer
OTA	Over-the-air
SOA	Service-Oriented Architecture
TSN	Time-sensitive networking
TTE	Time Triggered Ethernet

I. INTRODUCTION

Driven by the global trends of "connected, automated, electrified, and shared", a fundamental change in automotive E/E architecture is currently underway. In the early days, automobiles were dominated by mechanics. Driven by the development of electronic devices and electronic control units (ECUs), the electronic components of automobiles have grown rapidly, which has led to continuously increasing complexity of automotive E/E architecture. In recent decades, the electronic system of cars has developed into a highly complex network with dozens of dispersed ECUs, or even hundreds in luxury automobiles [1]. In the evolution from mechanical devices into mobile servers with complex functions and high interconnection, automobiles have experienced continuous improvements related to power,
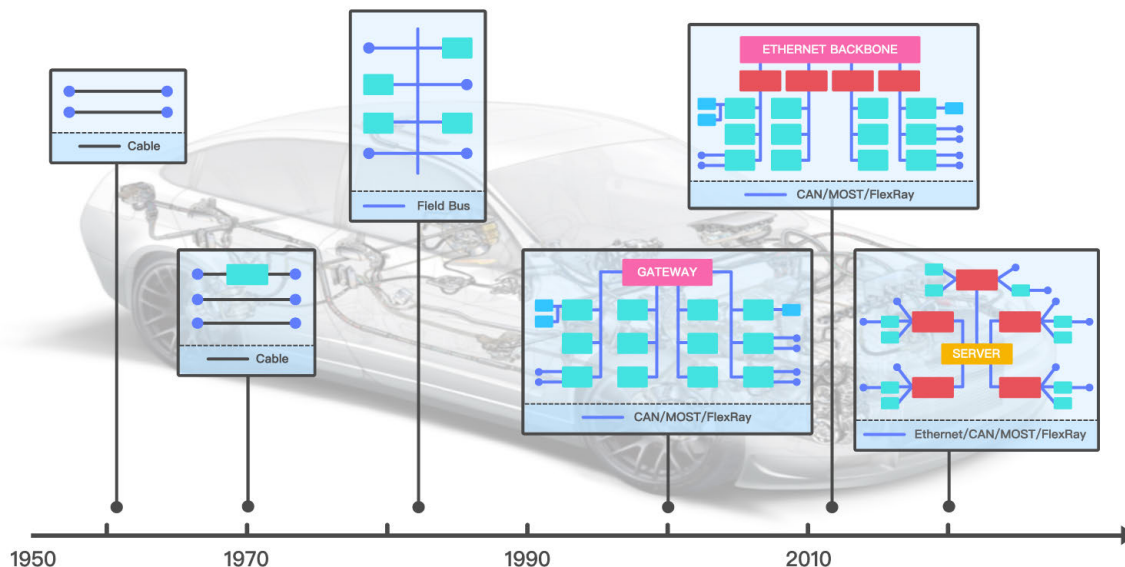
FIGURE 1. Evolution map of E/E architecture.

fuel economy, functional safety and comfort. In this process, the E/E architecture has gradually evolved and undergone a paradigm shift in the last decades. We have summarized our survey of the historical evolution in terms of the automotive E/E architecture, as illustrated in Fig. 1. Details will be presented in the following sections.

The E/E architecture has been recognized as the key foundation for supporting automotive applications and the increasing complexity of automobiles. Generally, the automotive E/E architecture is a network infrastructure, used to interconnect and organize ECUs and mechanical/electronic components. A previous article defined the E/E architecture as the fundamental organization of automotive E/E components, including ECUs, sensors, power systems and wire harnesses, to realize the expected functions, especially the mutuality and interactions among these elements from different perspectives [2]. Given this definition, the E/E architecture is considered from the following three perspectives, the physical topology, i.e., the packaging, placement and wire routing, logical topology with emphasis on the interactivity among these elements, and the guidelines and principles for architecture design.

In a recent survey, Navale *et al.* presented an overview of the evolution and revolution of automotive E/E architectures [3]. The bottlenecks of current the E/E architecture were discussed, and the future of E/E architecture was envisioned based on new technology trends. Additionally, functionalities, such as automated driving, and their implementation in mainstream automobiles were expounded. Jiang also studied the evolution of automotive E/E architecture from the perspective of the latest technical trends, such as electrification, automated driving, and connectivity [2]. Zeng *et al.* reviewed the five most widely used in-vehicle protocols, namely, LIN,

CAN, FlexRay, Ethernet and MOST, from the perspectives of system cost, transmission capability and fault-tolerance [4]. Huang *et al.* investigated in-vehicle protocols from both wired and wireless perspectives, then identified the challenges to current solutions [5].

In recent years, automated driving and networked automobiles have been very popular research filed. Although the existing in-vehicle communication network based on CAN and FlexRay was sufficient to meet the requirements of communication bandwidth in the past few years, the increased level of intelligence and automated driving, as well as the escalation of in-vehicle infotainment systems, have made it increasingly difficult for traditional in-vehicle networks to support and effectively handle the growing requirements for high-speed, high-bandwidth communication [6]. On one hand, massive data has driven the emergence of new in-vehicle network architectures. In the early 2010s, Ethernet was gradually recognized as a potential in-vehicle communication protocol [6], [7].The Ethernet's security issues, such as authenticated access and data encryption standards, have also become a concern [8]. On the other hand, data sharing and coordinated control between automotive systems is becoming more common, especially for automated driving, which has promoted the integration of computing units and domains.

These new trends and requirements challenge the existing E/E architecture. Thus, a suitable E/E architecture for new technology trends should be developed. Hence, this paper reviews the development of E/E architecture in the past century from a novel requirement-driven perspective and provides a comprehensive view of the evolution. Focusing on the new changes to the E/E architecture driven by automated driving, recent progress from both academic and industrial

perspectives are summarized. Finally, a possible future E/E architecture is proposed.

This paper is organized as follows. First, the design method and evaluation of E/E architecture is introduced in Section II. The requirement-driven evolution of E/E architecture is demonstrated in Section III. The changes to E/E architecture driven by automated driving are introduced, and the standards, OEMs, suppliers, IT companies and simulation platforms are discussed in Section IV. On the basis of above analyses, potential future E/E architectures are discussed in Section V. Finally, this paper is concluded in Section VI.

II. THE DESIGN AND EVALUATION OF E/E ARCHITECTURE
A. THE DESIGN METHODOLOGY AND PROCEDURE OF E/E ARCHITECTURE

Scientifically, two main stabilization mechanisms exist in the design of E/E architecture, the bottom-up, hardware-oriented approach and top-down, software-oriented approach [9]. The bottom-up approach refers to a design method that is derived from the existing architecture and developed by adding new devices and functions. By contrast, the top-down approach involve the entire top-down design process and starts with the functional requirements analysis [10].

Since the bottom-up design method is based on the existing E/E architecture, the cost and difficulty of the development process are relatively lower. However, this approach is limited by the original framework. This method is also relatively limited when designing large distributed software-intensive systems. The top-down design method is a new design process that follows complete development steps and is typically used when developing new vehicle platforms, so it is costly and time consuming.

For the top-down design process, in the initial stage of the design, it is necessary to define the requirements and decompose the functions. The design of E/E architecture is closely related to its functionality, especially the division and design of domains. Two architectures, i.e., the functional architecture and technical architecture, are often used in modeling E/E systems [9]. The separation of the functional architecture (logical level) from the specific implementation details enables the establishment of a complex functional architecture in the redesign stage without implementation restrictions. Then, according to different functional requirements, functional modules are divided. Stolz *et al.* proposed that domains should be divided into groups that are as large as possible while maintaining synergies by means of functional and nonfunctional criteria, such as the functional dependency, innovation speed and technology similarity [10].

Moreover, when the functional modules are divided, the corresponding specific design requirements must be determined and how to implement the design at the software and hardware levels must be considered. The hardware layer also involves the physical topology of the in-vehicle network, which is closely related to the weight of the wiring harness, wiring pattern and cost.

B. EVALUATION OF E/E ARCHITECTURE

In the process of designing and selecting an E/E architecture, many trade-offs must be considered, and a reasonable evaluation method should be proposed to establish criteria for the evaluation and decision making. Generally, the indicators of architectural design include many complex factors, such as the physical component package size, layout, energy efficiency, architecture reliability, scalability, reusability, and cost. Many studies have focused on these various trade-off indicators [11]–[14].

Specifically, as a market competitive product, automobiles are largely driven by production costs. Depending on the market segment, the E/E architectures can be quite different. For example, [15] discussed the E/E architecture of Brazilian B entry automobiles. The E/E architecture of South America markets in the 2000s did not contain power window or door locking features, which was entirely different from typical architectures. Vehicles named Celta, Palio and Fox contained only 4 or 5 ECUs, and no network protocol or only low-cost 10.4 kbps was used in these vehicles.

The design of the next-generation E/E architecture will be largely related to automated driving. The function of automated driving, the compatibility of configurations, and the costs will influence and determine the next-generation E/E architecture. Different OEMs may have inconsistent positioning and technical maturity in automated driving systems, which may lead to different solutions in the next-generation architecture. However, the complexity of automotive architecture has driven the need for collaboration with OEMs and suppliers. By means of increased cooperation to formulate automotive industry standards, OEMs and suppliers will jointly develop the next-generation automotive E/E architecture [16].

III. THE REQUIREMENTS-DRIVEN EVOLUTION OF E/E ARCHITECTURE

Different from the aforementioned reviews, the evolution of E/E architecture is surveyed from the perspective of requirements in this paper. Requirements have been the essential power motivating the development of automotive industry, and various types of E/E architecture have been designed to satisfy the ever-increasing requirements.

The development of automobiles shows different characteristics in different stages. E/E architecture, as the infrastructure for the connection and organization of automotive electronic components, provides a basic guarantee for the normal operation of automobiles. The contradiction between requirements and the existing circumstances has promoted the technological progress and revolution of E/E architecture. The requirements that drive the evolution of E/E architecture include the following.

- Interconnection of ECUs.
- Improvement of automobile performances, such as braking, stability, safety and economy.
- Reduction in the cost and weight in automotive manufacturing.

- High bandwidth and low latency of in-vehicle communication.
- Connectivity with the outside.
- Flexible decoupling of hardware and software.

The market requirement has promote industry reform, and the requirement of new technology brought has created a new direction for academic research. Academic progress, in turn, has driven industry progress.

A. POINT-TO-POINT ARCHITECTURE

In 1912, Dayton Electrical Laboratory Company developed the first starter motor, and the advent of the first car with an electric start function marked the beginning of a new era in the history of automotive electrical systems. Furthermore, the first 7V DC generator was developed to charge batteries and provide energy to lighting and ignition systems, and in 1955, the electrical system began to transition to 12/14 V [17].

At the beginning of 1970s, the automotive industry entered an era of electrification. Electronic devices were equipped in automobiles to partially replace mechanical components, such as electronic ignition devices with transistors and diode-rectified alternators [9]. At this point, E/E architecture was almost unknown owing to the extremely small number of electronic devices in automobiles. Moreover, electrical systems, which has gradually replaced mechanical or hydraulic systems, have increased the requirements for electrical energy.

The emergence of integrated circuits (ICs) sparked a landmark revolution in automotive electrical systems. The rapid expansion of ICs, from small-scale applications to very large-scale integration (VLSI) promoted the formation of large-scale automotive networks, and ECUs were gradually applied in automobiles with the applications such as engines and airbags [2].

