telecom

MDPI

Review

Blockchain-Based Solutions for UAV-Assisted Connected Vehicle Networks in Smart Cities: A Review, Open Issues, and Future Perspectives

Paulo Álvares, Lion Silva and Naercio Magaia *

LASIGE, Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal; palvares@lasige.di.fc.ul.pt (P.Á.); lsilva@lasige.di.fc.ul.pt (L.S.)
* Correspondence: ndmagaia@fc.ul.pt; Tel.: +351-217500489

Abstract: It had been predicted that by 2020, nearly 26 billion devices would be connected to the Internet, with a big percentage being vehicles. The Internet of Vehicles (IoVa) is a concept that refers to the connection and cooperation of smart vehicles and devices in a network through the generation, transmission, and processing of data that aims at improving traffic congestion, travel time, and comfort, all the while reducing pollution and accidents. However, this transmission of sensitive data (e.g., location) needs to occur with defined security properties to safeguard vehicles and their drivers since attackers could use this data. Blockchain is a fairly recent technology that guarantees trust between nodes through cryptography mechanisms and consensus protocols in distributed, untrustful environments, like IoV networks. Much research has been done in implementing the former in the latter to impressive results, as Blockchain can cover and offer solutions to many IoV problems. However, these implementations have to deal with the challenge of IoV node's resource constraints since they do not suffice for the computational and energy requirements of traditional Blockchain systems, which is one of the biggest limitations of Blockchain implementations in IoV. Finally, these two technologies can be used to build the foundations for smart cities, enabling new application models and better results for end-users.

Keywords: blockchain; internet of vehicles; UAV; distributed ledger; smart cities

check for updates

Citation: Álvares, P.; Silva, L.; Magaia, N. Blockchain-Based Solutions for UAV-Assisted Connected Vehicle Networks in Smart Cities: A Review, Open Issues, and Future Perspectives. *Telecom* **2021**, *2*, 108–140. https://doi.org/10.3390/telecom2010008

Received: 19 December 2020
Accepted: 8 March 2021
Published: 12 March 2021

1. Introduction

The Internet of Vehicles (IoV) is a decentralized technology that expands from the preexisting Vehicular Ad-hoc Networks (VANETs) [1] while aiming at large-scale city-level coverage. IoV objectively offers the means of communication between nodes (i.e., vehicles, infrastructure, sensors, among others) in a network by unifying various technologies.

IoV aims to provide services and solutions that improve comfort, fuel consumption, and traffic congestion seamlessly or that allow for media streaming, file sharing, among others, all the while ensuring the satisfaction, security, and privacy of vehicles and users in the network and real-life [2].

IoV faces challenges on many fronts [3]: dealing with resource constraints of devices in the network (e.g., low battery, reduced computational power), which limits algorithm and application development; the lack of means to deal with constant node mobility in a seamless fashion, which can increase latency and decrease performance due to frequent handovers; or even in big data management, since devices in IoV networks constantly generate data, it can cause network congestion and reduce performance.

However, the most important challenge to face is security and privacy due to the shared data's sensitive nature (i.e., location, state of the car, personal data, pictures). This is because vehicles work in unprotected, heterogeneous, and vulnerable environments that cannot ensure trust between nodes by themselves and open up the possibility for cyber-attacks and exploitations [4]. Therefore, it is imperative to deal with, as one mistake or malicious attack could lead to accidents and, in worst-case scenarios, endanger human lives. The trust problem could be solved by introducing a third-party authority that validates every transaction between devices, but not only does this introduce a single point of failure, it also decreases the throughput of operations, which is not beneficial in IoV scenarios.

The smart city is a concept that aims at utilizing various types of technologies together, at the same time, to manage the city resources (e.g., water, energy, gas, transportation, network access, among others) to improve the lives of the population of that city and the administration of such resources [5]. The Internet of Things (IoT) is the major construction block of this concept. However, IoV working as an extension of the IoT baseline can boost the improvements that are already expected from smart cities, especially with unmanned aerial vehicles (UAV), which bring to the table new application options and opens doors to new solutions for existing problems.

The Blockchain, firstly presented as the underlying technology of Bitcoin [6], is a distributed ledger (or database) of transactions between nodes in a network that ensures various security and privacy properties, such as data authentication, data non-repudiation, privacy, traceability, and immutability, through the implementation of cryptography, digital signatures and consensus protocols. Through these characteristics, and later the implementation of Smart Contracts, Blockchain can solve decentralized networks' trust problem without a third-party authority, removing the single point of failure problem. Blockchain also provides the benefit of interoperability since it is deployed on top of an overlay Peer-to-Peer (P2P) network. The sum of all these assets makes Blockchain an ideal technology to solve the security and privacy problems of IoV. Even with the great benefits Blockchain presents, there is a problem with the underlying technologies and their effects on IoV and its restrictions. Typical Blockchain implementations use consensus protocols that propose a hard-to-solve problem that requires a lot of computational power and energy consumption. That proves to be detrimental to resource-constrained devices (i.e., sensors) that proliferate in IoV networks.

While there are already published surveys of this topic, such as Singh et al. [2] and Mollah et al. [7], most of the existing surveys only focus on one technology (either IoV or Blockchain), and the papers that mention both foci, mostly, on presenting their project (which limits the expression of detail, since the authors will only write about what they require to justify their decisions), and not to present in-depth, introductory research on both subjects.

This article reviews IoV and Blockchain, presenting them in a welcoming way for newcomers, early developers, and practitioners. It explains how these technologies work, how they come together, and the tools that are most popular to help in creating new solutions while going in-depth on these subjects and inciting readers to start developing and learning more about the subject. We also discuss smart cities and how UAV-assisted IoV and Blockchain can help to bring this concept to life and improve the results obtained from their implementation.

Our contributions are as follows:

- We review the IoV concept, its architectures, benefits, and flaws, how applications are grouped for development in this technology, quickly introducing the rising Social Internet of Vehicles, and presenting the main challenges and problems to be solved.
- We present the concept of UAV and smart cities and how UAV-assisted IoV can help realize the smart cities concept and the underlying technologies it requires.
- We review the Blockchain concept, its origins, and the technologies behind it that give it its properties and how they do it. We also list the properties of the three types of Blockchain, explain how the Blockchain assures certain security properties and how these properties correlate to the security challenges of the IoV.
- We present Blockchain-enabled solutions for IoV that offer various functionalities, such as safe data transmission, transport, or increase the performance of network communications. We also present Blockchain and UAV-assisted IoV based applications

for smart cities, focusing on what opportunities there are for their implementations given their attributes and the smart cities' requirements.

- We highlight and compare popular tools used to develop Blockchain-based solutions and network simulations, detailing some of their benefits and downsides, such as uniqueness, availability, pluggability, and efficiency.
- We discuss major challenges and open issues yet to be solved. We also present future perspectives such as the integration of novel network paradigms or machine learning.

The article is structured as follows. Section 2 presents the state-of-the-art of the Internet of Vehicles, mentioning popular definitions, architectures, applications, security challenges, and the IoV-smart city interaction. Section 3 presents the state-of-the-art of Blockchain technology, including underlying technologies like the consensus algorithm and cryptography, up to security properties and applications. Section 4 summarizes some applications of Blockchain on the Internet of Vehicles, focusing on how Blockchain is implemented, for what purposes, and how the solutions are tested and evaluated, and summarization of possible applications of IoV and Blockchain-enabled solutions for the smart city paradigm. Section 5 presents UAV-assisted IoV and Blockchain solutions for smart cities. Section 6 lists various network simulators used to replicate IoV scenarios and various Blockchain platforms that can be used to run a Blockchain system, with details on each technology presented. Sections 7 and 8 present open challenges of integrating Blockchain on IoV, and then both of these on smart cities and future developments to optimize this goal, respectively. Section 9 presents concluding remarks.

2. Internet of Vehicles and Smart Cities

It had been predicted by James et al. [8] that, by 2020, there would be, at minimum, 26 billion devices connected in some form to the Internet. D. Gary [9] predicted that this number would reach double that, up to 50 billion. Out of this number, vehicles are expected to occupy a large percentage, making it extremely important to invest in research on open problems and vulnerabilities that this number of connected devices brings out to the open. The main purpose of IoV is to enhance the efficacy, efficiency, and comfort of transportation and the facility level of cities, reduce costs, and ensure customer satisfaction [2].

2.1. Definition

With the growth of entities in VANETs over the years, new requirements appear that have to be fulfilled to appeal to the user's needs. In a vehicular network, vehicles frequently produce enormous amounts of data (from internal mechanical data to data related to the road or traffic state). The processing, analyzing, and evaluation of these large amounts of data is an arduous task that VANETs cannot handle due to the limited processing power of their devices. The limited applications of entrusted Internet services [10] and the connection/disconnection of vehicles due to getting in and out of the coverage area [11] are also noticeable constraints in VANETs, which drive the evolution towards the IoV. As such, vehicles in VANETs need to become "smart" objects that work cooperatively to ease these tasks' weight. This smart cooperation marks the line where VANETs start evolving into the IoV, as defined by the work of Islam et al. [12].

Another way IoV demarks itself from VANETs is in the sense that it aims at large coverage areas (city-scale or even global) and by aiming the integration of two technological visions: vehicle's networking and vehicle's intelligence, as proposed by Yang et al. [13], with a focus on integrating objects (i.e., humans, vehicles, units) and environments as to build an intelligent network. The coalition of smart vehicular systems and cyber-physical systems brings the possibility of developing a global network that offers services and gives quality-of-life improvements to drivers and service providers, helping to reduce traffic congestion, pollution, and accidents.

Moreover, IoV can be seen as an application of an IoT technology in an Intelligent Transportation System (ITS) technology, which is also the result of merging three different networks: the inter-vehicular network (from the vehicle to other vehicles), the intravehicular network (inside the vehicle-cyber-physical system) and the vehicular mobile Internet (from the vehicle to other objects on the network) [14]. It is an enormous distributed system for wireless communication and data exchange on a *vehicle-to-everything* (V2X) mode with defined protocols and data interaction standards, like IEEE 802.11p. These modes are represented in Figure 1.

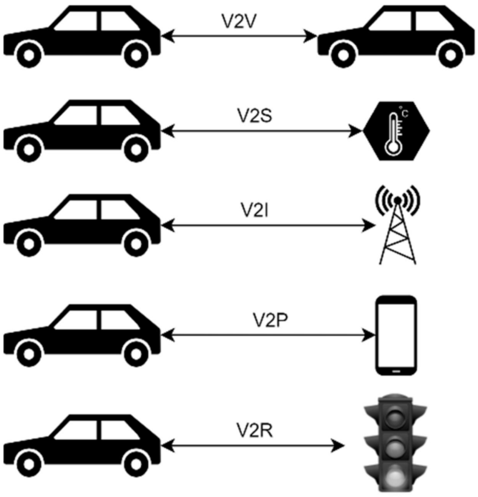


Figure 1. V2X communication modes. In order: vehicle-to-vehicle (V2V), vehicle-to-sensor (V2S), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P), vehicle-to-roadside unit (V2R).

2.2. Architecture

In terms of architecture, IoV can be interpreted in various ways. For example, researchers have divided the architecture into three [15], four [16], or even five layers [10]. Figure 2 presents a side-by-side comparison of the four architectures detailed below.

2.2.1. Three-Layer Architecture

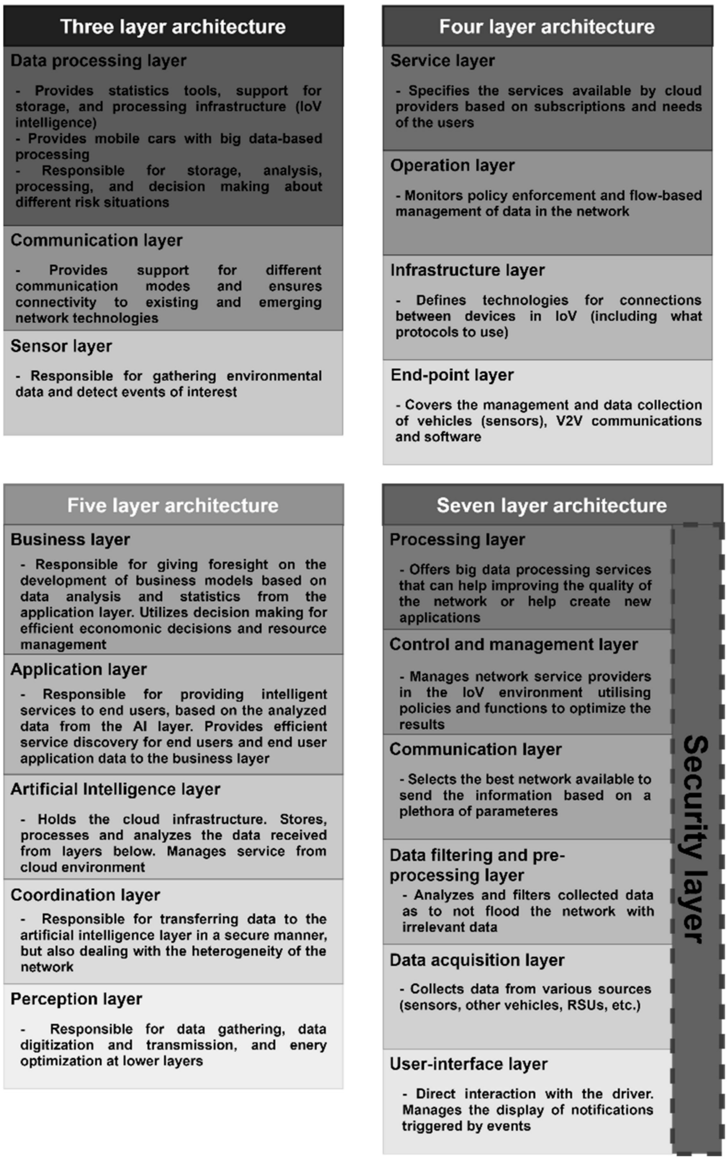
The three-layer architecture, presented by Nanjie [15], is divided based on the interactions with the technologies in the IoV environment, as follows:

- *Sensor layer* is responsible for the sensors in the vehicles and surrounding infrastructure.
- *Communication layer* is responsible for the wireless connections between entities in various vehicle-to-everything (V2X) modes (see Figure 1).
- *Data processing layer* is responsible for holding statistics tools and storage (acts as the IoV network's intelligence and provides big data processing to vehicles, providing decision making in risk situations).

