TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 13/13 pp918–930
DOI: 10.26599/TST.2021.9010029
Volume 26, Number 6, December 2021

# Dataflow Management in the Internet of Things: Sensing, Control, and Security

Dawei Wei, Huansheng Ning*, Feifei Shi, Yueliang Wan, Jiabo Xu, Shunkun Yang, and Li Zhu

**Abstract:** The pervasiveness of the smart Internet of Things (IoTs) enables many electric sensors and devices to be connected and generates a large amount of dataflow. Compared with traditional big data, the streaming dataflow is faced with representative challenges, such as high speed, strong variability, rough continuity, and demanding timeliness, which pose severe tests of its efficient management. In this paper, we provide an overall review of IoT dataflow management. We first analyze the key challenges faced with IoT dataflow and initially overview the related techniques in dataflow management, spanning dataflow sensing, mining, control, security, privacy protection, etc. Then, we illustrate and compare representative tools or platforms for IoT dataflow management. In addition, promising application scenarios, such as smart cities, smart transportation, and smart manufacturing, are elaborated, which will provide significant guidance for further research. The management of IoT dataflow is also an important area, which merits in-depth discussions and further study.

**Key words:** Internet of Things (IoTs); dataflow; security; privacy; management

## 1 Introduction

Along with the overwhelming development of big data, Artificial Intelligence (AI), and 5G, the Internet of Things (IoTs) has attracted tremendous attention due to its influences on daily life and social developments. The International Data Corporation reports that by 2025, there will be almost 41.6 billion connected devices worldwide, with a growth rate near 28.7%[1]. The actual penetration and popularization of the IoT, as well as heterogeneous and various applications, indeed intensify undeniable opportunities and advances.

However, with the continuous increase in connected sensors and devices, the generated dataflows are also growing drastically, coupled with massive volumes, heterogeneous types, high speed, strong continuity, and other stringent features. Particularly in consideration of the intriguing promotion of the Internet of everything, physical sensors, mobile devices, smart objects, and other things are devoted to intelligent and seamless communication by transmitting dataflow from ends to ends. For the IoT, the simplest dataflow could be considered transmitting data from end sensors to the predefined cloud servers. The tremendous and steep increases in IoT dataflow offer an unimaginable potential for providing better IoT-based services, spanning smart homes, smart cities, smart transportation, intelligent

• Dawei Wei, Huansheng Ning, and Feifei Shi are with the School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China, and also with Beijing Engineering Research Center for Cyberspace Data Analysis and Applications, Beijing 100083, China. E-mail: weidaweiustb@163.com; ninghuansheng@ustb.edu.cn; shifeifeiustb@163.com.
• Yueliang Wan is with Beijing Engineering Research Center for Cyberspace Data Analysis and Applications, Beijing 100083, China, and also with Run Technologies Co., Ltd., Beijing 100192, China. E-mail: yueliang@bjrun.com.
• Jiabo Xu is with School of Information Engineering, Xinjiang Institute of Engineering, Urumqi 830023, China. E-mail: xujiabo_math@aliyun.com.
• Shunkun Yang is with School of Reliability and Systems Engineering, Beihang University, Beijing 100191, China. E-mail: ysk@buaa.edu.cn.
• Li Zhu is with Northern Cloud Research Team, Engineering University of PAP, Xi'an 710086, China E-mail: zhuli_cnrs@foxmail.com.
* To whom correspondence should be addressed.
Manuscript received: 2021-03-25; accepted: 2021-04-09

medicine, etc.

To date, intensive studies have been widely performed concerning data analysis, knowledge generation, content curation, and so forth. For example, in 2015, Ning et al.[2] proposed and organized a special issue discussing the analysis and management problems of big data in the IoT. Strohbach et al.[3] designed a framework for streaming data analysis in the IoT in addition to the module of batch processing, which enables the real-time response for the incoming dataflow. Similarly, Puthal et al.[4] designed an IoT architecture with dataflow in consideration of various security and safety issues. Mohammadi et al.[5] focused on the streaming data analysis in the IoT and made a comprehensive survey on deep learning, as well as its promising applications in the IoT. In recent years, the research of IoT dataflow management has been promoted well with great advances, although a comprehensive overview that surveys the research progress in this field and envisions future directions is lacking.