In this period, these electronic devices substantially improved the automobiles performance. Connections of these devices was point to point and realized the transition from totally isolated to connected systems in a stepwise manner.

B. ARCHITECTURE EQUIPPED WITH VEHICLE BUS

Since the 1980s, the electronic progression of automobiles has gradually accelerated, and requirements for automotive safety, emissions and energy saving have become increasingly stringent. Within a few years, considerable electronic control systems, such as electronic fuel injection, ignition and emission system, anti-lock braking, airbag system, anti-collision system, GPS navigation and fault diagnosis, were applied. The emerging electronic technology has almost replaced the original electro-hydraulic control system. Furthermore, the microprocessor has been widely used in automobiles to achieve these specific functional requirements, and the control systems of the vehicles have become more comprehensive, which has improved the control precision and performance.

However, complexity of the automotive control system and wiring harness increase exponentially with the point-to-point connection. The wiring harness was too large and complicated to ensure the maintainability and reliability with the former wiring method. To simplify the wiring harness and improve communication efficiency, field bus were adopted. The number of communication lines around the vehicle can be effectively reduced by adopting the field bus as communication system. The signal transmission between all ECUs can be completed through a few buses, and precise control of vehicle performance can be realized via communication and coordinated control.

Manufacturers developed their own bus technologies initially, but the inconsistency made assembly and packaging difficult. The standardization and modularization of the data bus system made the design and reconstruction of automotive networks easier. BOSCH GmbH introduced the first fieldbus, the controller area network (CAN-bus), in 1983 [18], which was standardized in the beginning of the 1990s and has been widely used since then [19].

Subsequently, various kinds of field buses, such as the local interconnection network (LIN), media-oriented system transport (MOST) and FlexRay, were developed and applied in vehicles to satisfy different communication requirements in different vehicle domains. From 20 kbps to 20 Mbps, the vehicle bus makes the E/E architecture more cost effective, and standardization of the vehicle bus has ensured interoperability and consistency and promoted the assembly and mass production of vehicles.

C. ARCHITECTURE EQUIPPED WITH CENTRALIZED GATEWAY

To make full use of the respective advantages and constitute a low-cost and fully functional automotive network to improve automobiles performance, different field buses for different control systems and gateway-connected solutions are generally accepted.

Consequently, designing an appropriate E/E architecture has gradually become critical [20]. E/E architectures with a centralized gateway connected to different subnetworks were commonly applied until relatively recently. This kind of architecture is adopted by many manufacturers for various automobiles, such as the Volkswagen Passat [4], BMW 7 series [21] and Audi [22].

In terms of the power system, the load is connected directly to the battery using a switch or an electromechanical relay. This structure has led to complex and bulky wiring harnesses, and the addition of new electronic components further entails higher wiring complexity, longer and heavier cables, and higher costs.

With the increased implementation of high-power applications, such as electric heating and electric pumps, the requirement for electricity has continued to increase. The traditional 12 V power can hardly provide sufficient power for such applications. Thus, in the 2000s, 48 VDC-36 VDC high-voltage electrical systems were widely

discussed [23], [24]. The dual-voltage power supply network became a general architecture to smoothly transition to high-voltage networks. Since the early 1990s, international associations, such as MIT and SAE, had been promoting regular meetings with car manufacturers. They believed that the establishment of a 42 V DC voltage bus was the future standard for the automotive industry [17].

At the same time, the junction boxes in the power distribution system were replaced by intelligent power distribution units (PDUs). Compared to the junction box, the PDU adds intelligent functions, such as fault diagnosis and power management. Additionally, PDU modularizes the E/E system to reduce the complexity of the wiring harness [24].

Decentralization of power distribution gradually became popular. Compared to centralized distribution, decentralization reduces the wiring harness, thereby improving the weight of the system. Practical applications in Europe's popular cars verified these improvements in circuit quantity, length, weight, and cost [23]. The optimal number, location, and content of distribution boxes also needed to be determined, and several arrangements of distribution boxes are discussed in [25].

In terms of reducing the complexity of power network, several initiatives have been applied, for example, replacing traditional fuses with maintenance-free solid state fuses that can be placed in key locations instead of easily accessible locations, thereby reducing the complexity of the power supply network [3].

The central gateway architecture improves the security of the automotive bus system. Moreover, the gateway converts different protocols and regulates network traffic. The increase in electronic devices and the power requirement made the power supply system more important, and the application of high-voltage systems and PDUs has improved the efficiency of power supplies and reduced the complexity of power distribution systems.

D. DOMAIN-BASED ARCHITECTURE

ECUs typically support only one specific application, such as electronic power steering. Only a few components are cooperative, such as the combination of brakes and airbag ECUs. In addition, the sensor data and algorithms are processed locally in each dedicated ECU [26].

With the ever-increasing number of automated driving functions, the traditional method of adding a new ECU for each new function will increase the complexity of the E/E architecture. Excessive distribution leads to complex wiring, poor networking performance and a reduction in the communication speed of critical information. In the previous central gateway–based architecture, all components and sensors are connected with the gateway. Both intradomain and cross-domain communication pass through the gateway, which leads to a high load of the gateway, especially when the requirements of bandwidth and latency increase [30]. Furthermore, breakdown of the centralized gateway can easily lead to collapse of the entire network [4].

Considering the bottleneck problems of ECUs and information security, domain control units (DCUs), such as Bosch, Continental and Delphi, were proposed by Tire1 [28]. The application of a domain controller relieved some of the stress on the system. However, due to the increased complexity and cost, domain-controlled architecture and an Ethernet backbone are critical for future E/E architecture development.

The domains provide appropriate isolation of the ECU based on similar functions and communication access. Each domain integrates a portion of the ECUs that communicate in the bus system. In addition, the gateway interconnects different networks and coordinates ECUs in different domains to communicate [16]. In general, DCUs contain OEMs' specific software components [29], and DCUs reduce the material costs and weight in automotive manufacturing [30].

Currently, domain-based E/E architecture is applied in most newly unveiled cars. OEMs had different but similar classification criteria of automotive domains based on the specific functions, and E/E architectures are commonly divided into body, powertrain, chassis, infotainment, ADAS and telematics domains [31], [32]. In addition, the ADAS domain gradually developed and became one of the fastest growing areas. ADAS DCUs usually require powerful computing processors that are supplied by various solution providers, including NVIDIA, Infineon, Renesas, TI, NXP and Mobileye [28].

Domain-based architecture is an inevitable consequence of excessive decentralization. The application of a domain controller effectively improved the gateway load and ECU bottleneck problems. An important driver of this architecture is the automotive Ethernet and switches. However, when local processing is not performed, additional delays that may be introduced between the sensor and the actuator must be considered. Therefore, a guaranteed determined delay may be required for automobiles [26].

E. ZONE-BASED ARCHITECTURE

To reduce the wiring cost and weight, Brunner proposed the zone-based architecture, where in addition to functional domain clusters, components can be integrated based on their physical location in the vehicle [27]. In this way, the components around the vehicle are no longer classified by function, and automotive software and hardware platforms must be reconsidered. Reducing engineering time and cost by reusing components and smooth integration is important.

The architecture is also beneficial for power supply. The area controller can provide both communication and power supply for corresponding peripheral components [27]. At present, different networks are applied for communication and power supply; however, the introduction of Ethernet can combine these networks and reduce cables and costs, providing greater flexibility in the network layout.

Power over Ethernet (PoE) is a technology that provides power while an Ethernet cable transmits data signals. After the first PoE standard IEEE 802.3af (15.4 W) [33] was proposed in 2003, PoE + (IEEE 802.3at) [34], which
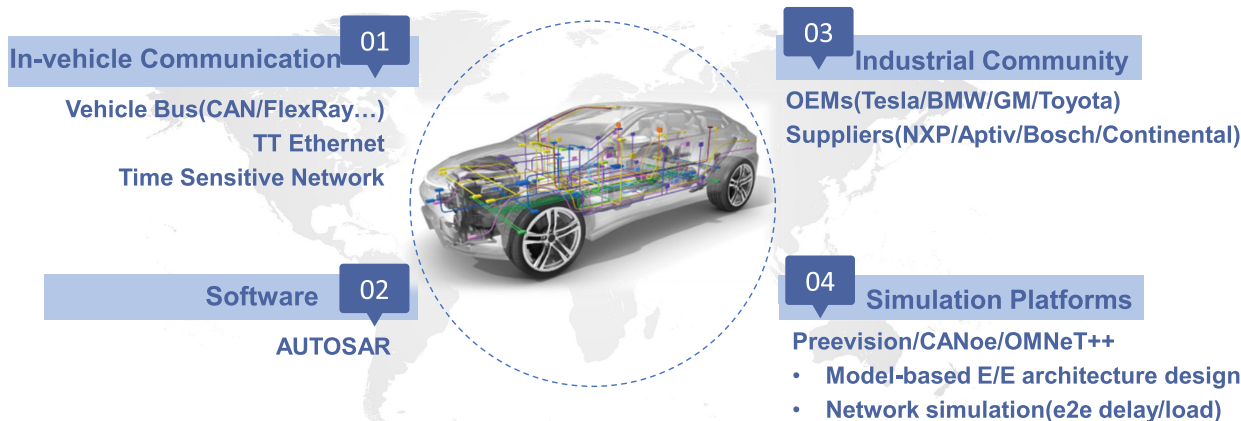
FIGURE 2. State-of-the-art E/E architecture.

supports higher power (25.5 W), was proposed. However, PoE is not entirely suitable for the application of automotive Ethernet. Conventional PoE is designed for 4-pair cable Ethernet, and the automotive Ethernet physical layer protocol BoardR-Reach uses one pair of un-shielded twisted-pair cables. Therefore, PoE cannot be used directly on the vehicle Ethernet. The IEEE has developed a new power supply standard called power over data lines (PoDL) for automotive and industrial applications, as defined in IEEE 802.3bu [35].

The zone-based architecture is greatly beneficial in reducing cabling, especially for Ethernet, with the combination of communication and power. However, the architecture entails higher requirements for software platforms owing to the location clustering.

IV. STATE-OF-THE-ART E/E ARCHITECTURE

Automated driving was divided into 6 levels by SAE. Different levels correspond to different degrees of vehicle automation. The functions of human drivers are gradually taken over by the automotive system, and systems therefore need more sensing capabilities and greater decision-making intelligence. In the process of continuously improving of autonomous intelligence, the improvement in perception and decision-making capabilities requires the support of the underlying architecture. The vehicle requires large-bandwidth, low-latency, scalable and flexible E/E architecture.

In this way, the existing bottlenecks of automotive E/E architecture can be summarized as follows:

1) Bandwidth bottleneck: The sudden increase in data volume caused by the increase in sensors has limited the bandwidth of traditional in-vehicle networks.
2) Wiring complexity: The wiring complexity and wiring harness weight brought by the increase of nodes have a negative impact on vehicle maintenance, weight reduction, power performance and economy.
3) Low deterministic latency: Intra-domain and cross-domain communications increase, and existing bus

technologies are difficult to meet the requirements of low deterministic latency in the case of high concurrency and high traffic.
4) Flexible architecture and scalability: It is difficult to achieve online upgrade and maintenance of vehicle equipment, scalability, and dynamic network configuration requirements.