2.2.2. Four-Layer Architecture

The four-layer IoV system architecture, presented by CISCO [16], is divided as follows:

- *End-point layer*, which is responsible for the vehicles (and sensors), V2V communications, and software.
- *Infrastructure layer*, which handles all communication technologies used by entities.
- *Operation layer*, responsible for policy enforcement and flow-based management of network providers.
- *Service layer* that handles the services offered by the cloud infrastructure that is connected to the network.



Three layer architecture

Data processing layer

- Provides statistics tools, support for storage, and processing infrastructure (IoV intelligence)
- Provides mobile cars with big data-based processing
- Responsible for storage, analysis, processing, and decision making about different risk situations

Communication layer

- Provides support for different communication modes and ensures connectivity to existing and emerging network technologies

Sensor layer

- Responsible for gathering environmental data and detect events of interest

Four layer architecture

Service layer

- Specifies the services available by cloud providers based on subscriptions and needs of the users

Operation layer

- Monitors policy enforcement and flow-based management of data in the network

Infrastructure layer

- Defines technologies for connections between devices in IoV (including what protocols to use)

End-point layer

- Covers the management and data collection of vehicles (sensors), V2V communications and software

Five layer architecture

Business layer

- Responsible for giving foresight on the development of business models based on data analysis and statistics from the application layer. Utilizes decision making for efficient economonic decisions and resource management

Application layer

- Responsible for providing intelligent services to end users, based on the analyzed data from the AI layer. Provides efficient service discovery for end users and end user application data to the business layer

Artificial Intelligence layer

- Holds the cloud infrastructure. Stores, processes and analyzes the data received from layers below. Manages service from cloud environment

Coordination layer

- Responsible for transferring data to the artificial intelligence layer in a secure manner, but also dealing with the heterogeneity of the network

Perception layer

- Responsible for data gathering, data digitization and transmission, and enery optimization at lower layers

Seven layer architecture

Processing layer

- Offers big data processing services that can help improving the quality of the network or help create new applications

Control and management layer

- Manages network service providers in the IoV environment utilising policies and functions to optimize the results

Communication layer

- Selects the best network available to send the information based on a plethora of parameteres

Data filtering and pre-processing layer

- Analyzes and filters collected data as to not flood the network with irrelevant data

Data acquisition layer

- Collects data from various sources (sensors, other vehicles, RSUs, etc.)

User-interface layer

- Direct interaction with the driver. Manages the display of notifications triggered by events

Security layer

Figure 2. Comparison between the various layered architectures for IoV.

2.2.3. Five-Layer Architecture

The five-layer architecture, proposed by Kaiwartya et al. [10], is divided as follows:

- *Perception layer* that handles data gathering, data digitization, and transmission, and energy optimization at lower layers.
- *The coordination layer* is not only responsible for transferring data to the artificial intelligence (AI) layer in a secure manner but also being responsible for dealing with the heterogeneity of the network structure, unifying received information.
- *The artificial intelligence layer* holds the cloud infrastructure, which stores, processes, and analyzes the data from layers below, utilizing this analysis for decision making while also managing cloud systems' services.
- *The application layer* is responsible for providing intelligent services to end-users, based on the processed and analyzed information of the AI layer, which serves for service discovery from smart applications.
- *Business layer*, a novelty from the last two architectures, which is responsible for giving foresight on the development of business models, based on data analysis and statistics, which come from the *application layer* and is later transformed by analysis tools, while utilizing decision making for budgeting and optimization usage of resources.

2.2.4. Seven-Layer Architecture

The work of Contreras-Castillo et al. [11] refers that the three and four-layered proposed models have weaknesses and do not contemplate important concerns of IoV systems, such as the need for the existence of layers for security and data dissemination/transmission, the communication between the driver and the vehicle (interface), and network congestion.

The authors then present a complete layered architecture interpretation which divides the IoV into the following seven layers:

- *Processing layer* that, as the name suggests, is responsible for big data processing using various cloud computing technologies, which helps develop strategies for business models.
- *The control and management layer* manages the network service providers in IoV, utilizing policies and functions for that objective.
- *The communication layer* is responsible for selecting the best network to serve the needs of the user.
- *The data filter and pre-processing layer* analyze data to avoid network congestion by the transmission of irrelevant information.
- *The data acquisition layer* collects data from their respective sources (i.e., sensors, infrastructure connection points, other vehicles).
- *The user interface layer* is responsible for dealing with how the information is passed from the vehicle and sensors to the driver and users inside the car.
- Finally, the *Security layer*, transversal to the six ones noted before, and is one of the major differences between this architecture and the first two presented before, being responsible for all security properties guaranteed, using proposed solutions to mitigate the damage from cyberattacks and malfunctions.

2.2.5. Comparison

It is possible to conclude that the four architectures share some similarities between themselves. All the architectures have the following characteristics: layers responsible for sensors, vehicles, and data collection (sensor, end-point, perception, and data acquisition layers); layers that handle and manage communications between entities (communication, infrastructure, perception, and coordination, and communication layers); the data processing, service, artificial intelligence and application, and the processing layers which are all responsible for data processing and decision making, among other services, usually provided by the cloud infrastructure. The management of network providers is only present in the four, five, and seven-layered architectures in the operation, coordination, and control and management layers. Only the five and seven-layered architectures have layers responsible for or help in the development of business models: the business and processing layers, respectively. Finally, only the seven-layered architecture has layers that handle data pre-processing and, most importantly, security. While the five-layer architecture does not present a security layer, they mention the secure transmission of information between layers through the usage of protocols to achieve a safe exchange of data.

Each architecture presented has benefits and downsides, as the increasing complexity maybe be useful for some research and not so much for others. Given this, the choice of what architecture framework should be followed is in the researcher's hands, which must make their decision based on what their project requires from the IoV architecture and what aspects are relevant to consider to fulfill the needs of the work in question.

Benalia et al. [3] state that the three and five-layer architectures do not consider the security challenges of IoV. It is also possible to add the four-layer architecture to this affirmation, giving the analysis above. The authors further noticed that even the most complete one, i.e., seven-layered architecture, does not take advantage of new paradigms and technologies, such as using 4G/5G communications to deal with high latencies and low bandwidth or even using computing paradigms, such as edge and fog computing, for pre-processing and data management.

They then present a three-layered generic architecture, which is presented in Figure 3, and is organized as follows:

- *Terminal layer* (or *IoV layer)* that is responsible for gathering information on the road through millimeter-wave (*mm-wave*) *V2X* communication modes (*mm-wave* is a new 5G network technique that can promise multigigabit communication services [17]).
- *Edge computing layer* that has:
    - *Fog infrastructure sublayer* that is responsible for data processing, analysis, computing storing, networking, and security.
    - *Fog virtualization sublayer*, which is further divided in:
        - *the upper level* has two planes that manage the network (*control plane* that manages the local and global data planes and provides programmability and flexible management, and *global data plane* that contains forwarding and data processing devices).
        - *the lower level* contains the local data plane, which has nodes that forward and receive data to and from fog and cloud computing.
    - *Fog service sublayer* which offers traditional fog services adapted to IoV, such as:
        - *Fog Vehicular Infrastructure as a Service* (offers data processing, storage, and analysis while having the capability of adopting another infrastructure to serve other needs).
        - *Fog Vehicular Platform as a Service* (offers different operational systems and computational environments to ensure the fulfillment of the heterogeneous needs of drivers and vehicles).
        - *Fog Vehicular Software as a Service* (offers fog-based software divided into user applications and safety applications, which is be explained below).
- *The cloud computing layer* provides services similar to the latter four mentioned architectures: big data processing, analysis, storage, and analytics tools, which can then help develop business models.

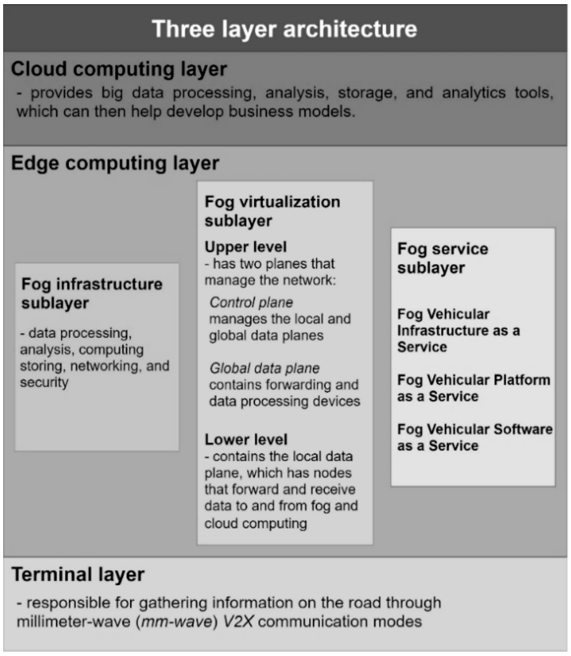

Three layer architecture
Cloud computing layer
- provides big data processing, analysis, storage, and analytics tools, which can then help develop business models.
Edge computing layer
Fog infrastructure sublayer
- data processing, analysis, computing storing, networking, and security
Fog virtualization sublayer
Upper level
- has two planes that manage the network:
Control plane
manages the local and global data planes
Global data plane
contains forwarding and data processing devices
Lower level
- contains the local data plane, which has nodes that forward and receive data to and from fog and cloud computing
Fog service sublayer
Fog Vehicular Infrastructure as a Service
Fog Vehicular Platform as a Service
Fog Vehicular Software as a Service
Terminal layer
- responsible for gathering information on the road through millimeter-wave (mm-wave) V2X communication modes

Figure 3. A three-layer architecture that covers the flaws of the last four, taking into account security and utilizing new network computing paradigms to achieve better performance and ease data dissemination in IoV.

This final layered architecture was made with the intent to allow more flexible dissemination of data in IoV while taking into account the advantages of new rising technologies, such as cloud and fog computing paradigms that increase network performance. While this architecture is the most complete, not every project needs the same notion of architecture to fulfill its needs, and it all depends on what are the required aspects that need to be considered to develop the idea and what each architecture layout has to offer. If the project does not or cannot envelop fog computing techniques, then the best architecture to follow is one of the other four. If the architecture is not a big factor, then the first three-layer architecture is the best one to use as the basis for the project in question, aiming for simplicity.

Computing paradigms for IoV have also acquired popularity in the research community to appropriately cater to the new requirements that these environments impose to achieve the best performance for users and applications [18].

2.3. *Applications*

Applications for the IoV vary in functionality and objectives. Researchers have divided these applications according to the services they aim to provide and in what shape or form they provide them.

2.3.1. Taxonomy

According to Yang et al. [13] and Wu et al. [19], IoV applications can be divided into categories. Figure 4 presents a taxonomy of IoV applications.

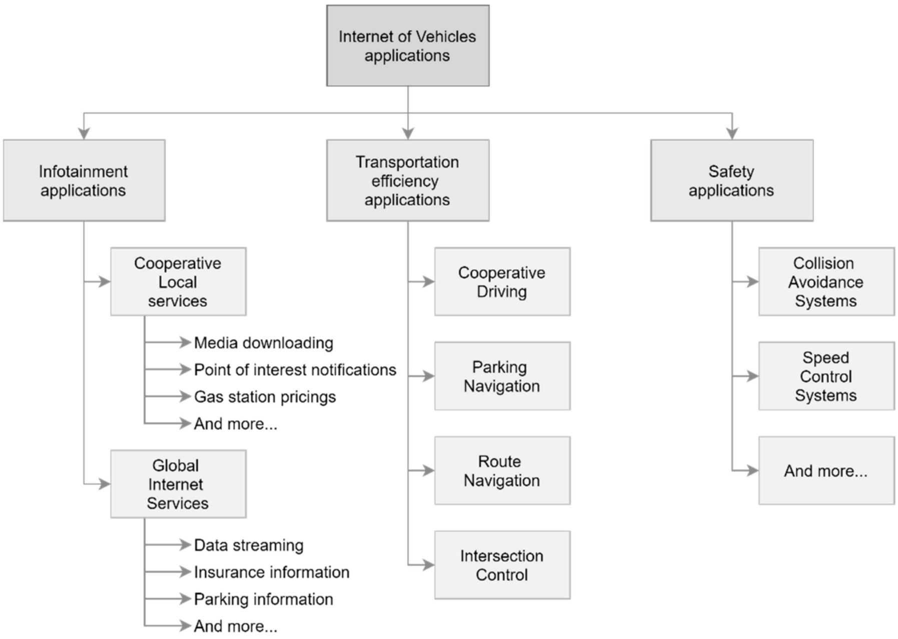


Figure 4. Taxonomy of applications for the Internet of Vehicles.

Firstly, the *User* or *Infotainment applications*, which are applications that provide value-added services and have multiple requirements in terms of real-timeliness or communications. These applications range from video/music streaming, file transfer to weather information and local point of interest information. The services these applications offer can also be further divided into:

- *cooperative local services*—relates to infotainment from local-based services (i.e., point of interest notification and media downloading).
- *global internet services*—relates to data obtained from services like insurance management and parking zone information, which are constantly updated.

Some *User applications* that have been developed are, for example, the Cooperative Video Streaming over Vehicular Networks, [20], which provides basic QoS over 3/3.5G networks. In this application, helpers (other vehicles) can voluntarily share bandwidth with requesters and improve QoS by transmitting video through the established dedicated short-range communications (DSRC) channels.

Next, Yang et al. [13] and Wu et al. [19] diverge on how they approach the taxonomy of the rest of the applications. While the latter has two more categories, which are *Safety applications* and *Transportation efficiency applications*, the former does not do this differentiation. *Safety* applications aim at providing services to ensure safe driving through notifications and, possibly, car control. Given that it is the most researched technology, *Safety applications* mainly refer to collision avoidance systems (CAS), vehicle-based systems that serve two purposes:

- *collision warning*—warns the driver that a collision is about to occur, and it can also warn when a collision happened down the road to prevent congestion.
- *driver assistance*—controls the car for steady-state or emergency intervention (e.g., braking before a collision).