Compared with data analysis, the management of dataflow mainly focuses on an overall understanding and a holistic comprehension. Faced with overwhelming flows of data generated by considerably connected devices and sensors, it is of extreme significance to provide efficient regulation and oversight to fully exploit IoT resources and achieve the best values. To date, dataflow management mainly refers to monitoring and switching flows of data at high speed according to the predefined rules and requirements. For example, Singh et al.[6] discussed the dataflow management and compliance in cloud computing, under which data containment, access controls, and encryption were introduced, as well as the information control models. Carney et al.[7] introduced a new type of DataBase Management System (DBMS) named Aurora, which supports real-time stream monitoring and operations.

In this paper, we initially provide a systematic overview of IoT dataflow management, ranging from dataflow sensing, mining, control, security, and privacy protection. To further introduce the current status of IoT dataflow management, we introduce the platforms that have been applied to IoT dataflow management, and analyze the role of dataflow management in different IoT scenarios. Notably, the IoT dataflow discussed here refers to that generated by a complete IoT system or scenario composed of various sensors and devices, such as smart homes, smart transportation, and smart manufacturing. The main contributions of this paper are as follows:

- Introduce and analyze the key challenges faced by IoT dataflow to guide further research on dataflow management.
- Provide a comprehensive concept of IoT dataflow management systematically and conclude key techniques through the entire process, spanning dataflow sensing, mining, control, security, and privacy protection.
- Elaborate on and overview representative tools or platforms for IoT dataflow management to inspire future research and industrial developments.
- Analyze and envision typical application scenarios of IoT dataflow management, therefore demonstrating its promising possibilities in daily life and industrial manufacturing.

The remainder of this paper is arranged as follows: Section 2 concludes and analyzes the typical challenges of IoT dataflow. Section 3 provides a general concept of dataflow management and overviews key techniques related to dataflow management. Section 4 elaborates popular tools or platforms for IoT dataflow management. Section 5 analyzes the applications of dataflow management in various IoT scenarios. Section 6 gives the conclusion of this paper.

## 2 Representative Challenges of IoT Dataflow

In recent years, with the overwhelming increase in connected sensors and devices, especially coupled with the development of AI and 5G, there are more possibilities for flows of data to be generated and transmitted in the IoT, with large volumes and high speed, as well as demanding real-time requirements. Figure 1 shows that the annual size of the global datasphere has undergone a large increase, and it will be near 175 ZB in 2025[8]. In this section, we summarize and conclude representative challenges that are faced by IoT dataflow in the state of the art, and we hope to provide significant guidance for further research.

### 2.1 High speed

IoT dataflows are generated at high speed. In particular, with the invention of mobile and communication technologies, electric devices, such as smart sensors, phones, and tablets, are continuously generating streaming data at a higher speed[9]. For example, in smart traffic systems, various cameras are adopted for traffic monitoring. The higher the resolution of the camera, the more information that can be analyzed,

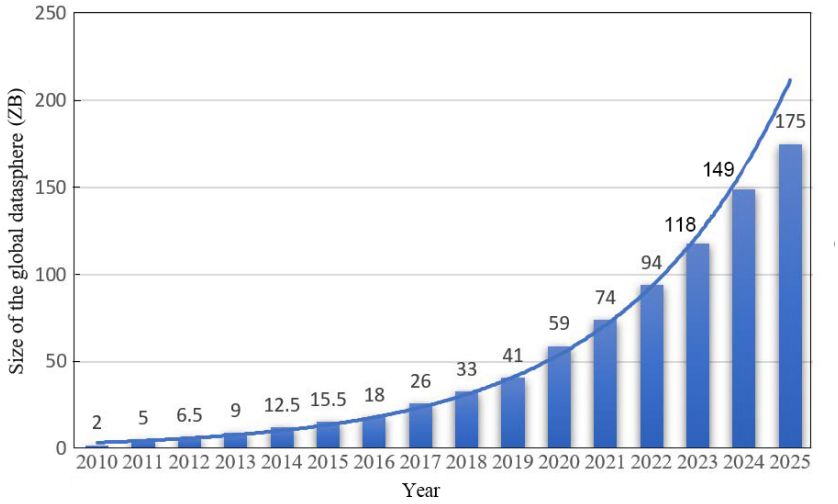


**Fig. 1 Annual size of the global datasphere from 2010 to 2025[8].**

which brings about a substantial increase in the demand for network bandwidth. Nevertheless, with technical innovation in production and manufacture, the speed for each sensor, as well as the scale of sensor-deployed systems, has been largely enhanced and improved, let alone for complicated devices, such as cameras and monitoring systems, which specialize in transmitting large-scale video streams.