In order to solve the above bottlenecks, standardization organization, the automotive industry including OEMs and suppliers,and academia are committed to developing and innovating E/E platforms,as shown in Fig.2, which will be illustrated below.

A. IN-VEHICLE COMMUNICATION STANDARDS

1) VEHICLE BUS

Since the 1990s, in-vehicle network has gradually matured and become widely used. Vehicle manufacturers first developed their own bus technology, but the inconsistencies between products made it difficult to assemble and debug vehicles. Therefore, the standardization and modularization of data bus systems are gradually being motivated to better promote the design and construction of automotive networks. There are many different standards for in-vehicle networks.

According to the transmission speed of different communication protocols, SAE divides the in-vehicle network bus into classes A-D. Each type contains various protocols. Some are still widely used, and some have been gradually phased out [36].

Class A contains UART, Sinebus, LIN, etc. Among these protocols, LIN is widely used in automobile body control, such as the door, lighting and other occasions with low real-time requirements. Low-speed CAN, SAE J1580, etc. can be summarized as class B protocols. High-speed CAN, SAE 1939 (based on CAN2.0), FlexRay, ByteFlight, etc. can be classified as Class C. Class C is mainly used in automotive real-time control systems, such as engine and chassis control,

TABLE 1. Comparison with different E/E architectures.

Type	Features	Strength	Weakness
Point-to-point architecture	Electronic devices were connected directly with the actuator.	Isolated mechanical or hydraulic systems were connected.	Only few components were electrified and connected
Vehicle bus based	The signal transmission between all ECUs can be completed through a few buses	Standardization and modularization of vehicle bus made the E/E architecture easier	A great variety vehicle bus and the data transmission speed were low
Centralized gateway based	Different subnetworks with different vehicle buses were connected with a centralized gateway	The gateway converts different protocols and regulates network traffic, security of the automotive bus system is improved	As the number of ECUs increase, functions of the gateway could be very complex, which is harmful for security
Domain-based	DCUs integrate a portion of the ECUs, contain specific software components and reduce the material costs and weight in automotive manufacturing	DCU effectively improved the gateway load and ECU bottleneck problems, the complex and cost could be decreased	Additional delays that may be introduced between the sensor and the actuator.
Zone-based	Electronic components can be integrated based on their physical location in the vehicle.	Zone-based architecture is greatly beneficial in reducing cabling, especially for Ethernet, with the combination of communication and power.	The architecture entails higher requirements for software platforms owing to the location clustering.

safety control and other systems. And the protocol MOST is summarized as Class D and used in automotive multimedia applications.

With the development of automotive electronics, new requirements for deterministic, real-time, and fault-tolerant have been proposed for traditional in-vehicle networks. In order to meet these needs, different from the event-triggered mode, the concept of time-triggered networks was proposed to improve the performance of in-vehicle networks, such as TTP(Time Triggered Protocol) [37], TTCAN [38], FlexRay, and TTEthernet(Time Triggered Ethernet) [39].

2) TIME TRIGGLED ETHERNET
Considerable data caused by the development of intelligent driving are beyond the in-vehicle networks nowadays. Therefore, the network with higher bandwidth, lower latency and higher security, such as automotive Ethernet, is gradually gaining people's attention. Ethernet shows ob?vious advantages in terms of bandwidth requirements, openness, scalability, physical layer transmission costs, technology maturity and standardization [40]. At the same time, high-bandwidth Ethernet links can support software-driven architecture and centralized processing, while greatly reducing the need for the number of bus types.Ethernet was firstly deployed in the

diagnostic system [41]. Afterwards the increasing requirement of bandwidth in vehicles made the Ethernet-based network architecture popular for OEMs, such as GM, BMW, and Mercedes-Benz.

The OPEN SIG Alliance [42] that aims to establish BroadR-Reach as a standard for the physical layer of automotive Ethernet is developing and standardizing BroadR-Reach technology, such as 100BASE-T1, 1000BASE-T1. UTP Ethernet consists of a single pair of twisted-pair copper wires, which is small, flexible, lightweight, and cheap to manufacture. The IEEE 802.3 Ethernet working group has set up many standards groups on the physical layer of automotive Ethernet, such as IEEE 802.3 Greater than 10 Gb/s Automotive Ethernet Electrical PHYs Study Group [43].

Although Ethernet has been widely adopted, traditional Ethernet fundamentally lacks the deterministic quality of service (QoS) characteristics of end-to-end flow. Its uncertain delay and low reliability cannot meet the requirements of automotive control networks [44]. This is the major challenge for Ethernet to become an automotive communication standard. The main limitations of Ethernet's lack of support for deterministic low-latency applications are reflected in the following aspects: synchronization,

management mechanisms, policy enforcement mechanisms and flow control mechanisms [45].

To this end, Tier1 companies such as Bosch, NXP, Cisco and TTTech are gradually introducing automotive Ethernet related solutions and hardware. The real-time Ethernet standards such as Audio / Video Bridging (AVB), Time Triggled Ethernet(TTEthernet) and Time-Sensitive Networking (TSN) have been proposed to solve these problems in Ethernet technology and provide different levels of end-to-end bounded delay guarantees for different priority data in vehicles. In addition, organizations like AVnu Alliance [46] and OPEN Alliance [42] are gradually forming and participating in automotive Ethernet standardization projects to promote the use of automotive Ethernet.

TTEthernet, an extension of normal Ethernet, is a scalable and open real-time Ethernet platform. It was proposed by TTTech and mainly used in security Related applications in transportation and industrial automation. TTEthernet allows the time triggered traffic to coexist with low-priority standard event-triggered Ethernet traffic. Kopetz *et al.* [47] regarded TTEthernet as the combination of standard Ethernet and TTP/C. Three traffic classes consisting of time-triggered(TT), rate-constrained(RC) and best-effort(BE) traffic are included [127]. TT traffic is transmitted according to a TT schedule table. And time synchronization is necessary so that nodes can rely on schedules to complete the sending and receiving of time-triggered traffic, ensuring that time-triggered services can conflict-free transmit over the network. The current mainstream time synchronization methods include IEEE 1588 [49] and SAE AS6802 [50].

Steinbach *et al.* proved that TTE can provide comparable the latency and jitter as low as FlexRay while achieving much higher communication bandwidth than FlexRay [51].The time scheduling algorithm is the focus of TTE design. Due to the complex network topology, chaotic application requirements, and numerous hardware constraints in the actual control system, its complexity is determined by the network topology, traffic characteristic and related constraints such as hardware and application-oriented constraints [52], [53].Many scheduling mechanisms are proposed [53]–[56]. Steiner [53] introduced a kind of SMT solver. Firstly, the messages that need to be scheduled, the network topology, and the constraints is abstracted as the input of the SMT solver. Then the solver calculates whether the resource can be allocated in the corresponding physical network and then generate the corresponding configuration arrangement.

3) TIME-SENSITIVE NETWORKING

In addition, the IEEE 802.1 work group has developed the Audio / Video Bridging (AVB) standards to address some of the key issues in Ethernet technology. AVB is listed by the academic and industrial circles as another automotive Ethernet service assurance technology, originally developed for automobiles audio, video, and multimedia systems. Ethernet AVB is a collection of different standards. The Ethernet AVB protocol set includes IEEE 802.1 As (precise

clock synchronization protocol) [57], 802.1 Qav (queue and forwarding rule protocol) [58], 802.1 Qat (flow reservation protocol) [59],and 802.1 BA [60]. Based on traditional Ethernet networks, it provides high quality of service (QoS) through accurate clock synchronization, guaranteed bandwidth, and limited delay. It supports various audio-based, Network multimedia applications for video.

The IEEE's AVB working group was renamed the Time-Sensitive Networking Task Group in 2012. TSN evolved from AVB, also known as the second-generation AVB. Based on the revision of the original standard, it has more comprehensively expanded and improved the original AVB protocol set. The IEEE 802.1 TSN TG standard and protocol extends the traditional Ethernet data link layer standard to ensure data transmission with limited latency, low latency jitter, and extremely low packet loss, making it ideal for industrial control and automotive applications. Its application range extends from the original audio and video bridging network to various fields of high stability requirements such as industrial automation and automated driving. Through the combination of these protocol standards, TSN can complete the scheduling management of the network and provide excellent results. Table2 lists the main sets of existing TSN protocols and their functions by category. TSN protocols can be divided into four types which respectively focus on flow Synchronization [57], flow management [59], [61]–[63], flow control [58], [64]–[68] and flow reliability [69]–[71].

In the past few years, many timing analysis methods for Ethernet have been studied. For example, IEEE802.1Q [72] analyzes the end-to-end delay of standard full-duplex switched Ethernet, and [73] proposes a real-time calculation timing analysis technique and analyzes the Ethernet based on the automotive E/E architecture AVB. [74] proposed the worst-case analysis using strict priority and credit-based shaping algorithms in Ethernet AVB, and evaluated in a typical centralized topology use case in industrial automation. [75]–[77] proposed a network calculus-based analysis method for deterministic and delay range analysis of TT flow in TTEthernet.

Many researches have analyzed and compared these different protocols using traffic shaping methods. For example, Boiger [78]compared BLS and PS shapers, and Meyer *et al.* [79] proposed a network scenario to analyze the effect of TAS on AVB's Class A traffic. The author proposed a theoretical analysis of the maximum end-to-end latency of AVB Class A traffic and compares it with simulation results.Then, Thangamuthu *et al.* [80] made a detailed comparison of standard TSN traffic shaping methods. The article compared BLS, TAS, and PS shapers in detail and analyzed the worst-case end-to-end delay of AVB traffic and control traffic in small car network scenarios. Simulation shows that for a typical 100 Mbps Ethernet network, most applications meet vehicle-mounted latency requirements except for applications with strict latency requirements. However, the analysis in [80] has the limitation that does not consider the competition between the frames with the same priority.

TABLE 2. TSN protocols list.

Class	Protocol
Flow synchronization	IEEE 802.1AS:Time Synchronization
Flow management	IEEE802.1Qcp:YANG Data Model IEEE 802.1Qat:Stream Reservation Protocol (SRP) IEEE802.1Qcc:Enhancements to SRP and Centralization Management IEEE802.1CS:Link-Local Reservation Protocol (LRP)
Flow control	IEEE 802.1Qav:Forwarding and Queuing of Time Sensitive Streams IEEE802.1Qbv:Enhancements to Traffic Scheduling (Time-Aware Shaper (TAS)) IEEE802.1Qbu+IEEE802.3br:Interspersing Express Traffic (IET) and Frame Preemption IEEE802.1Qch:Cyclic Queuing and Forwarding (CQF) IEEE802.1Qcr:Asynchronous Traffic Shaping (ATS)
Flow reliability	IEEE802.1CB:Frame Replication and Elimination for Reliability (FRER) IEEE802.1Qca:Path Control and Reservation (PCR) IEEE802.1Qci:Per-Stream Filtering and Policing (PSFP)

In this regard, based on considering this limitation, Thiele's Different shapers of the topology such as TAS, PS [81], BLS [82] performed formal timing analysis and worst-case delay analysis to evaluate whether these shapers can meet the stringent requirements of automotive networks for deterministic low latency.

In addition to the shapers described above, other traffic control mechanisms including asynchronous traffic scheduling algorithm such as UBS [83] and ATS [84], [85], pre-shaping [86] and frame preemption [87], [88], [132] are analyzed, focusing on the worst-case performance such as end-to-end delay and jitter of critical traffic.