CAS can be extended to cooperative collision avoidance systems (CCAS), where vehicles share their information with neighboring vehicles to diminish even more the chance of collisions.

There is also some research on speed limiting applications that prevent speeding. Abdelsalam et al. [21] developed a system based on RFID technology, with RF transmitters on roads, RF receiver modules on vehicles, and engine control units to establish speed values. One of the earliest works with CCAS was CarTALK 2000 [22], which developed *Cooperative assistance systems*, information, and warning functions.

Finally, *Transportation efficiency applications* are applications that aim at improving efficiency for vehicles and drivers similarly by providing solutions that improve efficiency in things like fuel consumption and travel time. These applications can be divided into four different categories, based on their main work environment and objectives:

- *intersection control* is the biggest research area in efficiency solutions. These applications aim at controlling traffic at intersections, which requires complex solutions to reduce waiting time and retain fairness. They can be split into two types of approaches:
  - ○ *traffic-light-based* applications schedule traffic lights based on traffic volume with V2V (the cluster of vehicles at the intersection makes decisions) or V2I (a controller makes the decisions) communication approaches.
  - ○ *non-traffic-light-based* applications apply maneuver manipulation, controlled by the intersection controller, to drive on the intersection or vehicle scheduling algorithms.
- *route navigation*, also known as vehicular network-based navigation, it is investigated to avoid the negatives of using GPS-based approaches. It utilizes parameters such as real-time traffic information, fuel consumption, speed, and road condition data, among others, to choose the best route for the vehicle.
- *cooperative driving* applications that aim at coordinating a group of vehicles, so they drive in the same way as one. This improves energy efficiency, traffic flow and helps prevent accidents.
- *parking navigation* applications that use algorithms to track optimal routes that lead to the closest available parking zones.

Huang et al. [23] developed a cooperative adaptive driving application, a variation of cooperative adaptive cruise control, utilizing mobile edge computing and platooning techniques to avoid accidents and improve traffic flow. Kowshik et al. [24] present an algorithm for multiple vehicles that enforces safety in intelligent intersections through time slot allocation. Wu et al. [25] developed algorithms that schedule vehicles with V2V communications based on distributed mutual exclusion. Chen et al. [26] and Collins et al. [27] present route navigation systems based on the vehicle's fuel consumption and traffic congestion on roads, respectively.

2.3.2. Social Internet of Vehicles

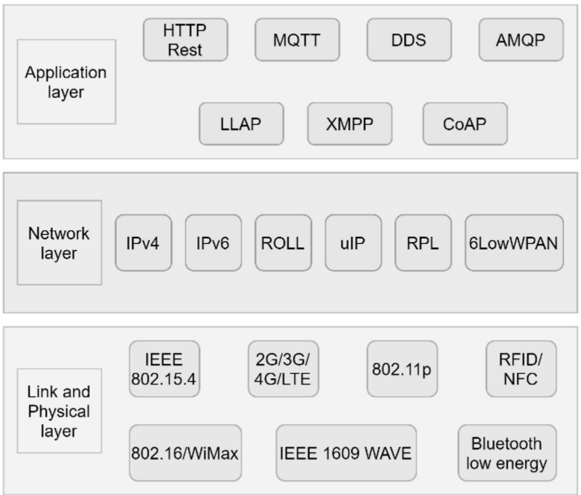
As an evolution of the paradigm of IoV, the Social Internet of Vehicles (SIoV) allows for the creation and management of relationships between vehicles in IoV, based on the context they are inserted, network architecture, or application requirements, behaving like a social network of vehicles [28]. These webs of relationships can introduce vehicles to other participants and enable vehicles to autonomously create new relationships that prove to be beneficial, be it because they improve traffic efficiency (through sharing road information) or because these vehicles have data useful for installed applications [29]. This paradigm opens doors to new applications and can help the development of the applications mentioned above, given the focus on data exchange and trust relationships between vehicles and between infrastructural nodes. Adding to this, the usage of Deep Learning in the application scenario is also a possibility to get better results, as it has been researched and evaluated for IoT [30].

2.4. Main Challenges

IoV faces challenges on many fronts, especially because it is such a recent paradigm with a lack of research.

2.4.1. Standardization

One of the main challenges in IoV is the need to develop standards and protocols to achieve interoperability and ease the development of applications. Many consortia and organizations are trying to develop these standards and protocols that answer to the requirements posed by IoV, and there are already some prominent protocols for these systems, as showcased in Figure 5. These protocols are utilized in IoT as well, as the two paradigms share similarities [31]. Even so, with the imminent turn to the fifth generation, i.e., 5G, these protocols will probably need to be revised.



HTTP REST - HyperText Transfer Protocol REpresentational State Transfer;
MQTT - Message Queueing Telemetric Transport;
DDS - Data Distribution Service;
AMQP - Advanced Message Queueing Protocol;
LLAP - Lightweight Local Automation Protocol;
XMPP - Xtensible Message and Presents Protocol;
CoAP - Constrained Aplication Protocol
IPv4 - Internet Protocol version 4;
IPv6 - Internet Protocol version 6;
ROLL - Routing Over Low power and Lossy networks;
uIP - micro Internet Protocol;
RPL - Routing Protocol for Low power and Lossy Networks;
6LowWPAN - IPv6 over Low power wireless personal area networks
RFID - Radio Frequency IDentification;
NFC - Near Field Communication

Figure 5. Protocols on the Internet of Vehicles.

2.4.2. Resource Constraints

While pre-processing information can lead to less network congestion and more accurate results in higher layers, the computing resources (e.g., computing power, energy power, storage space, etc.) limitations of devices in lower layers can limit the usefulness of doing it and even end up hindering the security aspect of the system, since the usage of cryptographic algorithms can become limited. The challenge here can lead both ways: make more powerful hardware or make algorithms for pre-processing/security that requires fewer computing resources while maintaining the minimum levels of satisfaction for their implementation to be worth it.

2.4.3. Service Management

Given the large number of services that can be provided in IoV environments, it becomes a challenge for vehicles to manage services to obtain optimal solutions for them at a given point in time, minimizing costs and delivery time.

2.4.4. Node Mobility

In IoV environments, the nodes (vehicles) are constantly moving as they move in the real world. This regular mobility and the rapid topology changes present a challenge when trying to keep vehicles connected to the network and the Internet since they lead to packet loss, link failures, and even frequent network disconnections. Nevertheless, there is a difference between IoV and regular networks affected by mobility. Specifically, IoV mobility is somewhat predictable since vehicles and their movement are limited to road layout and topology, road signs, and other vehicles.

2.4.5. Scalability

It is also extremely important that the IoV can have numerous amounts of devices connected at a given time without a significant decrease in performance, ensuring high scalability. Otherwise, it will not serve its purpose of achieving large-scale coverage and guaranteeing that services reach the users.

2.4.6. Data Dissemination, Routing, and Management

The IoV also opens doors to the integration of different services and technologies [32]. With the heterogeneity of entities, services, and networks in IoV environments, disseminating and routing data from the source to the target destination while guaranteeing a high quality of service becomes difficult. These heterogeneous environments also limit the coordination and collaboration of different networks and subnetworks. There is also the problem of data management, given that the entities in IoV generate massive amounts of data, making it difficult to aggregate, storage, process, analyze and provide decision making over this data. The challenge here is to design new protocols and schemes that can ensure that not only the data reaches the target under certain QoS limits but also enable cooperation between networks and ease big data management.

2.4.7. Security

The tolerance to heterogeneity and interoperability in IoV systems, which is strictly needed to allow these thousands of different vehicles, sensors, and other components to communicate in the network, brings out a bigger need for data security. Zhang [4] states that vehicles operate in unprotected and vulnerable environments, with vulnerabilities in the cloud, V2V, and local communications. The security vulnerabilities in communications in V2X modes could open the possibility for cyber-attacks by manipulating data streams or connection points [11]. Another issue is the possibility of hacking vehicles, which could lead to accidents and fatalities. In 2015, hackers at the Black Hat hacking conference demonstrated how they hacked vehicles through their PCs, far away from the vehicle, and it should be seen as a threat to security [11]. There's also the problem of misbehaving vehicles that interfere in the exchange and dissemination of data in the network [33]. As such, researchers have proposed various ideas on how to reduce these vulnerabilities while fulfilling the security requirements of IoV and safeguarding the network and its participants. Mokhtar et al. [34] and Sharma [35] have listed the security requirements for IoV, which are summarized in Table 1.

In SIoV, with an even bigger need for data availability, connectivity, and autonomy of vehicles, there is a problem in maintaining security, especially, the privacy may be compromised due to secondhand data sharing. The new applications can also emphasize security vulnerabilities due to inherent communication issues or no control in data usage.

Table 1. Security requirements for IoV [34].

Security Requirement	Description
Data authentication	Vehicles identities should be verified when transferring data
Data integrity	Sent and received data should be verified to ensure correct data transferal
Data confidentiality	Data transmission between vehicles should be secret
Access control	Vehicles should be allowed to access services they are entitled to
Data non-repudiation	Vehicles should not be able to deny the authenticity of another vehicle
Availability	Communication between vehicles should be ensured even under bad conditions in the event of an attack
Anti-jamming	Malicious vehicles cannot interrupt communications between other vehicles
Impersonation	A vehicle cannot impersonate another entity in the network
ID traceability	A vehicle's identity can be retrieved from sent messages
Vehicle privacy/anonymity	Sent messages can only be accessed by authorized vehicles and remote nodes. The vehicle's identity should be hidden.

2.5. The UAV and the Smart City Paradigm

2.5.1. UAV

UAV are a type of aerial vehicle that does not require a pilot and can move and interact with the environment independently. They are also a part of the unmanned aircraft system (UAS), which requires a UAV, a ground control system (GCS), and a communication link [36]. UAVs can fly autonomously with pre-arranged flight plans or create their own plan mid-flight. The UAV consists of two parts: the machines themselves and the payload (what modules and other technologies it carries/can carry to provide further services for users and connected devices).

Studies are being made on the usage of UAV as drone base stations (DBS) [37,38] which complement other base stations and expand the already existing networks by increasing coverage, availability, reducing latency, network congestion, or even in the spread of 5G in hard to reach zones [39], as seen in Figure 6. Shi et al. [40] list the benefits of UAV usage as base stations:

1. Drones can move immediately, in real-time, to provide dynamic coverage according to necessity. This makes the overall connectivity robust as they can change as the environment does too.
2. As the deployment is through the air, it is not necessary to rent sites. Moreover, costs related to cables, towers will diminish or simply be non-existent.
3. Line-of-Sight (LoS) is one reliable communication among the possible types of connections since both transmitter and receiver are aligned, and the connection is direct. LoS' connections are much easier through open space where drones are supposed to be. So, reliable connections, at least at a certain level, would be possible even inside cities.
4. UAV swarms is a novel multi-purpose technique to provide services. A connected swarm can guarantee ubiquitous connectivity to end-users on the ground. In a nutshell, end-users benefit from higher data rates with lower latency, and providers can lower their costs and raise profits simultaneously.

2.5.2. Smart Cities

As stated before, the smart city is an extensive concept. It is a combination of perspectives with the purpose to optimize resources and their management sustainably and efficiently, as to improve the life of the citizens. Founoun et al. [41] list some *smart* characteristics common to the smart cities' concepts:

- *Smart Living*—any individual should be able to have quality health conditions, safety, and cultural and educational access, also good accessibility in its housings.

- *Smart Mobility*—any individual should be able to access its territory from local to a national scope. Intelligent Transportation Systems intermediate these accesses so that they can be made safely and sustainably.
- *Smart Environment*—any smart city should deliver good land use planning, reasonable pollution control, proper natural resource usage.
- *Smart Economy*—any smart city should promote local product use (local economy), incentivize entrepreneurship, and innovation culture based on e-services.
- *Smart People*—any individual should be invited to incorporate life-long learning precepts, social and ethnic diversity precepts, community and creativity sense, and a citizen-level awareness.
- *Smart Governance*—any smart city should be governed with transparent, public decisions and have public and social services.

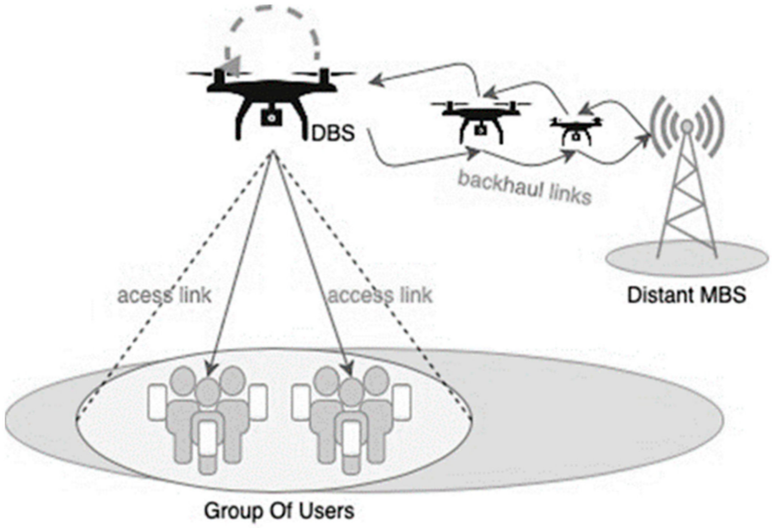


Figure 6. An example of a DBS application.

To achieve these, various technologies have to interoperate simultaneously and share information to process the state of the various environments and, finally, make decisions. However, this interoperability can be a difficult challenge to overcome and that is where IoT and IoV come into place. They guarantee interoperability in their networks, as well as offer properties such as security, scalability, and big data management, that are essential for a safe and reliable smart city.