The real-time generated data are expected to reflect practical situations more comprehensively; however, they also pose demanding requirements for the abilities of computing and storage. For example, for smart transportation systems, real-time monitoring of traffic flow and congestion is conducive to timely analysis and feedback of road conditions, so that citizens can make reasonable arrangements and intelligent decisions, while the high speed also poses many challenges for providing real-time process and response, such as in aspects of transmission, processing, and data security. Edge computing, fog computing, and other distributed IoT architectures are used to handle this challenge by processing the data closer to sensors to alleviate the pressure of data transmission and processing. However, this practice will lead to problems, such as increasing data security risks and a single processing model. Many studies have concentrated on real-time processes that aimed at high-speed dataflow in smart traffic. For instance, Rathore et al.[10] designed an architecture based on Giraph and Spark to achieve parallel analysis near real-time in smart traffic. Nallaperuma et al.[11] adopted online learning to handle the big arriving data with high speed, instead of using offline and incremental learning. The use of edge nodes is also a main method for handling high-speed dataflow[12, 13].

### 2.2 Strong variability

Strong variability describes the dynamic characteristics of streaming dataflow. For example, in-car identification of smart transportation systems, the dataflow will achieve its peak during rush hours while presenting a relatively stable state in other periods. In other words, data are generated and transmitted at different rates in the IoT and are deeply influenced and changed by internal and external factors. Different devices and sensors in given IoT scenarios may generate data at different rates and cannot achieve total consistency. In addition, the dataflow shows its dynamics according to practical situations.

Because the IoT dataflow is composed of data in motion from resources to destinations, the strong variability is a challenge that must be on everyone's mind in case of problems of imbalanced load. Understanding the dynamic variability of flows of data is of extreme significance for monitoring the real-time network status and realizing allocation and optimization of limited resources in time. For example, Santos et al.[14] presented a flow monitoring solution in IoT networks for better traffic supervision and analysis by considering factors, such as architectures of IoT networks and scalability of environmental surroundings, as well as limited resources.

### 2.3 Rough continuity

In contrast to traditional static data, IoT dataflow is equipped with general continuous features. The rough continuity we emphasize here is from a macro perspective and is much more suitable for complete IoT

scenarios and systems, instead of single device or sensor. Although some devices and sensors work in a triggered or intermittent manner, the total dataflow is continuous and unbounded, in which the old data are still being processed while incoming data arrive[15].

In this case, the continuously generated dataflow imposes higher requirements on the capabilities of sensing, transmitting, processing, and storing of IoT systems, and has become one of the paramount challenges. Considering the issues brought by continuous flows of data, Wickramaarachchi and Simmhan[16] proposed strategies for updating targeted applications with minimal disruption. Bhatnagar et al.[17] proposed a grid-based synopsis for establishing cluster schemes to help knowledge discovery from streaming dataflow.

### 2.4 Demanding timeliness

As for timeliness, it mainly comes from the demanding requirements of some applications that need a real-time response. For example, in smart traffic, it is sufficient to achieve the comprehensive monitoring of traffic to take appropriate actions in a very short time interval. Generally speaking, streaming dataflow is superior at its high timeliness, which could provide real-time analysis, while it also imposes demanding requirements in dataflow sensing, control, optimization, management, etc., to guarantee a timely response.

Presently, many studies consider demanding timeliness when handling streaming dataflow. For instance, Timely dataflow is a distributed computation model aimed at processing streaming dataflow in parallel, and it shows high performance in processing the real-time flows of data with low latency[18, 19]. Ellis[20] overviewed key techniques regarding analysis and visualization in streaming data, as well as the common architectures for processing real-time streaming dataflow. Given the demanding timeliness of streaming dataflow, the most fundamental procedure is to enhance the efficiency of data sensing, acquisition, distribution, analysis, storage, management, and other follow-up works to provide the expected feedbacks.

## 3 Key Techniques for IoT Dataflow Management

As Ning[21] once proposed that the IoT can be divided into unit IoT and ubiquitous IoT according to the applications that are provided. Unit IoT refers to the system that can provide specific applications, such as face recognition, intelligent monitoring, and smart control, which are very common in daily life and industrial manufacturing. An explicit characteristic of unit IoT is that most of them focus on a single scenario, under which the number and type of sensors and actuators, as well as processing and computing modes, are relatively limited. The ubiquitous IoT could be considered as a joint constitution of multiple units of IoT, for example, the complicated scenarios of smart cities, Internet of Vehicles, Industry 4.0, etc. Most of them are equipped with various sensors and devices, and could establish robust connections across different applications and domains, therefore providing efficient and diversified services.