4) DISCUSSION

We concluded the communication protocols discussed above in table 3. Especially, considering about the application prospects of Ethernet, TTEthernet and TSN are discussed. Alderisi *et al.* [90] compared the performance of TTEthernet and AVB in ADAS, and the results show that both meet the vehicle requirements. But TTEthernet p erforms better on deterministic transmission due to the time trigger mechanism of TT stream, While due to the reservation mechanism AVB is more flexible in network configuration under changing bandwidth requirements. Compared with TTEthernet, TSN also provides a bandwidth reservation mechanism through 802.1Qat and 802.1Qcc. AS for flow control, the TT flow in TTEthernet is transmitted according to the TT schedule. In TSN, many shapers such as CBS, TAS, BLS, PS, and ATS are used to ensure the real-time requirements and the problem that low priority traffic is always blocked by high priority. TSN allows frame preemption for high priority traffic. Among these shapers, similar to the TT flow in TTEthernet, TAS uses a gate list to ensure that a specified set of traffic can access the network within a certain time interval. However, the scheduling configuration in TTEthernet is to provide a fixed sending time of TT frames for each node. TAS uses

TABLE 3. In-vehicle communication protocols.

Standard	Bandwidth	Delay	Application
LIN	20Kbps	10ms	Body/Comfort, such as Lights, door locks
CAN	1Mbps		Body/Comfort; Powertrain; Chassis, such as engine control, transmission system, ABS braking system, etc
FlexRay	10Mbps	200μs	Powertrain; Chassis
MOST	25Mbps		navigation, multimedia
AVB	25Mbps	2ms	multimedia
TTEthernet	10Gbps	200μs	multimedia, ADAS, backbone
TSN	10Gbps	200μs	multimedia, ADAS, backbone

a gate list to indicate the behavior of the corresponding queue, and the configuration is more flexible. In terms of redundancy, IEEE 802.1 Qca and 802.1CB protocols have been used in TSN to achieve redundancy. The above points reflect the advantages of TSN [91].

B. AUTOSAR

Automotive software is critical to develop consistent, reliable, and high-quality electronic controls and interfaces. Software is considered as a key strategic element that needs to be easily added with new features and software should be upgradable and maintainable [15]. Car-grade standards and requirements are needed in automotive industries.

The automobiles needs to integrate components developed by various suppliers. The specifications of these components are defined by OEMs. But the actual construction of the components is done by the supplier. These components usually come with their own electronic control unit (ECU) and software stack. These separate software stacks constitute a large amount of software in modern automobiles [92]. The complexity of E/E architecture is one of the challenges in automotive industry.

All parties involved in the software and hardware design of automobiles need to be tightly coupled. The overall design process is hindered not only by heterogeneous development systems, but also by the various modeling and specification languages of development tools used by different OEMs, Tier1, and Tier2. Reusability, testability, and security are needed to be addressed in the above systems [93]. Basic software, interface applications and data buses must be standardized [94].

Therefore, the increasingly complex automotive E/E architecture, the scalability and transferability of automotive production, the flexibility of product updates, and the reliability of E/E architecture, need to be improved [95]. Leading automotive manufacturers and suppliers have introduced AUTOSAR (Automotive Open System Architecture), which improves the complexity through reusable software components and possibly standard hardware components [96], [97]. These years more and more companies in the industry have joined the AUTOSAR alliance, including OEM (automotive manufacturers), Tier1 (auto parts suppliers), chip manufacturers. AUTOSAR has also become the future of automotive E/E design.

AUTOSAR Alliance aims to establish a standardized software architecture for automobiles. The goal of AUTOSAR is to increase the reuse of software components, especially between different automobile platforms, and between OEMs and suppliers. AUTOSAR enhances the transferability of system functions throughout the automotive industry, maintainability of the entire life cycle, including the integration of functional modules from different vendors, and software updates [92].

The AUTOSAR standard contains a set of specifications that describe software architecture components. These standard rules and interfaces are managed by standard description files established by the AUTOSAR Alliance throughout continuous modification and update. The AUTOSAR standard defines an architecture that separates application software from infrastructure-related basic software. It distinguishes three main software layers running on a microcontroller: application software, runtime environment (RTE), and basic software (BSW). The use of AUTOSAR allows the ECU functionality to be abstracted into the middleware layer. Using a common middleware framework means that applications can be developed once and deployed multiple times, saving development time and reducing complexity [92].

Simultaneously, the development of vehicle domain controllers is also supported and promoted by the standardization of the software architecture. The separation of software and electronic hardware is achieved through standardization of basic software and software interfaces [10]. In the process of migrating from the initial software to AUTOSAR, automotive OEMs gradually introduced AUTOSAR-compatible software components into the overall architecture [95], such as migrating from OSEK/VDX to AUTOSAR. OSEK/VDX was originally a joint project of the German automotive industry, which aimed at developing an open architecture of distributed systems in vehicles [98]. Many studies have focused on the migration process of AUTOSAR systems [92], AUTOSAR modeling and development methods [99]–[102], the challenges and problems faced by AUTOSAR [103]–[105].

The trend of autonomous driving also puts forward new requirements for the next-generation software platform including flexible and dynamic deployment of software applications, support of high-computing hardware, security, determinism, and real-time performance. These new requirements challenges the AUTOSAR platform [106]. As a global standardization alliance, in addition to the traditional classic software platform, AUTOSAR is also developing a second software platform-the AUTOSAR Adaptive Platform to adapt the tendency of autonomous driving [107].

C. INDUSTRIAL SOCIETY
1) OEMs
There is no universal standard E/E architecture among OEMs. But most of them are similar. The trend of high-performance computing architecture is the consensus. Increasing computing power will gradually lead to a central EE architecture. Intelligent vehicle architecture with high-performance central computing unit will replace the outdated distributed computing architecture.

Tesla has been working to reduce the wiring length of vehicles. The wiring of Model S is about 3 kilometers, and Model 3 has been shortened to 1.5 kilometers. However, Tesla is still developing a new wiring architecture for future vehicle platforms, hoping to reduce the Model Y to 100 meters [108]. Manual wiring makes the assembly a labor-intensive process in the automobile production. Tesla hopes to use robots to more easily manipulate the wiring harness system, thereby reducing costs. But more details of the Tesla's E/E architecture have not been disclosed.

GM has released a new generation of E/E architecture, mounted on newly-unveiled Cadillac CT5. The architecture is 5 times faster than the current speed with OTA function. The new architecture is equipped with 100 Mbps, 1 Gbps and 10 Gbps Ethernet. At the same time, it focuses on safety performance by benchmarking the architecture as an aviation system [109].

BMW introduced a layered software architecture. Top of the architecture is a central computing platform. The next level is the integrated control unit which integrated specific control functions. The next is the commodity control unit which is standard parts without specific OEM functions. The bottom of architecture is the sensor and actuator. BMW hopes

that all central computers have a unified software platform, which improves the reuse and reduces development and verification workload [110].

In Toyota's vision for next-generation architecture, a central and zone-based architecture is expected and recognized as ultimate goal of Physical E/E Architecture [111]. In this way, minimized wire, flexibility of new ECUs and additional software can be guaranteed. To ensure security, Toyota listed three key factors including isolation, OTA and individuality [112].

2) SUPPLIERS

With the development of ee architecture, the relationship between OEM and Tier1 is sustainably changing. The integration of ECUs has weakened the role of original ECU electronics suppliers, while DCU suppliers become increasingly important. At the same time, Tier1 also gradually cooperates with OEMs by providing hardware, middleware and system platforms. Meanwhile, OEMs are responsible for developing software systems related to autonomous driving [28]. Thus, many Tire1 companies are actively exploring the promising future E/E architectures. Under the trend of autonomous driving, they will seize the OEM market with technological maturity and foresight.

Tier1 suppliers choose to cooperate with chip suppliers for solution design and development of central domain controller, such as Continental ADCU (The Assisted & Automated Driving Control Unit) which is the safe & secure multipurpose processing platform [113]. Nvidia launches a scalable open autonomous driving computing platform NVIDIA DRIVE AGX [118]. And NXP developed the next generation S32x automotive processing platform [114]. The platform aims to maximizes hardware and software reuse across products and applications and enables over-the-air updates.

In the anticipation of NXP, 3+1 Compute elements of E/E architectures are proposed. 3 means sensor and actuator, domain and centralized computing, and 1 means cloud. The E/E architecture is motivated to domain-based architecture, which is composed of Body and comfort, powertrain and chassis, infotainment domain and high performance ADAS computer, with the gateway processor and Ethernet switch [114]. At the same time, Ethernet(Gbps) will be applied in backbone with CAN, FlexRay or 100Base-T1 used intra domains.

Aptiv proposed a smart automobile architecture, which is mainly composed of the brain, the nervous system and cloud. The brain is mainly composed of three supercomputers, including an autopilot supercomputer, a central location sensor super-computer, and a network-side security supercomputer [115].

The redundant design of Aptiv's intelligent vehicle architecture is extraordinary. Computer, the network or signal, and then the power, the redundant setting of these three elements makes the automotive safety and reliability fully guaranteed. Accidents can be avoided when the computer, network or energy fails [116].

Bosch listed the evolution of E/E architecture and envisages a centralized E/E architecture in the future [3]. With the increasing of software, the fusion of DCUs motivated the application of zone-based ECUs and vehicle computer. And then some functions in vehicle computer will be transferred into cloud computing.

Similarly, Continental put forward a centralized architecture. The core concept of this architecture is the centralized processing and "all-to-some" communication [117], which is driven by software.

Both OEMs and suppliers are looking for the possibility of the next-generation E/E architecture for more market share. In general, their road-maps are similar but some are more aggressive such as Tesla. To some extent, this is related to the different approaches they took in the design of the E/E architecture. However, the trend of software and hardware decoupling is consensus, and the role of software will become more and more important.

D. SIMULATION PLATFORMS

Before the mass production of automobiles equipped with a new E/E architecture, the E/E architecture needs to be simulated for analysis and verification. Modification and iteratively updates are essential. Many studies have used simulation tools to model and analyze the performance of E/E architectures and in-vehicle networks. The main simulation analysis platforms contain the E/E architecture design platform Preevision [119], automotive bus development and analysis tools CANoe [120]. In addition, timing requirements are the basis of automotive real-time embedded systems. Other timing analysis tools are also used to analyze the timing of automotive embedded systems, such as OTAWA, TuBound, Chronos, RapiTime, Cheddar, MAST, SymTA/S and RapidRMA. These softwares are described in detail in [121]. Besides, discrete event simulator OMNeT ++, NS3, etc. are commonly used in network simulation.

Preevision focuses on E/E architecture design, while Symtavision and OMNeT ++ focus on simulation and verification. Preevision is a mature model-based tool for developing E/E systems including requirements design, software, communication design and harness development. Many researches have carried out development work from architecture design to wiring layout based on Preevision [122].

Many researches on in-vehicle networks are based on OmNeT++ including CAN, FlexRay and especially Ethernet. In OMNeT++, some mature frameworks is available such as FiCo4OMNeT for CAN and FlexRay [123] and open-source model library INET. Lots of researches on TSN are being done on OMNeT++. For example, Steinbach *et al.* [124] developed CORE4INET framework which is still updating. Häckel *et al.* [125] modeled a SDN-based networking architecture.