Given the expected increase in connected devices in the near future, be it vehicles or non-mobile sensors and base stations, maintaining these wireless connections with low latencies and quality of service requirements is one of the major challenges for smart cities. The emerging 5G technologies and new computing paradigms (fog, edge, and cloud computing), as mentioned above, can prove to be a viable solution to these problems, but a change in the way we perceive the network topologies can also improve the outcome of these implementations for the better. Through the usage of UAVs as DBS, which can communicate with ground base stations (GBS), the coverage of networks substantially increased, allowing for more devices to be connected in a bigger area, enabling the communication between devices in different locations in the same city that otherwise wouldn't be able to and expanding the coverage of administrative services to the whole city. This is showcased in Figure 7, where DBS's are used with 5G technology to extend the coverage area of services for other vehicles and users.

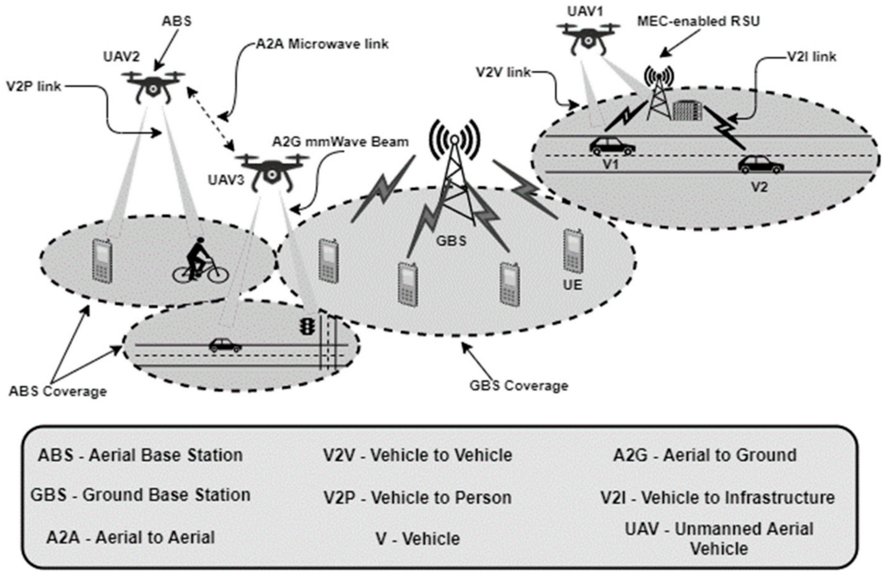


Figure 7. An example of the 5G-enabled drone architecture.

3. Blockchain

Blockchain is a technology deeply studied by researchers, given its distributed aspect that allows it to be implemented over P2P overlay networks, which are common nowadays, and the security benefits it brings.

3.1. *Definition*

Blockchain was firstly introduced in 2008 by Satoshi Nakamoto [6], as the technology behind Bitcoin, a virtual currency exchange system that uses cryptography (i.e., digital signatures) to avoid the implementation of trust-based models and, consequently, the requirement of a trusted third-party participant to validate transactions. Blockchain is described as a distributed ledger/database of transactions that can assure various security properties by using consensus protocols and cryptography algorithms [42].

The ledger takes the form of a distributed chain of blocks [43], as represented in Figure 8, where every block has the following:

- its hash,
- the hash of the previously added block,
- a timestamp,
- a nonce (for calculations),
- the number of transactions,
- a Merkel tree of transactions.

The ledger is distributed because every node in the network knows the chain of transactions, making them public for all, removing the single point of failure problem, affecting trust-based models with third-party authenticators. Before being added to the chain, a new block must be validated by the network participants through the usage of distributed consensus protocols. This, with the aid of cryptography algorithms and techniques, helps assure data integrity and traceability once a block enters the Blockchain while solving the problem of double-spending. The usage of consensus protocols removes the need for a third-party authority in transactions, which would reduce the system's throughput and makes these transactions automatic.

3.1.1. Merkel Tree

In the Merkel tree, each block has the root hash, which is the hash of all transactions, and each consequent non-leaf node is a hash of the concatenation of the two values below, forming a binary tree of hashed transactions. Given that transactions are hashed, any attempt at altering with transactions that have already been executed will result in different resulting hashes for the Merkel tree, allowing for easy detection of tampered transactions.

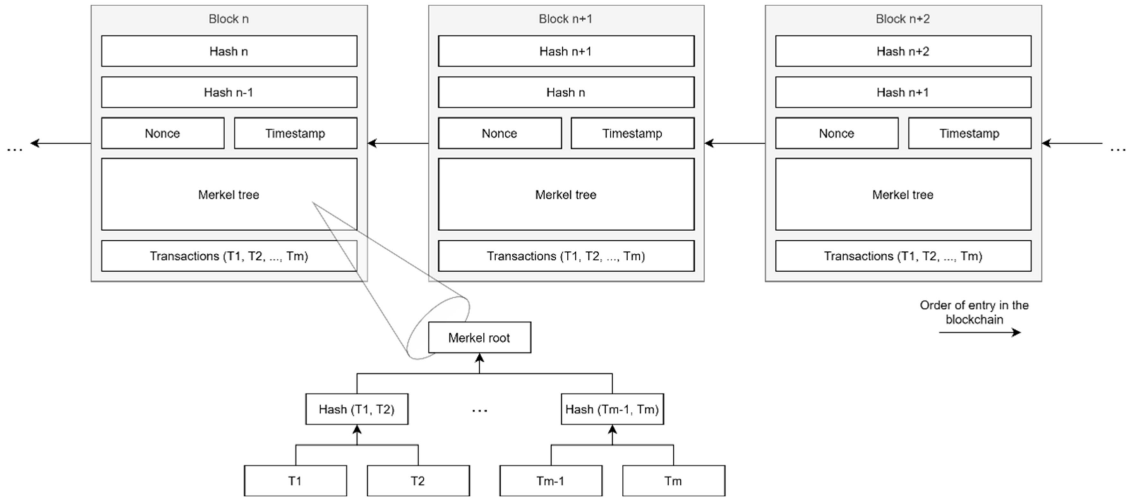


Figure 8. Blockchain structure example with three blocks.

3.1.2. Chain Forks/Discrepancy

On somewhat rare occasions, two blocks might be appended to the Blockchain at the same time, pointing to the same parent block. Blockchain systems rely on consensus mechanisms to solve these forks [7]. For example, in systems that use the *proof-of-work* consensus protocol, the longest chain is the one picked as it signifies a bigger investment in effort [6]. In *proof-of-stake*, the chain with the most total consumed coin age is chosen [44].

3.2. Technologies

Blockchain utilizes various technologies to achieve its famous security and privacy properties, being consensus protocols and cryptography techniques the main two forces behind these strengths.

3.2.1. Cryptography

Each user has two keys: a public and a private one. Asymmetric encryption is used for communications, where the public and private keys are utilized for encrypting and decrypting transactions, respectively [45,46]. The private key is also used to sign transactions before sending, and the receivers can verify this signature by using the sender's public key. This verification is made by every node that receives the transaction, which then disseminates it further. When the nodes that take part in the transaction verify and accept it, it is validated and placed on a timestamped block with the previous block's required hash. This block is then broadcasted back to the network, verified by the nodes, and then added to the Blockchain.

3.2.2. Consensus Protocols

Blockchain utilizes distributed consensus protocols to validate transaction blocks before they are inserted into the Blockchain, which ensures data immutability. More than one consensus protocol can be implemented together to fulfill application requirements.

Some examples of popular consensus protocols, and their definition, are:

Proof-of-Work

The pioneer consensus protocol for Blockchain was *proof-of-work* [6]. In PoW, nodes are tasked to solve arduous mathematical puzzles, a task also called "mining". Once a nonce is found that gives the block's hash a given format (usually hash has to start with *n* zero value bits), the node that found it broadcasts the solution to the network. When at least 51% (i.e., the majority) of the nodes verify it, it can be added to the Blockchain. PoW computation costs, especially at the energy level, can difficult the implementation in new systems that rely on lightweight nodes, such as IoV.

Proof-of-Stake

Proof-of-stake (PoS) [44] introduces a new "coin age" concept, defined as currency amount times holding time. This new value can then be used by the owner to "pay himself" through a special transaction called *coinstake*, which enables the generation of a block for the owner while consuming the specified coin age. The hash of the block generated needs to reach a certain hash target protocol, but in a limited search space (in contrast with PoW's unlimited search space), greatly reducing the energy consumption and, consequently, Blockchain's energy dependency. PoS also grants better block generation and transaction confirmation speeds, since only one block is created each cycle, with fewer nodes proposing blocks by round [47]. In Delegated PoS, a group of *witnesses* is voted by the nodes to generate blocks and are shuffled after a certain condition is met (be it time restrictions or blocks produced [48]), and are paid for each block generated. It could happen, however, that nodes with less coin age do not get a chance at generating a new block. Leased PoS is an enhance of PoS to solve this scarcity issue by allowing nodes to lease their coin age to other nodes with a higher chance of generating a block. When they do, the nodes that leased their coin age, receive a part of the reward for the generation [49].

Byzantine Fault Tolerance

Another common consensus protocol is *Byzantine Fault Tolerance*, where nodes vote for a majority decision. Various implementations of BFT have been developed, e.g., *Practical BFT* [50]. In these protocols, a proposer node, which changes in round-robin order, broadcasts a *pre-prepared* message that can only be accepted by the rest of the nodes if they have not accepted one already. If the majority does accept, the proposer sends a *prepared* message. Once other nodes are prepared, the proposer sends a *commit message* to all of them. Once the message is accepted, the state of the Blockchain is altered in each node. PBFT adds a timeout condition to tolerate faulty primary nodes (*proposer*) [51]. There is also a delegated BFT, where nodes are divided into two roles: *ordinary* and *bookkeepers*. Ordinary nodes do not take part in deciding consensus, only choosing which bookkeeper nodes they back. A random bookkeeper is selected, and it broadcasts its transaction data to the network and, when 66% of bookkeepers accept the data as valid, it is added to the Blockchain [52]. The majority of BFT protocols can handle 1/3 of faulty nodes.

Proof-of-Activity

Proof-of-activity, as described by Bentov et al. [53], is a consensus protocol that is a hybrid between PoW and PoS. The protocol starts by using complex mathematical tasks (the pseudorandom number and the finding of the *satoshi*, the creation of an empty block header, and the derivation of pseudorandom stakeholders from a hash). When nodes have enough stake, PoS algorithms start to occur (nodes with more stake have higher chances of being chosen). Forks in the chain are dealt with just like in PoW (i.e., the longest chain represents the biggest amount of effort), and the fees for transactions are divided by all found stakeholders.

Proof-of-Burn

It was presented as an alternative to PoW and PoS [54]. In this protocol, nodes are encouraged to spend their currency, and only then can they generate blocks and get the respective rewards. In Slimcoin [55], Proof-of-burn (PoB) is used as PoS, where the higher the amount of burned currency, the higher the chances of generating a block in the next round (instead of coin age). The burning of currency also keeps its value overall since there is less currency after the operation, making it rarer. The nodes burn the currency and receive a PoB, a string that proves that the burning took place, which is then used to receive a given reward [56].

Proof-of-Elapsed-Time

In this protocol, first proposed by Intel for its Sawtooth Lake platform, each node generates a random time value that has to follow a distribution set by the scheme. This time represents the waiting time before the block can generate blocks and can be updated every time the node generates a new block. This protocol solves the energy dependency problem of PoW and the "one CPU one vote" problem presented by Satoshi Nakamoto [57].

Proof of Capacity

This protocol is similar to PoS, where the "stake" is the capacity (or storage space) of the hard drive of nodes. The bigger this value, the higher the chances of nodes get to generate blocks [58].

Proof of Authority

Originally planned for Ethereum based private networks, Proof-of-Authority (PoA) has a set number of nodes (called *authorities*) that achieve consensus between them to generate blocks requested by *clients*. Every time a "step" (i.e., the unit that demarks time in PoA) passes, a new leader is elected from the authorities [59]. This protocol works better in private and consortium type Blockchains where there is a known set number of nodes.

Proof of Importance

In this protocol, the nodes have a ranking that grows with each successful validation of blocks and transactions made. Nodes with higher rankings have higher chances of generating blocks and, as such, the network itself has a higher trust value between nodes [60]. It differentiates itself from PoW and PoS because it allows smaller nodes, with fewer stakes or CPU power, also to participate in the network, making it so only participating nodes (the only ones beneficial to the network) get rewards.

Proof of Luck

In this protocol, a random number is assigned to each block (its *luck*). Every time a node wants to start to generate a block, it waits a set interval of time, receiving other blocks from other nodes and, if during this interval the node receives a luckier block, it substitutes its own (only if the parent block is the same). If, after that interval, its block is still the luckiest, the node broadcasts its block to the network and proceeds to generate the block header's hash for the next block [61]. In this protocol, forks are common, and, as such, nodes verify the total luck of each chain and choose the one with the best total, as it represents the chain with the closest desirable behavior.

Proof of Exercise

In the work of Shoker [62], it is proposed a protocol that is based on and tries to solve problems from PoW, such as Puzzle hardness, Block sensitivity, and Easy verification. In this protocol, nodes are challenged to solve computation-intensive problems (massive matrix operations) instead of solving hash-related tasks. This proves that the computing power of the Blockchain can also be used to solve real scientific problems, which opens doors for investigation.

3.2.3. Smart Contracts

Szabo [63] proposed the idea of smart contracts in the 1990s, defining them as a "computerized transaction protocol that executes the terms of a contract". In the context of Blockchain systems, smart contracts can be defined as small pieces of executable software programs that execute, automatically and independently, instructions when certain previously accorded conditions are met. These executions are accounted for as transactions for storing purposes, being inserted in a block, and added to Blockchain every time they are executed. Given the aforementioned properties of smart contracts, they offer appropriate access control and contract enforcement. It is also possible to conclude that these contracts are deterministic as every input returns a defined output, even if the contract is called repeated times.

The smart contract lifecycle is composed of four phases, depicted in Figure 9:

1. Creation:
   a. Negotiation of obligations, prohibitions, and rights.
   b. It is an iterative process with multiple rounds until an agreement is reached.
2. Deployment:
   a. Deployed to platforms on top of Blockchains.
   b. Digital assets of involving parties are locked.
   c. A new contract has to be created for emendations due to the immutability property of Blockchains.
3. Execution:
   a. Automatically executed when the previously negotiated conditions are met.
   b. Resulting transactions and updates are stored in the Blockchain.
4. Completion:
   a. Every transaction has been completed, and the currency has been transferred/removed to/from the involving wallets.
   b. The digital assets are unlocked and available for other transactions.