Dataflow emerges from large-scale distributed IoT systems, and in turn, it contributes much to help improve the efficiency of heterogeneous and streaming flows of data. However, considering the challenges faced by IoT dataflow, strengthening and optimizing the management of IoT flows of data is very significant for providing guidance for better data mining and analyzing its potential values. As shown in Fig. 2, based on existing studies of dataflow, we analyze and conclude a general prototype of that in the IoT, in which key modules of data sensing, data mining, data control, and security and privacy are illustrated. Sensors collect the streaming data and transmit it to other devices with the necessary information of dataflow management in IoT systems. The dataflow is processed by processors, cloud servers, edge nodes, and gateways that meet the IoT system's dataflow requirements rather than being processed on specific devices. Finally, all information will provide services to users with external applications.

As mentioned above, the key modules of dataflow management are divided into sensing, mining, control, security, privacy protection, etc. We will discuss each aspect in the following sections.

### 3.1 Dataflow sensing

Generally speaking, for those unit IoT applications, only a few types of sensors would be used, and interactions between different applications or domains are rare. However, as IoT applications become increasingly intelligent, they provide richer services while imposing demanding requirements for more sensors, devices, applications, and domains to have interactions with. These sensors and smart gateways can sense and transmit data with different characteristics (smart cameras transmit streaming video data while smart switches only

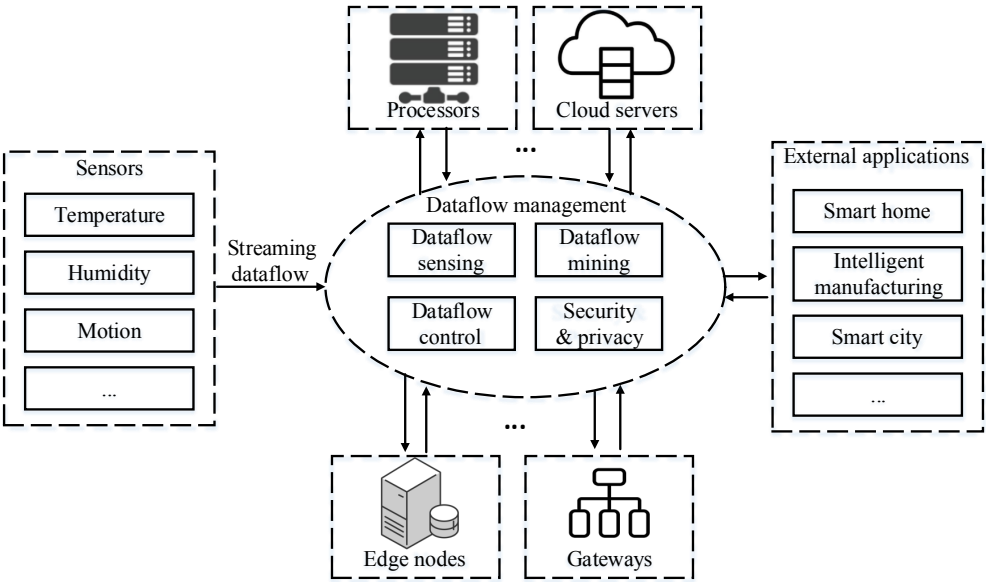


**Fig. 2 General prototype of dataflow in the IoT.**

transmit data when its state is changed). On this basis, many protocols have been developed for various IoT applications[22]. Most of them are developed for specific applications, and they are designed to meet different needs, such as safety and efficiency[23]. The protocols are not compatible with each other, such that the transmission of dataflow between different applications faces sensing problems. Therefore, accessing and understanding heterogeneous sensors in distributed IoT applications remain a prominent challenge.

The sensing of the IoT streaming data can be divided into two parts: the sensing between sensors and applications, and the sensing between IoT applications. The sensing between sensors and applications mainly focuses on the heterogeneous transmission protocols used by different IoT sensors, while many transmission protocols cannot directly access the Internet for data transmission. The main problem of data sensing between IoT applications is protocol fragmentation, mainly of the application layer protocols.

At present, the IoT is a heterogeneous system composed of different types of devices. Large differences between the physical layer and data link layer emerge when IoT dataflow is transmitted. Much dataflow cannot transmit through normal routers because protocols may not be recognized by the routers. In distributed IoT systems, edge nodes or fog nodes will retransmit the dataflow and help the applications sensing the dataflow[24, 25]. However, different frameworks target different IoT application environments, which means that different protocol architectures are only applicable to respective IoT applications. This situation further aggravates the fragmentation of the IoT protocols and complicates the sensing between applications.