In addition, many researchers have developed their own simulation models based on the OMNeT ++ platform with

different model establishment and protocol implementation. Alderisi *et al.* [90] Used OMNeT ++ to evaluate the performance of AVB and TTE two technologies in automobiles ADAS and multimedia systems; Heise *et al.* [126] implements all non-time-based features in TSN based on the OMNeT ++ simulation framework for avionics networks including flow authentication, flow filtering and flow supervision, frame preemption. And an open source simulation framework was provided. Steinbach *et al.* [127] compared the simulation prototype of the Volkswagen Golf 7 with a complete heterogeneous in-vehicle network design. And the paper compared the results of traditional central gateway based network with real-time Ethernet backbone network. The results show that real-time Ethernet solutions can provide better end-to-end delay and jitter, while providing significant bandwidth reserves.

Jiang *et al* [128] developed a TSN simulation model based on OMNET++ and implemented 802.1Qbv, 802.1AS on the basis of an open source framework. Jiang introduced how to perform network scheduling calculations and how to derive gate control list, which was verified through simulation. Similarly, Falk *et al.* [129] proposed a simulation model NeSTiNg (Network Simulator for Time-Sensitive Networking) to analyze the performance of a converged IEEE 802.1 TSN network. It is a TSN specific extension to the popular OMNeT ++ / INET framework for networks simulation, especially the time-triggered scheduling mechanism of IEEE Std 802.1Qbv, also includes frame preemption of IEEE 802.1Qbu and IEEE 802.3br.

V. PROSPECTIVE TRENDS FOR E/E ARCHITECTURE
The automobile in the future could be described as ''computers-on-wheels'' and the innovation of E/E architecture is inevitable. The E/E architecture will gradually evolve into a ''communication-control-computation'' platform. Several key trends will lead the innovation of automotive E/E architecture, and technological breakthroughs make transformations possible. Some future possibilities and potentials will be discussed. Fig.3 shows the aspects including overall architecture, in-vehicle communication, power supply, and software platform, which will be continually improved in the coming decades. With the development of automated driving, more and more electron devices, such as ECUs, sensors, and actuators will be equipped on the automobiles. It is predicted that there will be more than 4,000 electronic nodes in a L5 level automated driving automobile. Therefore, strong computing and communication capability are essential. In addition, to simply wiring harness system, power and data should be transferred mixed.

A. ARCHITECTURE
At present and several years in the future, plenty of automotive E/E architectures will still be domain-based architectures. It is partly dependent on the current level of intelligent driving and limited by the existing E/E manufacturing platforms. At the same time, some aggressive companies chose the

zone-based architecture, which is likely to be the direction of automotive industry in the next ten years. During the period, the increase in computing power will also make it possible for the central computing platform. Automobiles are going to be more like a moving server. The maturity of cloud will also promote the virtualization of the architecture. Considering the huge amount of data and the interaction delay, proper edge comuputing may be considered in the future. The architecture that combines centralization and distributed computing improves the flexibility of ECU virtualization owing to its edge processing capabilities. At the same time, the architecture does not rely solely on the central processing, which improve the security of automobiles.

B. IN-VEHICLE COMMUNICATION
With the development of intelligent driving, more sensors will be equipped on the automobile in order to satisfy the functional requirements of automated driving, such as camera, LiDAR, ultrasonic sensor, IMU(inertial measurement unit) and so on. Data fusion makes the perception of the environment more comprehensive and reliable. However, it also means greater bandwidth requirements, higher reliability and deterministic low latency, which is the basic guarantee for upper-layer computing. Taking the camera for an example, many cameras are used for object detection, recognition and classification, such as surround-view cameras, front-view cameras, and in-car cameras. For example, a 1080p 60fps camera probably requires 3Gbps bandwidth theoretically. And a 4K 60fps HD camera probably requires more than 10Gbps. Ethernet-based in-vehicle network is an inevitable trend in the future.

And considering about the safety criticality of calculation and decision-making and the uncertainty caused by insufficient computing power, low deterministic delay in communication is very important. Therefore, under the trend of more complex in-vehicle traffic, how to schedule and control traffic so that the delay and jitter of critical and non-critical traffic are acceptable will gradually become the focus. At the same time, frequent changes in network topology or traffic patterns make the static configuration methods no longer adapted. And automobiles probably need to dynamically change the in-vehicle network in the future. New communication sensors, new ECUs or software applications may be added to vehicles [130]. In this case, software-defined networking (SDN) approaches have been proved to be effective [130]–[135]. It can ensure the quality of service even in changing topologies by implementing monitoring, analysis, planning and execution. Applying the SDN to in-vehicle network is becoming a promising research.

C. POWER SUPPLY
The wiring harness of in-vehicle network should be paid more and more attention in the future to reduce the wiring length and weight, which is beneficial to the automotive lightweight, fuel economy and maintainable. As mentioned above, PoE is a promising solution. Simultaneously, the trend of automobile
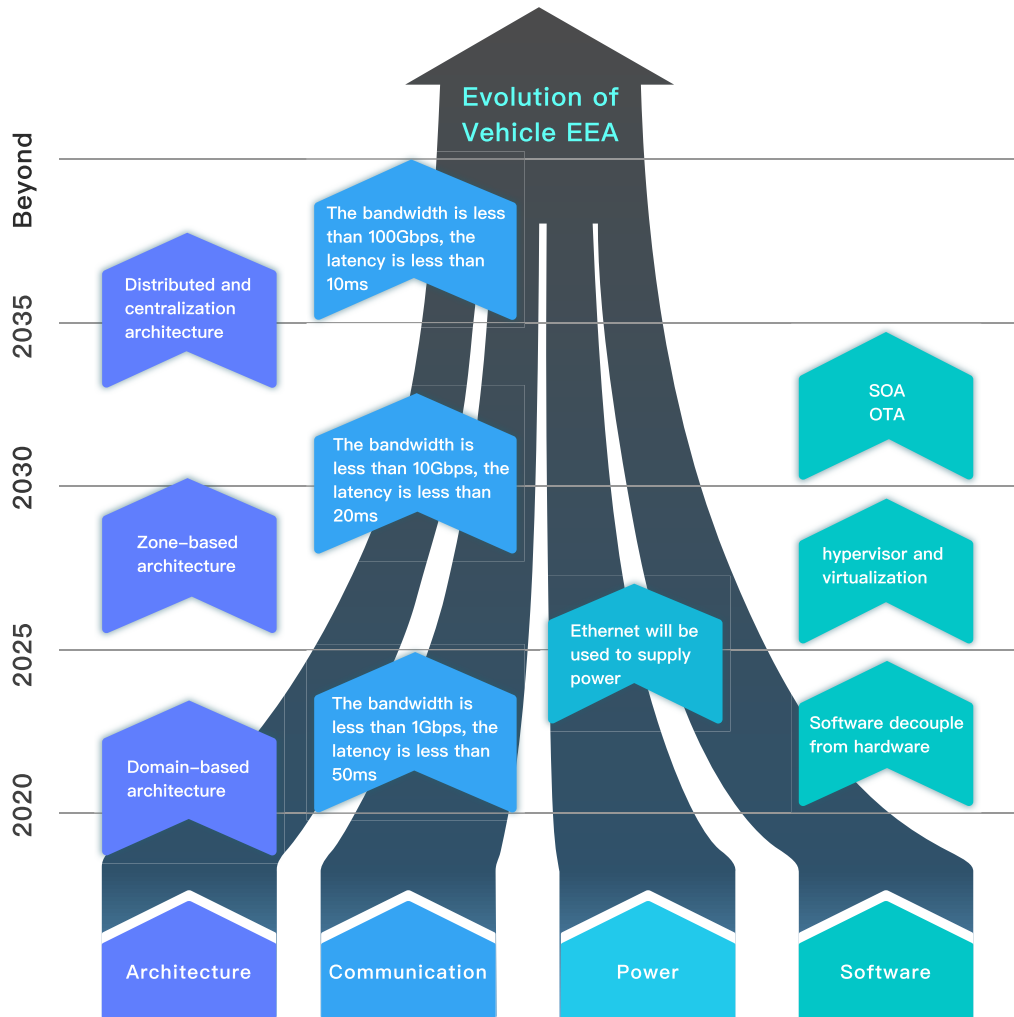
FIGURE 3. Future trends E/E architecture.

electrification has put forward higher requirements on the power network, especially the electromagnetic interference of high-voltage systems. The use of measures such as shielding and filtering, as well as reasonable arrangements on the wiring, require special attention.

D. SOFTWARE PLATFORM

Software in automotive systems has been increasing crucial. Automotive systems will be more software-intensive in the future. Therefore software engineering is gradually becoming an integral part of the automotive E/E architecture [136].

Originally, the update of automobile software is implemented at the vehicle service stations, connecting to the Internet via the On Board Diagnostic (OBD) interface. Software updates over the air (SOTA) releases the update time or space constraints. For the software service package released by the OEM back-end, the ECU is updated through the WiFi, cellular network or satellite communications. SOTA can fix system flaws, reduce warranty and recall costs and enhance the

vehicle's service capabilities [2] At the same time, the safety and availability of the vehicle should be ensured as much as possible in the OTA update process to avoid vehicle downtime or even update failure [137]. It is also important to have sufficient available power when updating to avoid update interruption [138].

Service-Oriented Architecture (SOA) is an IT architecture that supports service orientation and allows integration of distributed and heterogeneous components based on standardized interfaces [139]. SOA emphasizes the separation of services from applications. The separation of software and hardware is an essential element of SOA [140]. Recently, SOA has become the key to the integration and interoperability of different applications and systems in an organization. In SOA, applications are split into small units that can share services between different applications. As a result, SOA promotes integration and demonstrate the high degree of adaptability. SOA enables the easy adaptation to changes and new requirements, rapid design and integration of complex functions [141]. SOA has proven to be an efficient and

flexible software design method, mainly due to its scalability and reusability.

SOA has a huge impact on the IT industry and it is already a proven architecture model [142], [143]. It has also been successfully carried out in other industries such as industrial, financial and transportation fields [141]. An end-to-end approach to integrate vehicle EE architectures for such heterogeneous applications is SOA. The more flexible the functions required while running, the more communication bandwidth and computing power are required [143]. Therefore, for automotive embedded systems, the solution of Ethernet technology, provides a basis for the development of SOA. Gopu [142] discussed the feasibility study on whether SOA can be implemented on automotive embedded systems. It showed that SOA can be implemented by modifying existing protocols or creating standards. For example, the AUTOSAR architecture can use SOME / IP and integrate SOA into the automotive embedded system without affecting existing ones. With the development of automotive Ethernet and the increase use in Internet services in vehicles, SOA will be more suitable and applied to vehicles [143]. BMW has proposed an SOA method for the next generation of EE architecture, which reduces system complexity through the abstract services, strict encapsulation and hierarchy provided by SOA [144].

VI. CONCLUSION

The E/E architecture needs continuous transformation and innovation to adapt to new automotive technologies. This paper summarizes the design and evaluation of the E/E architecture, and the development of the last century in detail. In addition, we elaborate the state-of-the-art E/E architecture from the perspective of automotive industry, organizational standards, and academic research. Finally? we propose the prospective trends based on current bottlenecks.