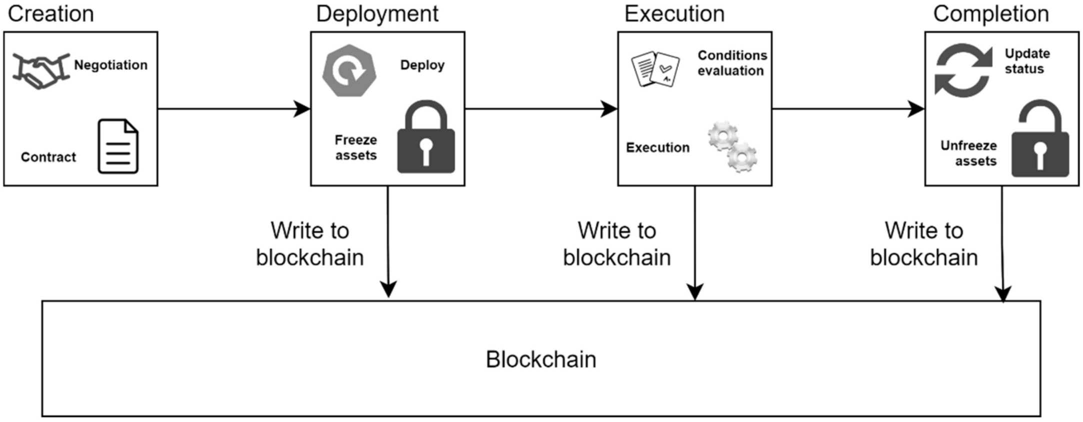


Figure 9. Smart contract lifecycle.

3.3. Benefits and Challenges

By converging all the technologies and paradigms mentioned above, Blockchain ensures various security properties and benefits for network nodes. However, it does not come without its challenges and obstacles.

Types of Blockchain

There are three types of Blockchains: public, private, and consortium Blockchains, which are a combination of the first two.

Public Blockchains are decentralized, immutable, preserve data non-repudiation (see Section 2.4.7), transparency, and traceability of transactions but have low scalability and flexibility. They do not require permission for access (*permissionless*), which means any node can enter and leave the network at any time.

Private Blockchains are the opposite: centralized, mutable, might not preserve data non-repudiation, transparency, or traceability of transactions, but have high scalability and flexibility. They require permissions before accessing the Blockchain (*permissioned*), which means nodes require permissions to access the network, call functions, or even see data.

In the middle of both, there are Consortium Blockchains. These Blockchains are partially decentralized, partially immutable, partially refusable, and partially traceable while achieving somewhat good scalability and flexibility. They are permissioned as well.

The usage of these three types of Blockchain depends on where and how they are deployed and implemented in the environment. For example, usually, private Blockchains are used in private networks.

Given the information above, the following is deductible: Blockchain assures *Data authentication* since every user has a pair of keys and their public key is identifiable. As such, it also ensures the property of *Data non-repudiation*, because since data is hashed with the user's keys, the data is connected to that user, and its actions cannot be denied. Each private and public key is unique, and since they are used to sign block hashes and transaction records, *Impersonation* becomes almost impossible. *Privacy* is maintained, but not completely, by making Blockchain addresses anonymous. However, the key management authority will always know what vehicle a key belongs to. *Data integrity* is maintained since data blocks have hashes that are calculated once. If the data is tampered with, it will generate a new, different hash, which would be detected rapidly, as mentioned above. The usage of cryptography ensures the properties of *Data confidentiality* and *Access control* in exchanging messages and smart contracts. Finally, Blockchain can also assure *Availability*, by allowing communication in heterogeneous environments since it can be deployed in overlay P2P networks. These properties of the Blockchain cover most of the security requirements of IoV systems mentioned in Table 2.

Table 2. Summary of comparison between types of Blockchain.

Property	Type of Blockchain		
	Public	**Private**	**Consortium**
Centralization	Decentralized	Centralized	Partially decentralized
Transparency	Transparent	Nontransparent	Partially transparent
Traceability	Traceable	Traceable	Partially traceable
Mutability	Immutable	Mutable	Partially immutable
Data Repudiation	Non-refusable	Refusable	Partially refusable
Scalability	Low	High	Good
Flexibility	Low	High	Good
Permission	Permissionless	Permissioned	Permissioned

While having many benefits, typical Blockchains suffer from low throughput, specially PoW-based Blockchains like Ethereum [64]. The cryptographic functions can be way too computationally and energy-intensive for the resource-constrained devices that are now proliferating in new network paradigms such as IoV, especially when using consensus protocols like PoW. There is also the debate between assuring complete privacy and anonymity or having data traceability, since achieving complete privacy and anonymity does not allow the Certificate Authority (CA) (the entity that assigns keys to nodes) to trace malicious transactions to the dishonest nodes in the network. A Public Key Infrastructure (PKI) is used by CAs to achieve privacy while guaranteeing traceability, making it so there is no way that attackers can link public keys to real identities [7].

3.4. Applications

Many works [65–67] have been done that expose how Blockchain has been implemented in growing systems to assure the security and privacy needs of applications for various purposes.

Some areas where Blockchain applications have been developed or show promising future paths are explored below.

3.4.1. Finance

Blockchain started by being used for cryptocurrency exchanges over networks ever since its genesis in 2008. New cryptocurrencies are still being developed, but there is also a research investment for their application in banking and financial development. Nguyen [68] stated that Blockchain has the potential for sustainable development of the economy and financial growth.

3.4.2. Healthcare

Healthcare systems are full of very different technologies that have to work together to deliver the best service to users. As such, Blockchain is a great technology for guaranteeing interoperability in these highly heterogeneous systems [69]. There is also a research path involving electronic health records (EHR) and their relevance on improving health services responses and, consequently, their quality [70].

3.4.3. Governance

Blockchain can improve the management of citizen records and certification. One of the major research paths is e-voting. With e-voting, everyone with access to technology would be able to vote and engage more easily with the political life of their surroundings, without worrying about transportation or wasting time in lines. *FollowMyVote* [71] is a proposed project for e-voting with end-2-end communication, based on BitShares. *BitCongress* [72], a discontinued project, used Counterparty and Ethereum smart contracts to manage voters and votes.

3.4.4. Business and Industry

In terms of Business and Industry, Blockchain brings promising implementations, mostly focusing on the problem of supply chain management and energy management. For the supply chain, Blockchain can serve to improve characteristics such as visibility, optimization, food safety, or even the automatization of transactions between intermediaries. *Coindesk* [73] is a project developed by IBM and Walmart to help manage China's pork meat market to improve safety in the supply chain. In the energy management path, Blockchain can allow for cost reduction and better planning and improve the energy market system for customers and providers [74].

3.4.5. Internet of Things

Blockchain has been adapted for various needs and functionalities in the Internet of Things. Recent studies utilize Blockchain as a middleware to ensure security and address these paradigms' security problems, creating what is referred to as the Blockchain of Things (BCoT) [43]. There's also research on Blockchain solutions for unmanned aerial vehicles, like drones [75].

3.4.6. Other Areas

Education, Security, Integrity Verification, and Data Management are also favorable research areas for Blockchain implementations. IoV is also a favorable research area for Blockchain implementations, as discussed in the following Section.

4. Blockchain-Based Solutions on the Internet of Vehicles

Given the characteristics of Blockchain and the security challenges of IoV, research and development for implementing the former in the latter have been growing over the years. Most of these implementations try to solve some security problems regarding data sharing in IoV networks.

In the work of Rathee et al. [76], a Blockchain-based framework to provide security, safety, and transparency for users, and vehicles, in cab-sharing scenarios (e.g., Uber) is proposed. In this framework, smart contracts are used, so information is not accessed and altered wrongly by unauthorized parties (i.e., secondhand data sharing). Vehicles and IoT devices need to be registered in the network to access its services via logging of information in a database and the Blockchain for traceability. IoT devices store their activities in the Blockchain, instead of vehicles, for ease detection of malicious behavior while maintaining a reasonable level of computational and storage costs.

For the Blockchain, managers are elected (a primary and a secondary, which takes over in the case of a faulty primary) to manage the Blockchain for some time and receive registration requests from other nodes, be them vehicles or singular devices. There are also miner and peer nodes, and the former helps validate the authentication of registration requests mentioned above. The managers then verify the requesters' authenticity and generate their public keys if the authenticity is verified. Implementing a rating-based system for providers allows users to evaluate which one will be chosen to give them a ride, providing a way to incentivize good behavior from providers. Providers can also choose which user to attend to based on multiple parameters. A Blockchain is maintained between the user and the provider to detect alterations of values (e.g., geographical position, time, route, among others).

The evaluation was made through a simulation in the NS2 network simulator with Blockchain techniques, aiming to get results under network congestion and compromised nodes, and comparing them with results from an already existing approach. An attacking scenario with an adversary model (i.e., malicious/hacked nodes present in the network) was utilized to test the proposed framework. Tests were made with malicious nodes passing as vehicles, IoT devices, and Blockchain nodes (miners and peers). The authors conclude that the proposed approach surpasses existing ones, with 86% success and accuracy rates, which are theorized to get higher with time due to the removal of ill-intended nodes. They also verify lower levels of network congestion with fake requests and that the possibility of attack with *n* compromised nodes was lower for the proposed approach. The altered amount of data, in the case of intruders messing with user's ratings, was also lower in the proposed approach. While still having some edges that need polishing, the proposed approach results achieved a 79% success rate compared with the already existing approach.

Wang et al. [77] presented a new scheme for vehicle registration and authentication in IoV based on Blockchain technologies, such as Ripple and BFT consensus protocols, and smart contracts, to eradicate the impact of malicious vehicles.

When a new node wants to join a contract group, it applies for it and is then evaluated by each node of the group through gathered data on that node in the infrastructure. If more than 51% of the nodes accept the node's integration, it is added to the group. Otherwise, it will be added to a watchlist of suspicious nodes and following applications to the contract group will be met with more restrictive conditions.

A distributed PKI system is utilized for trusted key distribution. For authentication, the vehicle sends a request to RSUs with their ID and public key. The RSU encrypts this data and sends it to the cloud service provider that, through the usage of consensus protocols, verifies the authenticity of the vehicle's identification.

The evaluation of this scheme was made using the Veins network simulator. The purpose was to simulate the average time and communication costs of the system. Veins provide a Mean operation that allows for checking time consumption in the various phases of authentication. The costliest phase was concluded to be the encryption in the key distribution center, with an average of 9 ms. The average time overhead for each layer was lower than 9 ms, which proves that the scheme can respond to requests promptly. The communication costs are, on average, 17 Kb for vehicle registration and 8 Kb for RSU verification, which is within the bounds of reasonability of technology for large-scale traffic networks.

Gao et al. [78] presented a system architecture for IoV where Blockchain, Software-Defined Networks (SDNs), and Fog computing. Blockchain solves security issues and settles trust between entities without a centralized trusted authority, as explained before. SDN technologies separate the control and data planes in a network to reduce structural complexity making nodes simply transmit data. At the same time, an SDN controller manages resource allocation, mobility, and rule generation. The fog computing approach helps reduce handovers in the network, which increases performance, and the RSUs and

OBUs (onboard units) that are present in these fog zones, are SDN-enabled which means the SDN controller also controls them. The authors also define RSUH (RSU hubs), which manages the control overhead between the controller and RSUs. A reputation-based trust system was implemented between vehicles to detect falsified information and help vehicles transmit only useful information while reducing the impact of attackers.

The evaluation was made with MATLAB combined with the NS-3 network simulator, using Hyperledger Fabric as the Blockchain platform, and the nodes were virtual machines deployed in Ubuntu containers. The two metrics that were taken into consideration for measurements were Packet Delivery Ratio and Transmission Delay. The former metric showed to be influenced by distance (optimal results were between 200 m and 500 m, due to route discovery and an increase in hops), the propagation model is chosen, the number of packets sent (i.e., more packets meant more collisions and a lower ratio), the number of vehicles and their speed (i.e., dense networks cause packet loss and a lower ratio). The Transmission Delay was influenced by distance (optimal results between 200 m and 500 m, due to MAC and the number of hops to the destination), the number of packets (i.e., the higher the number, the higher the contention and, consequently, the interference between nodes) and the density of vehicles in the network (i.e., more vehicles leads to more congestion). Due to these results, it was deemed to be a viable solution to Blockchain-enabled IoV communication and trust problems.

Kamal et al. [79] tackle the computation and energy power consumption in IoV during communications. They present changes to already existing protocols and software with no additional hardware requirements. The authors then propose low complexity solutions for operations like connectivity check, Blockchain development, and data provenance and forensics, including lightweight algorithms.

The experiments were performed with three MICAz motes [80], a wireless measurement system, each placed in a different car. The results obtained were then transferred to MATLAB to produce a simulation. The Received Signal Strength Indicator (RSSI) was measured to check the performance of the solution. Tests were made for Adversary Detection: first without an adversary (with and without a filter applied to the results) and then with an adversary (man-in-the-middle), and the Pearson Correlation Coefficient was calculated to check for value correlations (–1 for anticorrelation and 1 for high correlation). The filter test showed smoother results and value changes than the unfiltered one, with a high correlation in both cases (0.9754 and 0.9349, respectively). With the man-in-the-middle attack, the results were disparate and presented close to no correlation (i.e., 0.1282). Tests were also made for multimedia sharing, which proved to detect forged images by hash comparison. The Blockchain itself, due to its immutable property and how the block structure works, also serves to detect tampering and malicious activity to data (as explained before). Time complexity was calculated by increasing the link fingerprint's size (the binary version of the RSSI). From 30 to 3840 bytes, the time complexity was $O(1)$, the lowest in cryptographic-based solutions. After 3840 bytes of size, the complexity turned to $O(2^n)$, which would never happen in ideal situations. As such, it is concluded that data sharing and authentication can be achieved with lightweight encoding mechanisms and procedures.

Table 3 summarizes and compares Blockchain solutions based on their motivation, what is proposed, and the evaluation tools used.

Table 3. Comparison of Blockchain solutions.

Blockchain Solution	Motivation	Proposed Solution	Evaluation Tools
Rathee et al. [76]	Ensure security, safety, and transparency	Blockchain framework with smart contracts	NS-2, Blockchain platform not specified
Wang et al. [77]	Reduce impact of malicious nodes efficiently	Authentication and registration scheme	Veins
Gao et al. [78]	Increase performance with new paradigms	Blockchain-SDN-enabled solution with fog computing	MATLAB, NS-3, and Hyperledger Fabric
Kamal et al. [79]	Decrease power consumption	Various (lightweight algorithms, lesser complexity)	MICAz motes and MATLAB

5. UAV-Assisted IoV and Blockchain Solutions for Smart Cities

UAV-assisted IoV and Blockchain solutions for smart cities focus around various fields, including, but not limited to, coordinated UAV services, decentralized storage in UAV networks, or UAV networks for edge computing [81]. However, they can also work around other smart vehicles and their traffic control/fuel consumption [82].

5.1. Coordinated UAV Services

As UAVs' popularity rises, their application areas are also starting to expand. A group of UAVs can perform some tasks better compared to a single one. The range of the tasks varies, covering areas such as disaster recovery, surveillance, network relaying, energy-efficient device discovery, among others. The common element in these tasks is coordination among UAVs. However, such coordination needs improvements as it is based on communication between adjacent UAVs using some sort of broadcast/diffusion instead of a platform where can share common objectives and a certain level of knowledge to collaborate on this platform. Blockchain can be beneficial as it allows building a global channel for UAV communication.

There are more basic requirements for coordination between UAVs, such as preventing mid-air collisions, decision-making based on data collected from other UAVs, and secure communication. The latter could be ensured by Blockchain both in unicast and broadcast models and via the use of the public key cryptographic mechanisms. Furthermore, in such networks, a voting system scheme could be implemented, empowering UAVs to make many types of decisions, contextual choices on other UAV's views.

5.2. Decentralized Storage

Besides increasing network coverage, UAVs can execute data and computational offloading tasks. Due to the latter, they are considered vital technology of future IoT services provisioning, being used as dynamic platforms of storage, computation, and sensing in the air. Also, IoT advances in many industries, that is, agriculture, environmental monitoring, surveillance, logistics, health care, disaster monitoring, production process, are just examples of how wide it could be [83].

Vulnerabilities and more prominent exposure to security attacks result from such growth without proper security research to following along. Blockchain can then be used to secure distributed storage mechanisms because data needs to be assured to be legit, auditable, and safeguarded through proper cryptographic techniques.

Since offloading can be computational or of data, the use of decentralized UAV storage can ease the data carried by UAVs and end-users, therefore not overwhelming any of them, most of the time.

The crucial part of Blockchain use will be to secure data storage and forwarding from UAV nodes. Because this type of service is based on the opportunistic use of devices, Blockchain could implement a distributive reward network (such as cryptocurrencies). These rewards could be modeled as a mutual benefit earning, in which nodes will rationally tend to their maximum gains.

One example of published work on this topic is Lin et al. [84] work, where a resource trading system was implemented to help distribute computing resources concerning necessity in a smart city, assisted by IoV technology and Blockchain.

5.3. Blockchain-Based UAV Network for Mobile Edge Computing

Edge computing is a "counterpart" for the costs and issues of cloud computing. Physical proximity to devices can ensure lower latency and higher bandwidth. Likewise, it opens a new layer of privacy policy enforcement via the edge server before the data is released into the cloud service or storage.

Mobile Edge Computing (MEC) is similar but uses any sort of device that can be connected to 5G or other wireless networks and can be moved locally or geographically in a predetermined or even random manner.

UAVs seem uniquely tailored to such networks once they can use the air and not only a surface to move, and can carry out wireless connection, sensors, data, and more. They can be especially helpful in emergencies such as natural disasters, where the stationary or ground infrastructure is destroyed or incapable of responding. Therefore, in such situations, Blockchain can be used in MEC architectures if it can ensure that UAVs will trust each other, and it will be possible to build a flat architecture where other services can rely on with lesser concerns.

Lastly, Blockchain, along with smart contracts and transaction models, can be used to facilitate, but not only, the following: data deliverance, content sharing, and casting, and caching between the users connected to edge server nodes or caching servers (which can be a UAV network above in the air).

6. Tools

Various tools are used to simulate and evaluate Blockchain solutions in IoV networks. On the one hand, there are network simulators that allow for network construction and protocol integration based on development requirements. On the other hand, there are Blockchain platforms that provide support for Blockchain functions in the desired network environment. Most of the time, these technologies are used together to run a Blockchain system on nodes of a simulated network to get results that resemble more closely the results that would be obtained in real-life scenarios.

6.1. Network Simulators

Network simulators provide various properties and support for various types of networks. Given that the survey focuses on IoV, only simulators that support vehicular networks are taken into account.

6.1.1. MATLAB/Simulink

While MATLAB was not made to be a network simulator specifically, it is possible to simulate generic networks with its functionalities and Simulink. It has a VANET Toolbox [85] for vehicular network simulations, including support for V2V communication, but lacks development for V2I communications. MATLAB also offers 5G, LTE, and WLAN Toolboxes [86–88] for support of 5G, LTE, and WLAN technologies. There many other Toolboxes that serve other purposes or support other technologies, which make them powerful tools to help the simulation. The MATLAB community also has developed simulation software that is available for download. Even so, MATLAB is still used with other simulators to achieve better data visualization of simulation results.

6.1.2. NS-2

NS-2 [89] is a generic discrete-event network simulator that supports simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks, mobile network simulations with node movement, protocols, and traffic connection. NS-2 has an extensive manual with information on topics like Routing, Packet forwarding, Applications, and Transport in the network. It also provides help for installation and application. Nam (Network Animator) is a tool that can be used for network visualization (including network topology and packet-level animation) with data from NS-2. A negative side of NS-2 is that, as mentioned by the authors, it can have bugs that affect results (some unreliability).

6.1.3. NS-3

NS-3 [90] is a generic discrete-event network simulator that is well documented, easy to use, and debug. It caters to the entire simulation workflow's needs, from simulation configuration to trace collection and analysis. It supports real-time scheduling and network simulation, and as such, allows for real-world protocol implementation and simulation for IP and non-IP-based networks.

6.1.4. NCTUns

NCTUns [91] is a high-fidelity and extensible, generic network simulator supporting distributed simulation of large networks over multiple machines. It supports various protocols (e.g., WiMAX, TCP, UDP, 802.11p, among others) and types of networks, including support for ITS (i.e., V2V and V2I communications). NCTUns is an open-source and open system architecture, which enables the implementation of new protocols.

6.1.5. Veins

Veins [92] is a discrete-event vehicular network simulator based on OMNet++ [93], a discrete-event simulation platform, and SUMO [94], a mobility simulator. Veins is open source, which allows for customization of protocols by developers. It features support for various technologies according to IEEE standards, including 5G, model implementation from MATLAB, and human-driven/autonomous vehicle mobility simulation modes.

6.1.6. Summary

All the network simulators mentioned above support various network technologies and protocols that are valuable for research and development. While MATLAB is not a network simulator, serving best for data visualization and data management after the simulation is concluded, it can still be used as a generic network simulator with Simulink. MATLAB providing multiple pre-existing algorithms that can be deployed in the graphical environment of Simulink. It also allows for the input of datasets to fuel the simulations while providing the data visualization functions it is known for. Ultimately, MATLAB and Simulink can be used as a vehicular network simulator with the usage of the available toolboxes mentioned above that offer various technologies that are common on vehicular network simulations. These features distinguish the MATLAB+Simulink simulator from other generic and vehicular ones.

NS-2 and NS-3 share similarities, with NS-3 being a bit newer, while NS-2 has the downside of unreliability from bugs even though it does have more thorough documentation, but both are popular network simulators. NCTUns' author states that it was built with a new methodology (kernel re-entering), which gives it an advantage against other regular simulators (like NS-2), which eases the simulator consumption of CPU. Finally, Veins is specifically a vehicular network simulator, focusing solely on vehicular nodes and road infrastructure and technology, and this makes it the most recommended for simulating IoV networks since simulating node mobility in generic simulators, like the other mentioned above, can be troublesome.

6.2. Blockchain Platforms

Many Blockchain platforms exist on the market for various purposes and needs. Most of them are community-developed platforms and, hence, open-source.

6.2.1. Hyperledger

Hyperledger provides several Distributed Ledger Technologies (DLT) for various enterprise business applications. These technologies were built to be integrable with each other for mutual benefits.

Hyperledger Besu

Hyperledger Besu [95] is an Ethereum client made to interact with public Ethereum networks and private networks that run on an EVM. Even so, it has the proper permission schemes for consortium networks. Besu offers the possibility to utilize PoW, PoA, and IBFT consensus protocols, and it is programmed in Java. Through the integration of tools like Truffle, Remix, and web3j, Besu offers the means for smart contract and decentralized applications (dapps) development. However, Besu does not provide support for key management (creation, attribution, authentication), relying on a third-party key management tool, like EthSigner.

Hyperledger Burrow

Hyperledger Burrow [96] is a Blockchain node and smart contract execution engine that runs on an EVM and is optimized for public networks, but can also work in permissioned (private or consortium) networks. Burrow is developed in Go and offers consensus through BFT with the Tendermint algorithm. It was made to make Blockchain implementation simple, light, and fast. Burrow provides developers with bare-metal implementation instead of virtualization (containers). Smart contracts in Burrow are On-Chain and are written in the native language. It trades configurability and modularity with tight coupling of components, which removes the need to care about the underlying infrastructure.

Hyperledger Fabric

Hyperledger Fabric [97] is a framework, developed in Go, for private permissioned networks. It is highly modular, configurable, and pluggable, giving the developers the freedom to choose what technologies to implement and how to implement them (e.g., consensus protocols). Even so, generic Fabric works with PoS, but it can be changed to any intended one. Smart contracts are installed in the nodes and are developed in Golang before version 1.0, inclusive, or in Javascript from version 1.1 forward, and run in container environments. That said, a new approach to how peer nodes verify the execution of smart contracts removes the worry of non-determinism since deviant results are filtered before consensus, which allows Smart contracts to be written in standard programming languages and executed in a parallel fashion. Finally, Fabric does not require the implementation of a cryptocurrency for consensus.

Hyperledger Iroha

Hyperledger Iroha [98] is a simple Blockchain framework, written in C++, for private permissioned networks, that is made to help and ease application development with a variety of libraries, offering a lower complexity and easier management solution, with verifiable data consistency at low costs (with no mining, since it does not have a cryptocurrency) of Blockchain. It uses the high-performance YAC consensus protocol and provides built-in smart contracts (commands), which are On-Chain. Iroha presents robust permission and role-based access control over the nodes and the network.

Hyperledger Sawtooth

Hyperledger Sawtooth [99] is a highly modular and configurable Blockchain framework for private networks that offers an easy environment for application developers by allowing them to program in any language. Contract abstraction allows developers to write contract logic in the language of their choice through SDK transaction processors. Sawtooth offers Raft, PBFT, and PoET as options for consensus protocols, and a novelty is the fact that these consensus protocols can change during runtime to cater to the needs of the users. In Sawtooth's architecture, there is only one node type, which eases development, simplifying interactions.

6.2.2. Ethereum

Ethereum [100,101] is a simple, highly modular, and programmable Blockchain platform made for public permissionless networks. It introduced the Ethereum VM, a Turing-complete machine used by other Blockchain platforms (e.g., Hyperledger). It was intended for digital currency payments and transactions with the need for third-party authority, with its digital currency unit being called Ether. However, it can also be used for any other case of data exchange over a network. Ethereum works with PoW as a consensus protocol, but it is now changing to PoS due to PoW's high computational costs and energy consumption. It introduced fees or "gas" as a heuristic for miners to choose transactions from the transaction pool and users to prevent infinite-loop attacks. Smart contracts for Ethereum are mostly written in Solidity. Serpent and LLL can also be used for writing contract logic.

6.2.3. Corda

Corda [102] is an open-source Blockchain platform for private networks, developed in Java. Corda differs from other Blockchain applications in the fact that the nodes do not each holds a copy of a universal Blockchain, but only a ledger of the transactions in which they took part. Communications between nodes are point-to-point, and there are no message broadcasts about transactions. Each node in Corda is a Java VM environment with Corda services or CorDapps. Smart contracts are written in Java. Contracts can only verify internal validity, and Corda implements Oracles that verify external data, when required, for transactions and contracts. Since there is no broadcast, Notary clusters are used for uniqueness consensus to prevent double-spending of currency and validate transactions. Finally, consensus protocols for Corda are completely pluggable, giving the developers the freedom to choose the one that fits the best.

6.2.4. Tezos

Tezos [103] is an open-source platform for assets and dapps development in public networks. Tezos differentiates itself from other platforms by introducing self-amendment and on-chain governance, which allows for self-upgrades of the platform during runtime without heavy forks, which can lead to security leaks. Smart contracts in Tezos can be written in various high-level languages but are always compiled to Michelson, which helps reduce the risk of smart contract exploits. Finally, in Tezos, the consensus is achieved through a delegated PoS protocol.

Table 4 presents a comparison of Blockchain platforms based on network types, consensus protocols, programming language, and smart contract languages.

Table 4. Comparison of Blockchain platforms.