REFERENCES

[1] K. Michels. *Vehicle E/E-Architecture: Reduce to the Max.* Accessed: Nov. 27, 2019. [Online]. Available: https://www.eenewsautomotive.com/news/vehicle-ee-architecture-reduce-max

[2] S. Jiang, "Vehicle E/E architecture and its adaptation to new technical trends," SAE Tech. Paper 2019-01-0862, 2019.

[3] V. M. Navale, K. Williams, A. Lagospiris, M. Schaffert, and M.-A. Schweiker, "(R) evolution of E/E architectures," *SAE Int. J. Passenger Cars-Electron. Elect. Syst.*, vol. 8, pp. 282–288, Apr. 2015.

[4] W. Zeng, M. A. S. Khalid, and S. Chowdhury, "In-vehicle networks outlook: Achievements and challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 1552–1571, 3rd Quart., 2016.

[5] J. Huang, M. Zhao, Y. Zhou, and C.-C. Xing, "In-vehicle networking: Protocols, challenges, and solutions," *IEEE Netw.*, vol. 33, no. 1, pp. 92–98, Jan. 2019.

[6] R. Schwabel, "Technical challenges in future electrical architectures," SAE Tech. Paper 2011-01-1021, 2011.

[7] W. Stolz, K. Williams, T. Lorenz, and M. Piastowski, "Ethernet and IP—The solution to master complexity, safety and security in vehicle communication networks?" SAE Tech. Paper 2011-01-1042, 2011.

[8] R. Boatright and J. Tardo, "Security aspects of utilizing Ethernet AVB as the converged vehicle backbone," *SAE Int. J. Passenger Cars-Electron. Elect. Syst.*, vol. 5, pp. 470–478, Apr. 2012.

[9] A. Saad, W. Bauer, M. Haneberg, and J. Schiffers, "Intelligent automotive system services—An emerging design pattern for an advanced E/E-architecture," SAE Tech. Paper 2006-01-1286, 2006.

[10] W. Stolz, R. Kornhaas, R. Krause, and T. Sommer, "Domain control units-the solution for future E/E architectures?" SAE Tech. Paper 2010-01-0686, Tech. Rep., 2010.

[11] L. Guo, A. Ghosal, H. Zeng, P. Giusto, and A. Sangiovanni-Vincentelli, "Methods and tools for calculating the flexibility of automotive HW/SW architectures," *SAE Int. J. Passenger Cars Electron. Electr. Syst.*, vol. 5, no. 1, pp. 17–26, Apr. 2012.

[12] G. D. Poggetto and L. A. Moscato, "Criteria for selection of vehicle electrical architectures for emerging markets," SAE Tech. Paper 2010-36-0069, 2010.

[13] A. Ghosal, P. Giusto, P. Peranandam, P. Sinha, and H. Zeng, "Metrics for quantifying and evaluating ability of electronic control system architectures to accommodate changes," SAE Tech. Paper 2011-01-0447, 2011.

[14] C. P. Quigley, "Process for the selection of low cost and weight electrical architectures," SAE Tech. Paper 2011-01-1436, 2011.

[15] E. C. De Oliveira, "Electrical architectures and in-vehicles networks," SAE Tech. Paper 2007-01-1713, 2007.

[16] R. Blank, "EE-architecture the real competence field for automotive OEMs," SAE Tech. Paper 2015-01-0145, 2015.

[17] F. de Brito Santos and C. A. F. Sartori, "A contribution to new technologies and architectures evaluation for automotive electrical systems," SAE Tech. Paper 2004-01-3419, 2004.

[18] O. Avatefipour and H. Malik, "State-of-the-art survey on in-vehicle network communication (CAN-Bus) security and vulnerabilities," 2018, *arXiv:1802.01725*. [Online]. Available: http://arxiv.org/abs/1802.01725

[19] T. Nolte, "Share-driven scheduling of embedded networks," Ph.D. dissertation, Institutionen för Datavetenskap och Elektronik, Dublin, OH, USA, 2006.

[20] L. Jagannathan and J. Sankaran, "Development of E/E architecture for commercial vehicles in emerging Indian markets," SAE Tech. Paper 2013-01-2816, 2013.

[21] S. Tuohy, M. Glavin, C. Hughes, E. Jones, M. Trivedi, and L. Kilmartin, "Intra-vehicle networks: A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 534–545, Apr. 2015.

[22] U. Keskin, "In-vehicle communication networks: A literature survey," Comput. Sci. Rep., 2009, vol. 10.

[23] J. Bigorra, C. Borrego, J. Fontanilles, and J. Giró, "Innovative electrical and electronic architecture for vehicles with dual voltage power networks. In-vehicle application," SAE Tech. Paper 2000-01-0452, 2000.

[24] R. Hadeler and H.-J. Mathony, "Design of intelligent body networks," SAE Tech. Paper 2000-01-0152, 2000.

[25] T. Nguyen and J. Bell, "The benefit of co-developing vehicle electrical & electronic architecture between OEM and supplier," SAE Tech. Paper 2015-01-0475, 2015.

[26] Luc van Dijk. (2017). *NXP Semiconductors*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.nxp.com/docs/en/white-paper/FVNECUA4WP.pdf

[27] S. Brunner, J. Roder, M. Kucera, and T. Waas, "Automotive E/E-architecture enhancements by usage of Ethernet TSN," in *Proc. 13th Workshop Intell. Solutions Embedded Syst. (WISES)*, Jun. 2017, pp. 9–13.

[28] (2019). *Research and Markets*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.researchandmarkets.com/research/7vn7hb/global_automotive?w=5

[29] D. Reinhardt and M. Kucera, "Domain controlled architecture-a new approach for large scale software integrated automotive systems," *PECCS*, vol. 13, pp. 221–226, 2013.

[30] D. Diekhoff, "AUTOSAR basic software for complex control units," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2010, pp. 263–266.

[31] N. Navet, Y. Song, F. Simonot-Lion, and C. Wilwert, "Trends in automotive communication systems," *Proc. IEEE*, vol. 93, no. 6, pp. 1204–1223, Jun. 2005.

[32] M. Mody, J. Jones, K. Chitnis, R. Sagar, G. Shurtz, Y. Dutt, M. Koul, M. Biju, and A. and Dubey, "Understanding vehicle E/E architecture topologies for automated driving: System partitioning and tradeoff parameters," *Electron. Imag.*, vol. 2018, no. 17, pp. 1–358, 2018.

[33] *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications—Data Terminal Equipment (DTE) Power via Media Dependent Interface (MDI)*, IEEE Standard 802.3af-2003 (Amendment to IEEE 802.3-2002, including IEEE Std 802.3ae-2002), Jun. 2003, pp. 1–133.

[34] *IEEE Standard for Information Technology–Local and Metropolitan Area Networks–Specific Requirements—Part 3: CSMA/CD Access Method and Physical Layer Specifications Amendment 3: Data Terminal Equipment (DTE) Power via the Media Dependent Interface (MDI) Enhancements*, IEEE Standard 802.3at-2009 (Amendment to IEEE Std 802.3-2008), Oct. 2009, pp. 1–137.

[35] *IEEE Standard for Ethernet–Amendment 8: Physical Layer and Management Parameters for Power Over Data Lines (PODL) of Single Balanced Twisted-Pair Ethernet*, IEEE Standard 802.3bu-2016 (Amendment to IEEE Std 802.3-2015, IEEE 802.3bw-2015, IEEE 802.3by-2016, IEEE 802.3bq-2016, IEEE 802.3bp-2016, IEEE 802.3br-2016, IEEE 802.3bn-2016, IEEE 802.3bz-2016), Feb. 2017, pp. 1–77.

[36] J. Bell, "Network protocols used in the automotive industry," Univ. Wales, Aberystwyth, FL, USA, 2002, pp. 7–24.

[37] H. Kopetz and G. Grunsteidl, "TTP—A time-triggered protocol for fault-tolerant real-time systems," in *Proc. 23rd Int. Symp. Fault-Tolerant Comput. (FTCS)*, 1993, pp. 524–533.

[38] C. T. Triggered, "Thomas Fuehrer, Bernd Mueller, Florian Hartwich and Robert Hugel," SAE 2001-01-0073.

[39] W. Steiner, "TTEthernet specification," TTTech Computertechnik AG, Wien, Austria, Tech. Rep., Nov. 2008, p. 40, vol. 39.

[40] (2014). *Ixia*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.ixiacom.com/

[41] S. Singer, "Ethernet in infotainment-challenges and solutions," SAE Tech. Paper 2014-01-0250, 2014.

[42] (2019). *OPEN Alliance SIG*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.opensig.org/

[43] (2019). *IEEE 802.3 ETHERNET Working Group*. Accessed: Jan. 9, 2020. [Online]. Available: http://www.ieee802.org/3/B10GAUTO/index.html

[44] S. Kehrer, O. Kleineberg, and D. Heffernan, "A comparison of fault-tolerance concepts for IEEE 802.1 time sensitive networks (TSN)," in *Proc. IEEE Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2014, pp. 1–8.

[45] A. Nasrallah, A. S. Thyagaturu, Z. Alharbi, C. Wang, X. Shao, M. Reisslein, and H. ElBakoury, "Ultra-low latency (ULL) networks: The IEEE TSN and IETF DetNet standards and related 5G ULL research," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 88–145, 1st Quart., 2019.

[46] (2019). *AVnuAlliance*. Accessed: Jan. 9, 2020. [Online]. Available: https://avnu.org/

[47] H. Kopetz, A. Ademaj, P. Grillinger, and K. Steinhammer, "The time-triggered Ethernet (TTE) design," in *Proc. 8th IEEE Int. Symp. Object-Oriented Real-Time Distrib. Comput. (ISORC)*, May 2005, pp. 22–33.

[48] T. Steinbach, P. Meyer, S. Buschmann, and F. Korf, "Extending OMNeT++ towards a platform for the design of future in-vehicle network architectures," 2016, *arXiv:1609.05179*. [Online]. Available: http://arxiv.org/abs/1609.05179

[49] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems—Redline*, IEEE Standard 1588-2008 (Revision IEEE Std 1588-2002)-Redline, Jul. 2008, pp. 1–300.

[50] (2016). *SAE*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.sae.org/standards/content/as6802/

[51] T. Steinbach, F. Korf, and T. C. Schmidt, "Comparing time-triggered Ethernet with FlexRay: An evaluation of competing approaches to real-time for in-vehicle networks," in *Proc. IEEE Int. Workshop Factory Commun. Syst.*, May 2010, pp. 199–202.

[52] L. Wisniewski, M. Schumacher, J. Jasperneite, and C. Diedrich, "Increasing flexibility of time triggered Ethernet based systems by optimal greedy scheduling approach," in *Proc. IEEE 20th Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2015, pp. 1–6.

[53] W. Steiner, "An evaluation of SMT-based schedule synthesis for time-triggered multi-hop networks," in *Proc. 31st IEEE Real-Time Syst. Symp.*, Nov. 2010, pp. 375–384.

[54] Z. Zheng, F. He, and Y. Xiong, "The research of scheduling algorithm for time-triggered Ethernet based on path-hop," in *Proc. IEEE/AIAA 35th Digit. Avionics Syst. Conf. (DASC)*, Sep. 2016, pp. 1–6.

[55] W. Liu, Q. Li, F. He, and H. Xiong, "Research on time-triggerd-Ethernet synchronization and scheduling mechanism," *Aeronaut. Comput. Technique*, vol. 41, no. 4, pp. 122–127, 2011.