Blockchain Platform	Type of Networks	Consensus Protocol	Programming Language	Smart Contract Language
Hyperledger Besu	Public (Ethereum) and private networks	PoW, PoA, IBFT	Java	Truffle, Remix, web3j (Tools)
Hyperledger Burrow	Public (optimal), private, and consortium	BFT (Tendermint algorithm)	Go	Various
Hyperledger Fabric	Private	Various (pluggable)	Go	Golang (<1.0)/ Javascript (>1.1)
Hyperledger Iroha	Private	YAC	C++	Various
Hyperledger Sawtooth	Private	Raft, PBFT, PoET	Various	Various
Ethereum	Public	PoW, PoS (2.0)	Various	Solidity, Serpent, LLL
Corda	Private	Various (pluggable)	Java	Java
Tezos	Public	PoS (delegated)	Various	Various (Michelson)

7. Challenges and Open Issues

The convergence of the Blockchain and IoV technologies brings out many opportunities, but it is not without its challenges and obstacles. One of these challenges, and probably the one that sparked the most research, is dealing with the resource constraints of devices in IoV networks, which limit the implementation of traditional Blockchain systems [46] (such as the one implemented under Bitcoin) because not only it requires high computational power, that the devices in IoV do not hold, it also requires high energy consumption. This requirement is not favorable on systems like IoV, where devices try to extend their battery life to the maximum. Given that most resource consumption comes from the consensus algorithm used by the Blockchain approach, research focuses on developing new algorithms or system architectures to reach consensus in these distributed environments while utilizing fewer resources.

Mobility is also a great concern as frequent handovers can cause great delays in the network, which could decrease the efficiency, efficacy, and security of vehicles and applications, as decision-making results and notification alerts could be received with a delay that is not tolerable.

For the specific use of UAVs, due to their autonomous properties, regulations have to be created to mitigate the issues related to:

1. Real-life physics (aerodynamics, weather),
2. Technology development (sensors, actuators, types of engines, blades, wings),
3. Policy and laws (compliances, regulations, licensing),
4. Energy management (computing, engines versus battery),
5. Computing,
6. Standardization (protocols, services, networks, and data).

The advance/delay of one front influences another. Accidents related to other aircraft's physical safety, UAV crashes in residential areas and parks, misuse of footage breaking people's privacy are some examples of situations that have modified regional policies and have driven the creation of regulations for UAVs.

Smart cities are the hardest to implement and develop due to the complexity of the various technologies that intertwine to provide results, and due to the sheer scale, they have to work and deliver. The fact that it also depends on the development of emerging technologies (including, but not limited to, IoV and Blockchain) adding to the changes it requires on the city's physical distribution itself slows down the smart city concept development.

8. Future Perspectives

In the future, the development of these technologies can follow various directions. One of these paths is SIoV that applies social network algorithms and notions to IoV, focusing on the interactions between vehicles, devices, and infrastructure, to allow for more efficient communication strategies and AI implementation [28].

Another direction is the integration with SDNs, which reduce smaller devices' resource consumption, deviating responsibility to an SDN controller, for example [78]. This integration brings out benefits such as enhanced network and Blockchain management and improvements in terms of the lower complexity of smaller nodes in vehicles and other devices, with the removal of the control plane.

While reputation-based systems in IoV networks are usually used to incentivize data dissemination, like in Magaia et al. [104], where, with the help of machine learning and artificial immune systems, nodes in the network are incentivized to disseminate or cache data, these reputation-based systems can also serve other purposes. Dorri et al. [46] present a system architecture that substitutes the consensus algorithm of Blockchain altogether by utilizing a reputation-based system as a basis for their lightweight Blockchain-based solution for data sharing in the IoV, where each node stores a list of reputations of connected nodes with which they interact. These types of implementations can also be looked upon with great interest, as they prove to be viable takes on consensus and trust management in these environments.

Concerning smart cities, in the near future, smaller-scale implementations are likely to start being implemented, since they are less complex and easier to gather information, fix or exchange parts, for testing on different approaches. This would allow researchers to test various approaches on how to implement these smart systems and how to operate them efficiently. Some years later, with this knowledge at hand, a large-scale implementation in a large city might be possible, with the correct standards and regulations required.

9. Conclusions

In this article, we summarize the state-of-the-art for IoV and Blockchain technologies, including possible application areas and underlying technologies, with conclusions on why the latter's integration is so desired on the former due to the properties of both. Then, a listing of applications of Blockchain-based solutions in IoV is explored, where Blockchain is used to solve security problems in data sharing and/or storage in the context of IoV. Afterward, we review some of the most popular network simulators for IoV scenarios and some of the most popular Blockchain platforms available.

The Internet of Vehicles has a lot of potential in improving driving experiences and infrastructure in the foreseeable future. However, it also requires development in ensuring security for its users and service providers. Blockchain has been proven to be a viable solution to cater to the security needs of IoV systems. Even so, Blockchain still needs improvements to excel in these environments.

Smart cities open doors to great quality of life improvements in cities and resource management progress, but it is still far from having a technological basis strong enough to be rewarding due to their high costs, high technological requirements, which are still lacking, and a lack of standards.

Author Contributions: Conceptualization, P.Á., L.S., and N.M.; investigation, P.Á., L.S.; writing—original draft preparation, P.Á.; writing—review and editing, P.Á., L.S., and N.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by H2020-MSCA-RISE under grant No 101006411; and by Fundação para a Ciência e a Tecnologia (FCT) through the LASIGE Research Unit, ref. UIDB/00408/2020 and ref. UIDP/00408/2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Eze, E.C.; Zhang, S.; Liu, E. Vehicular ad hoc networks (VANETs): Current state, challenges, potentials and way forward. In Proceedings of the 2014 20th International Conference on Automation and Computing, Cranfield, UK, 12–13 September 2014. [CrossRef]
2. Singh, A.; Gaba, L.; Sharma, A. Internet of Vehicles: Proposed Architecture, Network Models, Open Issues and Challenges. In Proceedings of the 2019 Amity International Conference on Artificial Intelligence AICAI 2019, Dubai, United Arab Emirates, 4–6 February 2019; pp. 632–636. [CrossRef]
3. Benalia, E.; Bitam, S.; Mellouk, A. Data dissemination for Internet of vehicle based on 5G communications: A survey. *Trans. Emerg. Telecommun. Technol.* **2020**, *31*, e3881. [CrossRef]
4. Zhang, T. Cisco Confidential Challenges and Opportunities. 2015. Available online: https://site.ieee.org/denver-com/files/2016/02/IoV-Security-Challenges-and-Opportunities-zhang.pdf (accessed on 10 November 2020).
5. Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. Internet of Things for Smart Cities. *IEEE Internet Things J.* **2014**, *1*, 22–32. [CrossRef]
6. Nakamoto, S. *Bitcoin: A Peer-to-Peer Electronic Cash System*; Satoshi Nakamoto Institute, 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 18 November 2020).
7. Mollah, M.B.; Zhao, J.; Niyato, D.; Guan, Y.L.; Yuen, C.; Sun, S.; Lam, K.; Koh, L.H. Blockchain for the Internet of Vehicles towards Intelligent Transportation Systems: A Survey. *IEEE Internet Things J.* **2020**, *8*, 4157–4185. [CrossRef]
8. Skiba, D.J. The Internet of Things: A study in Hype, Reality, Disruption, and Growth. *Nurs. Educ. Perspect.* **2014**, *34*, 63–64. Available online: http://www.ncbi.nlm.nih.gov/pubmed/23586210 (accessed on 15 October 2020).
9. Davis, G. 2020: Life with 50 billion connected devices. In Proceedings of the 2018 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 12–14 January 2018. [CrossRef]
10. Kaiwartya, O.; Abdullah, A.H.; Cao, Y.; Altameem, A.; Prasad, M.; Lin, C.; Liu, X. Internet of Vehicles: Motivation, Layered Architecture, Network Model, Challenges, and Future Aspects. *IEEE Access* **2016**, *4*, 5356–5373. [CrossRef]
11. Contreras-Castillo, J.; Zeadally, S.; Guerrero-Ibanez, J.A. Internet of Vehicles: Architecture, Protocols, and Security. *IEEE Internet Things J.* **2018**, *5*, 3701–3709. [CrossRef]
12. Islam, A.; Hossan, M.T.; Jang, Y.M. Introduction of optical camera communication for Internet of vehicles (IoV). In Proceedings of the International Conference on Ubiquitous and Future Networks, ICUFN, Porto, Portugal, 29 June–2 July 2017; pp. 122–125. [CrossRef]
13. Yang, F.; Wang, S.; Li, J.; Liu, Z.; Sun, Q. An overview of Internet of Vehicles. *China Commun.* **2014**, *11*, 1–15. [CrossRef]

14. Meeting, W.G. White Paper of Internet of Vehicles (IoV). In Proceedings of the 50th Telecommunications and Information Working Group Meeting, Brisbane, Australia, 29 September–3 October 2014.
15. Nanjie, L. Internet of Vehicles: Your Next Connection. 2011. Available online: https://www.huawei.com/mediafiles/CORPORATE/PDF/Magazine/WinWin/HW_110848.pdf (accessed on 20 October 2020).
16. Bonomi, F. The Smart and Connected Vehicle and the Internet of Things. In *Invited Talk, Workshop on Synchronization in Telecommunication Systems*; 2013.
17. Feng, W.; Li, Y.; Jin, D.; Su, L.; Chen, S. Millimetre-wave backhaul for 5G networks: Challenges and solutions. *Sensors* **2016**, *16*, 892. [CrossRef] [PubMed]
18. Silva, L.; Magaia, N.; Sousa, B.; Kobusinska, A.; Casimiro, A.; Mavromoustakis, C.X.; Mastorakis, G.; Albuquerque, V.H. Computing Paradigms in Emerging Vehicular Environments: A Review. *IEEE CAA J. Autom. Sin.* **2021**, *8*, 491–511. [CrossRef]
19. Wu, W.; Yang, Z.; Li, K. Internet of Vehicles and applications. In *Internet of Things: Principles and Paradigms*; Morgan Kaufmann: Burlington, MA, USA, 2016.
20. Lee, C.H.; Huang, C.M.; Yang, C.C.; Wang, T.H. A cooperative video streaming system over the integrated cellular and DSRC networks. In Proceedings of the 2011 IEEE Vehicular Technology Conference (VTC Fall), San Francisco, CA, USA, 5–8 September 2011. [CrossRef]
21. Abdelsalam, M.; Bonny, T. IoV Road Safety: Vehicle Speed Limiting System. In Proceedings of the 2019 3rd International Conference on Communications, Signal Processing, and Their Applications, Sharjah, United Arab Emirates, 19–21 March 2019; pp. 1–6. [CrossRef]
22. Reichardt, D.; Miglietta, M.; Moretti, L.; Morsink, P.; Schulz, W. CarTALK 2000: Safe and comfortable driving based upon inter-vehicle-communication. In Proceedings of the Intelligent Vehicle Symposium, Versailles, France, 17–21 June 2002; pp. 545–550. [CrossRef]
23. Huang, R.H.; Chang, B.J.; Tsai, Y.L.; Liang, Y.H. Mobile Edge Computing-Based Vehicular Cloud of Cooperative Adaptive Driving for Platooning Autonomous Self Driving. In Proceedings of the 2017 IEEE 7th International Symposium on Cloud and Service Computing (SC2), Kanazawa, Japan, 22–25 November 2017; pp. 32–39. [CrossRef]
24. Kowshik, H.; Caveney, D.; Kumar, P.R. Provable systemwide safety in intelligent intersections. *IEEE Trans. Veh. Technol.* **2011**, *60*, 804–818. [CrossRef]
25. Wu, W.; Zhang, J.; Luo, A.; Cao, J. For Intersection Traffic Control. *IEEE Trans. Parallel Distrib. Syst.* **2015**, *26*, 65–74. Available online: http://www.scopus.com/inward/record.url?eid=2-s2.0-84919485495&partnerID=tZOtx3y1 (accessed on 16 October 2020). [CrossRef]
26. Chen, P.Y.; Guo, Y.M.; Chen, W.T. Fuel-saving navigation system in VANETs. In Proceedings of the 2010 IEEE 72nd Vehicular Technology Conference, Ottawa, ON, Canada, 6–9 September 2010. [CrossRef]
27. Collins, K.; Muntean, G.M. Route-based vehicular traffic management for wireless access in vehicular environments. In Proceedings of the 2008 IEEE 68th Vehicular Technology Conference, Calgary, AB, Canada, 21–24 September 2008; pp. 3–7. [CrossRef]
28. Alam, K.M.; Saini, M.; El Saddik, A. Toward social internet of vehicles: Concept, architecture, and applications. *IEEE Access* **2015**, *3*, 343–357. [CrossRef]
29. Butt, T.A.; Iqbal, R.; Shah, S.C.; Umar, T. Social Internet of Vehicles: Architecture and enabling technologies. *Comput. Electr. Eng.* **2018**, *69*, 68–84. [CrossRef]
30. Magaia, N.; Fonseca, R.; Muhammad, K.; Segundo, A.H.; Neto, A.V.; Albuquerque, V.H. Industrial Internet of Things Security enhanced with Deep Learning Approaches for Smart Cities. *IEEE Internet Things J.* **2020**, 1–14. [CrossRef]
31. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
32. Cheng, J.; Cheng, J.; Zhou, M.; Liu, F.; Gao, S.; Liu, C. Routing in internet of vehicles: A review. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2339–2352. [CrossRef]
33. Magaia, N.; Borrego, C.; Pereira, P.R.; Correia, M. EPRIVO: An enhanced PRIvacy-preserving opportunistic routing protocol for vehicular delay-tolerant networks. *IEEE Trans. Veh. Technol.* **2018**, *67*, 11154–11168. [CrossRef]
34. Mokhtar, B.; Azab, M. Survey on Security Issues in Vehicular Ad Hoc Networks. *Alex. Eng. J.* **2015**, *54*, 1115–1126. [CrossRef]
35. Sharma, S.; Kaushik, B. A survey on internet of vehicles: Applications, security issues & solutions. *Veh. Commun.* **2019**, *20*, 100182. [CrossRef]
36. Mehta, P.; Gupta, R.; Tanwar, S. Blockchain envisioned UAV networks: Challenges, solutions, and comparisons. *Comput. Commun.* **2020**, *151*, 518–538. [CrossRef]
37. Li, B.; Fei, Z.; Zhang, Y. UAV Communications for 5G and Beyond: Recent Advances and Future Trends. *IEEE Internet Things J.* **2019**, *6*, 2241–2263. [CrossRef]
38. Zeng, Y.; Wu, Q.; Zhang, R. Accessing From the Sky: A Tutorial on UAV Communications for 5G and Beyond. *Proc. IEEE* **2019**, *107*, 2327–2375. [CrossRef]
39. Feng, W.; Wang, J.; Chen, Y.; Wang, X.; Ge, N.; Lu, J. UAV-aided MIMO communications for 5g internet of things. *IEEE Internet Things J.* **2019**, *6*, 1731–1740. [CrossRef]
40. Shi, W.; Zhou, H.; Li, J.; Xu, W.; Zhang, N.; Shen, X. Drone Assisted Vehicular Networks: Architecture, Challenges and Opportunities. *IEEE Netw.* **2018**, *32*, 130–137. [CrossRef]

41. Founon, A.; Hayar, A. Evaluation of the concept of the smart city through local regulation and the importance of local initiative. In Proceedings of the 2018 IEEE International Smart Cities Conference (ISC2), Kansas City, MO, USA, 16–19 September 2018; pp. 1–6. [CrossRef]
42. Wang, H.; Zheng, Z.; Xie, S.; Dai, H.N.; Chen, X. Blockchain challenges and opportunities: A survey. *Int. J. Web Grid Serv.* **2018**, *14*, 352. [CrossRef]
43. Dai, H.N.; Zheng, Z.; Zhang, Y. Blockchain for Internet of Things: A Survey. *IEEE Internet Things J.* **2019**. [CrossRef]
44. King, S.; Nadal, S. Ppcoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. 2012. Available online: https://decred.org/research/king2012.pdf (accessed on 20 November 2020).
45. Ferrag, M.A.; Derdour, M.; Mukherjee, M.; Derhab, A.; Maglaras, L.; Janicke, H. Blockchain technologies for the internet of things: Research issues and challenges. *IEEE Internet Things J.* **2019**, *6*, 2188–2204. [CrossRef]
46. Dorri, A.; Steger, M.; Kanhere, S.S.; Jurdak, R. BlockChain: A Distributed Solution to Automotive Security and Privacy. *IEEE Commun. Mag.* **2017**, *55*, 119–125. [CrossRef]
47. Nguyen, C.T.; Hoang, D.T.; Nguyen, D.N.; Niyato, D.; Nguyen, H.T.; Dutkiewicz, E. Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities. *IEEE Access* **2019**, *7*, 85727–85745. [CrossRef]
48. Bach, L.M.; Mihaljevic, B.; Zagar, M. Comparative analysis of blockchain consensus algorithms. In Proceedings of the 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 21–25 May 2018. [CrossRef]
49. Leased Proof of Stake | Waves Documentation. Available online: https://docs.waves.tech/en/blockchain/leasing#leasing-benefits-for-the-node-owner (accessed on 11 November 2020).
50. Castro, M.; Liskov, B. Practical Byzantine Fault Tolerance and proactive recovery. *ACM Trans. Comput. Syst.* **2002**, *20*. [CrossRef]
51. Buchman, E. Tendermint: Byzantine Fault Tolerance in the Age of Blockchains. Ph.D. Thesis, The University of Guelph, Guelph, ON, Canada, 2016.
52. dBFT 2.0 Algorithm. Available online: https://docs.neo.org/docs/en-us/tooldev/consensus/consensus_algorithm.html (accessed on 11 November 2020).
53. Bentov, I.; Lee, C.; Mizrahi, A.; Rosenfeld, M. Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract]. *ACM Sigmetrics Perform. Eval. Rev.* **2014**, *42*, 34–37. [CrossRef]
54. Proof of Burn—Bitcoin Wiki. Available online: https://en.bitcoin.it/wiki/Proof_of_burn (accessed on 11 November 2020).
55. P4Titan. Slimcoin: A Peer-to-Peer Crypto-Currency with Proof-of-Burn. Whitepaper. 2014. Available online: https://slimcoin.info/whitepaperSLM.pdf (accessed on 20 November 2020).
56. Karantias, K.; Kiayias, A.; Zindros, D. Proof-of-Burn. In *Financial Cryptography and Data Security*; Springer: Cham, Switzerland, 2020. [CrossRef]
57. Chen, L.; Xu, L.; Shah, N.; Gao, Z.; Lu, Y.; Shi, W. On security analysis of proof-of-elapsed-time (PoET). In *Stabilization, Safety, and Security of Distributed Systems; Lecture Notes in Computer Science*; Springer: Cham, Switzerland, 2017; pp. 282–297. [CrossRef]
58. Tschorsch, F.; Scheuermann, B. Bitcoin and beyond: A technical survey on decentralized digital currencies. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 2084–2123. [CrossRef]
59. De Angelis, S.; Aniello, L.; Baldoni, R.; Lombardi, F.; Margheri, A.; Sassone, V. PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain. In Proceedings of the Italian Conference on Cyber Security, Milan, Italy, 6–9 February 2018; Volume 2058, pp. 1–11.
60. What Is POI | NEM Documentation. Available online: https://docs.nem.io/en/gen-info/what-is-poi (accessed on 12 November 2020).
61. Milutinovic, M.; He, W.; Wu, H.; Kanwal, M. Proof of luck: An efficient blockchain consensus protocol. *arXiv* **2017**, arXiv:1703.05435. [CrossRef]
62. Shoker, A. Sustainable blockchain through proof of exercise. In Proceedings of the 16th IEEE International Symposium on Network Computing and Applications NCA 2017, Cambridge, MA, USA, 30 October–1 November 2017; pp. 1–9. [CrossRef]
63. Christidis, K.; Devetsikiotis, M. Blockchains and Smart Contracts for the Internet of Things. *IEEE Access* **2016**, *4*, 2292–2303. [CrossRef]
64. Croman, K.; Decker, C.; Eyal, I.; Gencer, A.E.; Juels, A.; Kosba, A.; Miller, A.; Saxena, P.; Shi, E.; Sirer, E.G.; et al. On Scaling Decentralized Blockchains Initiative for CryptoCurrencies and Contracts (IC3). In Proceedings of the 3rd Workshop on Bitcoin and Blockchain Research, Accra Beach Hotel & Spa, Barbados, 16–26 February 2016; pp. 106–125. Available online: http://fc16.ifca.ai/bitcoin/papers/CDE+16.pdf (accessed on 15 November 2020).
65. Casino, F.; Dasaklis, T.K.; Patsakis, C. A systematic literature review of blockchain-based applications: Current status, classification and open issues. *Telemat. Inform.* **2019**, *36*, 55–81. [CrossRef]
66. Tama, B.A.; Kweka, B.J.; Park, Y.; Rhee, K.H. A critical review of blockchain and its current applications. In Proceedings of the ICECOS 2017—International Conference on Electrical Engineering and Computer Science, Sustaining the Cultural Heritage toward the Smart Environment for Better Future, Palembang, Indonesia, 22–23 August 2017; pp. 109–113. [CrossRef]
67. Di Francesco Maesa, D.; Mori, P. Blockchain 3.0 applications survey. *J. Parallel Distrib. Comput.* **2020**, *138*, 99–114. [CrossRef]
68. Nguyen, Q.K. Blockchain-A Financial Technology for Future Sustainable Development. In Proceedings of the 2016 3rd International Conference on Green Technology and Sustainable Development (GTSD), Kaohsiung, Taiwan, 24–25 November 2016. [CrossRef]

69. Mettler, M. Blockchain technology in healthcare: The revolution starts here. In Proceedings of the 2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom), Munich, Germany, 14–16 September 2016. [CrossRef]
70. Ekblaw, A.; Azaria, A.; Halamka, J.D.; Lippman, A.; Vieira, T. A Case Study for Blockchain in Healthcare: 'MedRec' prototype for electronic health records and medical research data. *Proc. IEEE Open Big Data Conf.* **2016**, *13*, 13.
71. The Secure Mobile Voting Platform of The Future—Follow My Vote. Available online: https://followmyvote.com/ (accessed on 16 November 2020).
72. Rockwell, M. BitCongress—Blockchain Based Voting System. Ethereum. 2014. Available online: https://forum.ethereum.org/discussion/110/bitc%20ongress-blockchain-based-voting-system (accessed on 25 November 2020).
73. Walmart Blockchain Pilot Aims to Make China's Pork Market Safer—CoinDesk. Available online: https://www.coindesk.com/walmart-blockchain-pilot-china-pork-market (accessed on 16 November 2020).
74. Mengelkamp, E.; Notheisen, B.; Beer, C.; Dauer, D.; Weinhardt, C. A blockchain-based smart grid: Towards sustainable local energy markets. *Comput. Sci. Res. Dev.* **2018**, *33*, 207–214. [CrossRef]
75. Han, T.; Ribeiro, I.; Magaia, N.; Preto, J.; Segundo, A.H.; Macêdo, A.R.; Muhammad, K.; Albuquerque, V.H. Emerging Drone Trends for Blockchain-Based 5G Networks: Open Issues and Future Perspectives. *IEEE Netw.* **2021**, *35*, 38–43. [CrossRef]
76. Rathee, G.; Sharma, A.; Iqbal, R.; Aloqaily, M.; Jaglan, N.; Kumar, R. A blockchain framework for securing connected and autonomous vehicles. *Sensors* **2019**, *19*, 3165. [CrossRef]
77. Wang, X.; Zeng, P.; Patterson, N.; Jiang, F.; Doss, R. An improved authentication scheme for internet of vehicles based on blockchain technology. *IEEE Access* **2019**, *7*, 45061–45072. [CrossRef]
78. Gao, J.; Agyekum, K.; Sifah, E.; Acheampong, K.; Xia, Q.; Du, X.; Guizani, M.; Xia, H. A Blockchain-SDN-Enabled Internet of Vehicles Environment for Fog Computing and 5G Networks. *IEEE Internet Things J.* **2020**, *7*, 4278–4291. [CrossRef]
79. Kamal, M.; Srivastava, G.; Tariq, M. Blockchain-Based Lightweight and Secured V2V Communication in the Internet of Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2020**, 1–8. [CrossRef]
80. Technology, C. MICAz: Wireless Measurement System. Prod. Datasheet 2008, 4–5. Available online: http://courses.ece.ubc.ca/494/files/MICAz_Datasheet.pdf (accessed on 27 November 2020).
81. Alladi, T.; Chamola, V.; Sahu, N.; Guizani, M. Applications of blockchain in unmanned aerial vehicles: A review. *Veh. Commun.* **2020**, *23*, 100249. [CrossRef]
82. Orecchini, F.; Santiangeli, A.; Zuccari, F.; Pieroni, A.; Suppa, T. Blockchain Technology in Smart City: A New Opportunity for Smart Environment and Smart Mobility. In *Intelligent Computing & Optimization*; Vasant, P., Zelinka, I., Weber, G.W., Eds.; ICO 2018; Advances in Intelligent Systems and Computing; Springer: Cham, Switzerland, 2018; Volume 866. [CrossRef]
83. Kota, S.; Giambene, G. Satellite 5G: IoT Use Case for Rural Areas Applications. In Proceedings of the SPACOMM 2019 International Conference on Advances in Satellite and Space Communications, Valencia, Spain, 24–28 March 2019; pp. 7–14.
84. Lin, X.; Wu, J.; Mumtaz, S.; Garg, S.; Li, J.; Guizani, M. Blockchain-based On-Demand Computing Resource Trading in IoV-Assisted Smart City. *IEEE Trans. Emerg. Top. Comput.* **2020**. [CrossRef]
85. VANET Toolbox: A Vehicular Network Simulator Based on DES—File Exchange—MATLAB Central. Available online: https://www.mathworks.com/matlabcentral/fileexchange/68437-vanet-toolbox-a-vehicular-network-simulator-based-on-des?s_tid=srchtitle (accessed on 19 November 2020).
86. 5G Toolbox Documentation. Available online: https://www.mathworks.com/help/5g/index.html (accessed on 19 November 2020).
87. LTE Toolbox Documentation. Available online: https://www.mathworks.com/help/lte/index.html (accessed on 19 November 2020).
88. WLAN Toolbox Documentation. Available online: https://www.mathworks.com/help/wlan/index.html (accessed on 18 November 2020).
89. The Network Simulator—ns-2. Available online: https://www.isi.edu/nsnam/ns/ (accessed on 18 November 2020).
90. ns-3 | A Discrete-Event Network Simulator for Internet Systems. Available online: https://www.nsnam.org/ (accessed on 18 November 2020).
91. EstiNet Network Simulator and Emulator (NCTUns). Available online: http://nsl.cs.nctu.edu.tw/NSL/nctuns.html (accessed on 18 November 2020).
92. Veins. Available online: https://veins.car2x.org/ (accessed on 18 November 2020).
93. OMNeT++ Discrete Event Simulator. Available online: https://omnetpp.org/ (accessed on 18 November 2020).
94. SUMO Documentation. Available online: https://sumo.dlr.de/docs/ (accessed on 18 November 2020).
95. Announcing Hyperledger Besu—Hyperledger. Available online: https://www.hyperledger.org/blog/2019/08/29/announcing-hyperledger-besu (accessed on 18 November 2020).
96. Hyperledger Burrow—Hyperledger. Available online: https://www.hyperledger.org/use/hyperledger-burrow (accessed on 18 November 2020).
97. Hyperledger Fabric—Hyperledger. Available online: https://www.hyperledger.org/use/fabric (accessed on 18 November 2020).
98. Hyperledger Iroha—Hyperledger. Available online: https://www.hyperledger.org/use/iroha (accessed on 18 November 2020).
99. Hyperledger Sawtooth—Hyperledger. Available online: https://www.hyperledger.org/use/sawtooth (accessed on 18 November 2020).
100. Ethereum Whitepaper | ethereum.org. Available online: https://ethereum.org/en/whitepaper/#ethereum (accessed on 18 November 2020).

101. How does Ethereum Work, Anyway? Introduction | by Preethi Kasireddy | Medium. Available online: https://medium.com/@preethikasireddy/how-does-ethereum-work-anyway-22d1df506369 (accessed on 18 November 2020).
102. Home | Corda Documentation. Available online: https://docs.corda.net/ (accessed on 18 November 2020).
103. Tezos | Get Started. Available online: https://tezos.com/get-started/ (accessed on 18 November 2020).
104. Magaia, N.; Sheng, Z. ReFIoV: A novel reputation framework for information-centric vehicular applications. *IEEE Trans. Veh. Technol.* **2019**, *68*, 1810–1823. [CrossRef]