[56] W. Steiner, "Synthesis of static communication schedules for mixed-criticality systems," in *Proc. 14th IEEE Int. Symp. Object/Compon./Service-Oriented Real-Time Distrib. Comput. Workshops*, Mar. 2011, pp. 11–18.

[57] *IEEE Standard for Local and Metropolitan Area Networks—Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks*, IEEE Standard 802.1AS-2011, Mar. 2011, pp. 1–292.

[58] *IEEE Standard for Local and Metropolitan Area Networks—Virtual Bridged Local Area Networks Amendment 12: Forwarding and Queuing Enhancements for Time-Sensitive Streams*, IEEE Standard 802.1Qav-2009 (Amendment to IEEE Std 802.1Q-2005), Jan. 2010, pp. C1–72.

[59] *IEEE Standard for Local and Metropolitan Area Networks–Virtual Bridged Local Area Networks Amendment 14: Stream Reservation Protocol (SRP)*, IEEE Standard 802.1Qat-2010 (Revision IEEE Std 802.1Q-2005), Sep. 2010, pp. 1–119.

[60] *IEEE Standard for Local and Metropolitan Area Networks–Audio Video Bridging (AVB) Systems*, IEEE Standard 802.1BA-2011, Sep. 2011, pp. 1–45.

[61] *IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks–Amendment 30: Yang Data Model*, IEEE Standard 802.1Qcp-2018 (Amendment to IEEE Std 802.1Q-2018), Sep. 2018, pp. 1–93.

[62] *IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks—Amendment 31: Stream Reservation Protocol (SRP) Enhancements and Performance Improvements*, IEEE Standard 802.1Qcc-2018 (Amendment to IEEE 802.1Q-2018 as amended by IEEE 802.1Qcp-2018), Oct. 2018, pp. 1–208.

[63] *IEEE Draft Standard for Local and Metropolitan Area Networks–Link-Local Registration Protocol*, Standard IEEE P802.1CS/D1.5, Jun. 2018, pp. 1–124.

[64] *IEEE Standard for Local and Metropolitan Area Networks—Bridges and Bridged Networks—Amendment 25: Enhancements for Scheduled Traffic*, IEEE Standard 802.1Qbv-2015 (Amendment to IEEE 802.1Q-2014 as amended by IEEE 802.1Qca-2015, IEEE 802.1Qcd-2015, IEEE 802.1Q-2014/Cor 1-2015), Mar. 2016, pp. 1–57.

[65] *IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks—Amendment 26: Frame Preemption*, IEEE Standard 802.1Qbu-2016 (Amendment to IEEE Std 802.1Q-2014), Aug. 2016, pp. 1–52.

[66] *IEEE Standard for Ethernet Amendment 5: Specification and Management Parameters for Interspersing Express Traffic*, IEEE Standard 802.3br-2016 (Amendment to IEEE 802.3-2015 as amended by IEEE St802.3bw-2015, IEEE 802.3by-2016, IEEE 802.3bq-2016, IEEE 802.3bp-2016), Oct. 2016, pp. 1–58.

[67] *IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks–Amendment 29: Cyclic Queuing and Forwarding*, IEEE Standard 802.1Qch-2017 (Amendment to IEEE 802.1Q-2014 as amended by IEEE 802.1Qca-2015, IEEE 802.1Qcd(TM)-2015, IEEE 802.1Q-2014/Cor 1-2015, IEEE 802.1Qbv-2015, IEEE 802.1Qbu-2016, IEEE 802.1Qbz-2016, IEEE 802.1Qci-2017), Jun. 2017, pp. 1–30.

[68] (2019). *IEEE TSN Task Group*. Accessed: Jan. 9, 2020. [Online]. Available: https://1.ieee802.org/tsn/802-1qcr/

[69] *IEEE Standard for Local and Metropolitan Area Networks–Frame Replication and Elimination for Reliability*, IEEE Standard 802.1CB-2017, Oct. 2017, pp. 1–102.

[70] *IEEE Standard for Local and Metropolitan Area Networks—Bridges and Bridged Networks—Amendment 24: Path Control and Reservation*, IEEE Standard 802.1Qca-2015 (Amendment to IEEE 802.1Q-2014 as amended by IEEE 802.1Qcd-2015 IEEE 802.1Q-2014/Cor 1-2015), Mar. 2016, pp. 1–120.

[71] *IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks–Amendment 28: Per-Stream Filtering and Policing*, IEEE Standard 802.1Qci-2017 (Amendment to IEEE 802.1Q-2014 as amended by IEEE 802.1Qca-2015, IEEE 802.1Qcd-2015, IEEE 802.1Q-2014/Cor 1-2015, IEEE 802.1Qbv-2015, IEEE 802.1Qbu-2016, IEEE 802.1Qbz-2016), Sep. 2017, pp. 1–65.

[72] J. Rox and R. Ernst, "Formal timing analysis of full duplex switched based Ethernet network architectures," SAE Tech. Paper 2010-01-0455, 2010.

[73] F. Reimann, S. Graf, F. Streit, M. Glaß, and J. Teich, "Timing analysis of Ethernet AVB-based automotive E/E architectures," in *Proc. IEEE 18th Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2013, pp. 1–8.

[74] J. Diemer, D. Thiele, and R. Ernst, "Formal worst-case timing analysis of Ethernet topologies with strict-priority and AVB switching," in *Proc. 7th IEEE Int. Symp. Ind. Embedded Syst. (SIES)*, Jun. 2012, pp. 1–10.

[75] L. Zhao, H. Xiong, Z. Zheng, and Q. Li, "Improving worst-case latency analysis for rate-constrained traffic in the time-triggered Ethernet network," *IEEE Commun. Lett.*, vol. 18, no. 11, pp. 1927–1930, Nov. 2014.

[76] M. Boyer, H. Daigmorte, N. Navet, and J. Migge, "Performance impact of the interactions between time-triggered and rate-constrained transmissions in TTEthernet," Tech. Rep., 2016.

[77] L. Zhao, P. Pop, Z. Zheng, and Q. Li, "Timing analysis of AVB traffic in TSN networks using network calculus," in *Proc. IEEE Real-Time Embedded Technol. Appl. Symp. (RTAS)*, Apr. 2018, pp. 25–36.

[78] C. Boiger, *How Many Transmission Selection Algorithms Do We Need?* IEEE Standard 802.1 Interim Meeting, 2013.

[79] P. Meyer, T. Steinbach, F. Korf, and T. C. Schmidt, "Extending IEEE 802.1 AVB with time-triggered scheduling: A simulation study of the coexistence of synchronous and asynchronous traffic," in *Proc. IEEE Veh. Netw. Conf.*, Dec. 2013, pp. 47–54.

[80] S. Thangamuthu, N. Concer, P. J. L. Cuijpers, and J. J. Lukkien, "Analysis of Ethernet-switch traffic shapers for in-vehicle networking applications," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, 2015, pp. 55–60.

[81] D. Thiele, R. Ernst, and J. Diemer, "Formal worst-case timing analysis of Ethernet TSN's time-aware and peristaltic shapers," in *Proc. IEEE Veh. Netw. Conf. (VNC)*, Dec. 2015, pp. 251–258.

[82] D. Thiele and R. Ernst, "Formal worst-case timing analysis of Ethernet TSN's burst-limiting shaper," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, 2016, pp. 187–192.

[83] J. Specht and S. Samii, "Urgency-based scheduler for time-sensitive switched Ethernet networks," in *Proc. 28th Euromicro Conf. Real-Time Syst. (ECRTS)*, Jul. 2016, pp. 75–85.

[84] A. Grigorjew, F. Metzger, T. Hoßfeld, and J. Specht, "A simulation of asynchronous traffic shapers in switched Ethernet networks," in *Proc. Int. Conf. Netw. Syst. (NetSys)*, Mar. 2019, pp. 1–6.

[85] Z. Zhou, Y. Yan, M. Berger, and S. Ruepp, "Analysis and modeling of asynchronous traffic shaping in time sensitive networks," in *Proc. 14th IEEE Int. Workshop Factory Commun. Syst. (WFCS)*, Jun. 2018, pp. 1–4.

[86] F. He, L. Zhao, and E. Li, "Impact analysis of flow shaping in Ethernet-AVB/TSN and AFDX from network calculus and simulation perspective," *Sensors*, vol. 17, no. 5, p. 1181, May 2017.

[87] W.-K. Jia, G.-H. Liu, and Y.-C. Chen, "Performance evaluation of IEEE 802.1Qbu: Experimental and simulation results," in *Proc. 38th Annu. IEEE Conf. Local Comput. Netw.*, Oct. 2013, pp. 659–662.

[88] J. Kim, B. Y. Lee, and J. Park, "Preemptive switched Ethernet for real-time process control system," in *Proc. 11th IEEE Int. Conf. Ind. Informat. (INDIN)*, Jul. 2013, pp. 171–176.

[89] D. Thiele and R. Ernst, "Formal worst-case performance analysis of time-sensitive Ethernet with frame preemption," in *Proc. IEEE 21st Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2016, pp. 1–9.

[90] G. Alderisi, A. Caltabiano, G. Vasta, G. Iannizzotto, T. Steinbach, and L. L. Bello, "Simulative assessments of IEEE 802.1 Ethernet AVB and time-triggered Ethernet for advanced driver assistance systems and in-car infotainment," in *Proc. IEEE Veh. Netw. Conf. (VNC)*, Nov. 2012, pp. 187–194.

[91] L. Zhao, F. He, E. Li, and J. Lu, "Comparison of time sensitive networking (TSN) and TTEthernet," in *Proc. IEEE/AIAA 37th Digit. Avionics Syst. Conf. (DASC)*, Sep. 2018, pp. 1–7.

[92] D. Kum, G.-M. Park, S. Lee, and W. Jung, "AUTOSAR migration from existing automotive software," in *Proc. Int. Conf. Control, Autom. Syst.*, Oct. 2008, pp. 558–562.

[93] J. Becker, O. Sander, M. Huebner, M. Traub, T. Weber, J. Luka, and V. Lauer, "Standards for electric/electronic components and architectures," SAE Tech. Paper 2008-21-0022, 2008.

[94] S. Bunzel, "AUTOSAR—The standardized software architecture," *Informatik-Spektrum*, vol. 34, no. 1, pp. 79–83, Feb. 2011.

[95] R. Hegde, M. Hegde, and K. Gurumurthy, "A paradigm shift from legacy to AUTOSAR architecture in future automotives," in *Proc. Int. Conf. Comput. Intell. Inf. Technol.* Berlin, Germany: Springer, 2011, pp. 548–553.

[96] D. D. Turner, "Determining the optimal distributed electronic module solution of an automotive system while incorporating harness routing alternatives in an electrical/electronic architecture tool environment," SAE Tech. Paper 2008-01-0283, 2008.

[97] B. Nagabhushana, "Security and protection aspects of automotive architectures for external influences," SAE Tech. Paper 2008-21-0021, 2008.

[98] Y. Lee, J. H. Kim, and J. W. Jeon, "Applying AUTOSAR network management in OSEK/VDX for compatibility of AUTOSAR and OSEK/VDX," in *Proc. FISITA World Automot. Congr.* Berlin, Germany: Springer, 2013, pp. 693–704.

[99] K. Jo and M. Sunwoo, "Development of localisation and mapping software for autonomous cars," *IET Intell. Transp. Syst.*, vol. 13, no. 2, pp. 406–416, Feb. 2019.

[100] J.-W. Kim, K.-J. Lee, and H.-S. Ahn, "Development of software component architecture for motor-driven power steering control system using AUTOSAR methodology," in *Proc. 15th Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2015, pp. 1995–1998.

[101] S. Mirheidari, A. Fallahi, D. Zhang, and K. Kuppam, "AUTOSAR model-based software component integration of supplier software," *SAE Int. J. Commercial Vehicles*, vol. 8, no. 2, pp. 544–548, Sep. 2015.

[102] S. Isac, N. P. Chaudhari, and K. C. Ananthoju, "An AUTOSAR based approach to time synchronize in-vehicle ECUs in mixed network architecture," SAE Tech. Paper 2019-26-0010, 2019.

[103] M. Graniou, H. Sivencrona, and R. Svenningsson, "Advantages and challenges of introducing AUTOSAR for safety-related systems," SAE Tech. Paper 2009-01-0750, 2009.

[104] S. Menon and P. Venugopal, "AUTOSAR software platform adoption: Systems engineering strategies," SAE Tech. Paper 2014-01-0289, 2014.

[105] J. Price, "Design automation and its challenges in AUTOSAR-based vehicle E/E systems," SAE Tech. Paper 2015-01-0178, 2015.

[106] S. Furst and M. Bechter, "AUTOSAR for connected and autonomous vehicles: The AUTOSAR adaptive platform," in *Proc. 46th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. Workshop (DSN-W)*, Jun. 2016, pp. 215–217.

[107] *AUTOSAR Adaptive Platform*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.autosar.org/standards/adaptive-platform/

[108] (2019). *Fred Lambert. Electrek*. Accessed: Jan. 9, 2020. [Online]. Available: https://ww.electrek.co/2019/07/22/tesla-revolutionary-wiring-architecture-robots-model-y/

[109] (2019). *Rich Pell*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.eenewseurope.com/news/new-gm-digital-vehicle-platform-enable-future-technologies

[110] (2017). *Paul Hansen*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.electronicdesign.com/markets/automotive/article/21804988/bmw-and-audi-want-to-separate-vehicle-hardware-from-software

[111] (2018). *Kenji Nishikawa and Kenji Hontani.TOYOTA Motor Cop.* Accessed: Jan. 9, 2020. [Online]. Available: https://assets.vector.com/cms/content/events/2018/VeCo18/presentations/VeCo18_Nishikawa.pdf

[112] (2017). *Yasuaki MORITA.Toyota Motor Europe NV/SA.* Accessed: Jan. 9, 2020. [Online]. Available: https://assets.vector.com/cms/content/events/2017/vSES17/presentations/vSES17_02_Morita.pdf

[113] (2017). *Continental*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.continental-automotive.com/en-gl/Passenger-Cars/Autonomous-Mobility/Enablers/Control-Units/Assisted-Automated-Driving-Control-Unit

[114] (2017). *NXP*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.nxp.com/docs/en/brochure/S32-Automotive-Processing-Platform.pdf

[115] (2019). *APTIV*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.aptiv.com/smart-vehicle-architecture

[116] (2018). *APTIV*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.aptiv.com/newsroom/article/smart-vehicle-architecture-the-backbone-of-the-future

[117] H. Zinner. (2019). *Julian Brand, Daniel Hopf, Continental AG. 802.1 TSN*. Accessed: Jan. 9, 2020. [Online]. Available: http://ieee802.org/1/files/public/docs2019/dg-zinner-automotive-architecture-evolution-0319-v02.pdf

[118] (2020). *NVIDIA*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.nvidia.com/en-us/self-driving-cars/drive-platform/

[119] *Preevision. Vector*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.vector.com/int/en/products/products-a-z/software/preevision/

[120] *CANoe. Vector*. Accessed: Jan. 9, 2020. [Online]. Available: https://www.vector.com/int/en/products/products-a-z/software/canoe/

[121] M. M. D. Santos, V. Ambiel, M. Acras, and P. Gliwa, "On the timing analysis at automotive real-time embedded systems," SAE Tech. Paper 2017-01-1618, 2017.

[122] J. Schäuffele, "E/E architectural design and optimization using preevision," SAE Tech. Paper 2016-01-0016, 2016.

[123] S. Buschmann, T. Steinbach, F. Korf, and T. Schmidt, "Simulation based timing analysis of FlexRay communication at system level," in *Proc. 6th Int. Conf. Simulation Tools Techn.*, 2013, pp. 285–290.

[124] T. Steinbach, H. D. Kenfack, F. Korf, and T. Schmidt, "An extension of the OMNeT++ INET framework for simulating real-time Ethernet with high accuracy," in *Proc. 4th Int. Conf. Simulation Tools Techn. (ICST)*, 2011, pp. 375–382.

[125] T. Häckel, P. Meyer, F. Korf, and T. C. Schmidt, "SDN4CoRE: A simulation model for software-defined networking for communication over real-time Ethernet," 2019, *arXiv:1908.09649*. [Online]. Available: http://arxiv.org/abs/1908.09649

[126] P. Heise, F. Geyer, and R. Obermaisser, "TSimNet: An industrial time sensitive networking simulation framework based on OMNeT++," in *Proc. 8th IFIP Int. Conf. New Technol., Mobility Secur. (NTMS)*, Nov. 2016, pp. 1–5.

[127] T. Steinbach, P. Meyer, S. Buschmann, and F. Korf, "Extending OMNeT++ towards a platform for the design of future in-vehicle network architectures," 2016, *arXiv:1609.05179*. [Online]. Available: http://arxiv.org/abs/1609.05179

[128] J. Jiang, Y. Li, S. H. Hong, A. Xu, and K. Wang, "A time-sensitive networking (TSN) simulation model based on OMNeT++," in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Aug. 2018, pp. 643–648.

[129] J. Falk, D. Hellmanns, B. Carabelli, N. Nayak, F. Dürr, S. Kehrer, and K. Rothermel, "NeSTiNg: Simulating IEEE time-sensitive networking (TSN) in OMNeT++," in *Proc. Int. Conf. Netw. Syst. (NetSys)*, Mar. 2019, pp. 1–8.

[130] J. L. Du and M. Herlich, "Software-defined networking for real-time Ethernet," in *Proc. ICINCO*, 2016, pp. 584–589.

[131] N. G. Nayak, F. Dürr, and K. Rothermel, "Software-defined environment for reconfigurable manufacturing systems," in *Proc. 5th Int. Conf. Internet Things (IoT)*, Oct. 2015, pp. 122–129.

[132] D. Thiele and R. Ernst, "Formal analysis based evaluation of software defined networking for time-sensitive Ethernet," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, 2016, pp. 31–36.

[133] A. Gopalakrishnan, "Applications of software defined networks in industrial automation," Tech. Rep., 2014.

[134] G. Kálmán, "Applicability of software defined networking in industrial Ethernet," in *Proc. 22nd Telecommun. Forum Telfor (TELFOR)*, Nov. 2014, pp. 340–343.

[135] S. B. H. Said, Q. H. Truong, and M. Boc, "SDN-based configuration solution for IEEE 802.1 time sensitive networking (TSN)," *ACM SIGBED Rev.*, vol. 16, no. 1, pp. 27–32, Feb. 2019.

[136] S. Clarke, B. Fitzgerald, P. Nixon, K. Pohl, K. Ryan, D. Sinclair, and S. Thiel, "The role of software engineering in future automotive systems development," *SAE Int. J. Passenger Cars-Electron. Elect. Syst.*, vol. 1, pp. 544–552, Jan. 2008.

[137] B. Steurich, K. Scheibert, A. Freiwald, and M. Klimke, "Feasibility study for a secure and seamless integration of over the air software update capability in an advanced board net architecture," SAE Tech. Paper 2016-01-0056, 2016.

[138] A. Bulmus, A. Freiwald, and C. Wunderlich, "Over the air software update realization within generic modules with microcontrollers using external serial FLASH," SAE Tech. Paper 2017-01-1613, 2017.

[139] J.-Y. Chung and K.-M. Chao, "A view on service-oriented architecture," Tech. Rep., 2007.

[140] M. Maul, G. Becker, and U. Bernhard, "Service-oriented EE zone architecture key elements for new market segments," *ATZelektronik Worldwide*, vol. 13, no. 1, pp. 36–41, Feb. 2018.

[141] L. C. Coronado-Garcia, J. A. Gonzalez-Fuentes, P. J. Hernandez-Torres, and C. Perez-Leguizamo, "An autonomous decentralized service oriented architecture for high reliable service provision," in *Proc. 10th Int. Symp. Auto. Decentralized Syst.*, Mar. 2011, pp. 327–330.

[142] G. L. Gopu, K. V. Kavitha, and J. Joy, "Service oriented architecture based connectivity of automotive ECUs," in *Proc. Int. Conf. Circuit, Power Comput. Technol. (ICCPCT)*, Mar. 2016, pp. 1–4.

[143] J. Rox and J. Gacnik, "Developing distributed vehicle functions with service-oriented architectures," *ATZ Worldwide*, vol. 117, no. 9, pp. 26–29, Sep. 2015.

[144] M. Traub, A. Maier, and K. L. Barbehön, "Future automotive architecture and the impact of IT trends," *IEEE Softw.*, vol. 34, no. 3, pp. 27–32, May 2017.

HAILONG ZHU received the B.S. degree in measurement and control technology and instrumentation from Northwestern Polytechnical University, in 2009, and the Ph.D. degree in control science and engineering from Tsinghua University, in 2019. He is currently a Postdoctoral Researcher with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications. His research interests include network and communication technologies in automated driving, deterministic networks, distributed artificial intelligence, and industrial control networks and systems.

WEI ZHOU is currently pursuing the M.Eng. degree with the Department of Automation, Tsinghua University, Beijing, China. Her research interests include in-vehicle networks, deterministic communication, and intelligent vehicles.

ZHIHENG LI (Member, IEEE) received the Ph.D. degree in control science and engineering from Tsinghua University, Beijing, China, in 2009, where he is currently an Associate Professor with the Department of Automation. He is also with the Tsinghua Shenzhen International Graduate School, Shenzhen, China. His research interests include traffic operation, advanced traffic management systems, urban traffic planning, and intelligent transportation systems. He serves as an Associate Editor for the IEEE Transactions on Intelligent Transportation Systems.

LI LI (Fellow, IEEE) is currently an Associate Professor with the Department of Automation, Tsinghua University, China, where he is involved in artificial intelligence, intelligent control and sensing, intelligent transportation systems, and intelligent vehicles. He has published over 80 SCI indexed international journal articles and over 50 international conference papers as a first/corresponding author. He serves as a member of the Editorial Advisory Board of Transportation Research Part C: Emerging Technologies. He also serves as an Associate Editor for the IEEE Transactions on Intelligent Transportation Systems.

TAO HUANG (Senior Member, IEEE) received the B.S. degree in communication engineering from Nankai University, Tianjin, China, in 2002, and the M.S. and Ph.D. degrees in communication and information systems from the Beijing University of Posts and Telecommunications, in 2004 and 2007, respectively. He is currently a Professor with the Beijing University of Posts and Telecommunications. His current research interests include network architecture and software-defined networking.

• • •