As mentioned above, many IoT protocols aim to solve the dataflow interaction between IoT applications; however, none of them succeed while only increasing the fragmentation of IoT protocols. W3C started research on the Web of Things (WoTs) in 2015 and expected to solve the fragmentation of the IoT by web technologies and solve the problems of IoT dataflow sensing[26]. The things in the WoT are the abstractions of physical or virtual entities described by the standardized WoT Thing Description (TD). All the WoT things can be sensed by WoT TDs. Semantic technologies are one of the solutions for dataflow transmission between heterogeneous platforms in the WoT, using shared ontologies for IoT contexts, such as SOSA[27], data models (e.g., RDF), and query languages, including SPARQL[28] and XQuery[29]. In the WoT, many sensors that cannot support the WoT protocol must use intermediate objects to connect themselves to the WoT, which increases the security risks of dataflow. Semantic technology solves the problem of fragmentation of IoT protocols to a certain extent through a unified semantic framework. However, it is difficult to handle properly when facing the high-speed problem of IoT dataflow.

### 3.2 Dataflow mining

Providing personalized services to users according to the surrounding environment is the foundation of IoT systems. Because the original data of sensors are with

very low density, confusion, redundancy, and poor reliability, data mining is the main method for improving the value density of sensor data. In contrast to traditional IoT applications, which must collect all data before starting mining, the dataflow applications can analyze while sensing and collecting, which brings significant performance improvements[5].

Several scholars have studied the application of a variety of machine learning and deep learning algorithms in IoT dataflow mining[30, 31]. Current studies on dataflow mining are relatively mature[32]. In this paper, we only briefly elaborate two typical types of IoT dataflow mining. One is to divide the full complex mining work into several processing stages in different IoT devices, and the streaming data are processed through these devices to obtain the desired results[33]. This type of algorithm provides better privacy security at lower communication costs, but how to train the learning model for each node and make them work together effectively are urgent issues that need studying.

The other type of dataflow mining is to expand the dense dataflow processing to multiple nodes in the clusters. Different nodes mine the data to different degrees, or each node is responsible for part of the streaming dataflow to alleviate the total processing pressure. This approach takes more factors into considerations, such as scalability, fault tolerance, data communication and routing, resource allocation, and task scheduling.

### 3.3 Dataflow control

In IoT systems, particularly the distributed IoT systems, dataflow must be transferred and processed on different ends or devices that may belong to different users. This feature imposes demanding requirements on the dataflow control of access and usage. Usually, access control takes resources as the centric part and specifies whether data can be accessed by certain parties. However, we cannot specify the purpose of the data. In other words, data can be used by processing nodes in any way, which brings severe security problems. The IoT dataflow contains much personal, protected, and other sensitive information, thus data producers want special ways to control and adjust dataflow according to their environment.

Information Flow Control (IFC) is a standard method that is popular in the field of IoT dataflow control. The main idea of IFC is to separate the dataflow management by adding tags to the streaming data, which contains management policies, including who can read it, how to process it, and whom the data are transmitted to. However, designing a correct information management system under the guarantee of appropriate computing costs remains problematic.

Matos and Cederquist[34] presented a simple language-based framework for studying IFC in distributed security settings in the face of code mobility. It proposed the distributed non-interference property to ensure that information flowing in a program abides by the allowed flow policies of the domains where they originate. In Ref. [35], the capability-based access control model was proposed for IoT systems. It assigns each device a capability token that specifies some correct access rights.

Another widely used IoT dataflow control model is the Usage CONtrol ABC (UCONABC) model, which integrates Authorizations, oBligations, and Conditions. When the values of these attributes change, the permission will also change in real-time. This method can meet the requirements of dynamic dataflow control, based on which some scholars have made extensive improvements. For example, Refs. [36, 37] added time logic to the model for strategy analysis. The study in Ref. [38] added the hardware-based validation. In 2009, Harvan and Pretschner[39] studied a state representation algorithm. They used the dataflow mode to track and represent the state of sensitive data and its copies to provide support for realizing state-based dataflow control, such as that of the UCONABC model.

Moreover, Ref. [40] proposed a unified model for managing and analyzing heterogeneous data in the IoT, which has been verified in Snap4City Pilot Helsinki and Antwerp. Platform Nifi is used to manage IoT dataflow through a set of IoT agents to complete subscriptions to all IoT devices[41]. Pasquier et al.[42] used CamFlow, a cloud system that aims to implement end-to-end IFC as a dataflow management model for virtual machines in cloud services. It protects applications from interfering with each other, allows flexible dataflow interaction and sharing across applications, and prevents data abuse caused by configuration errors.

### 3.4 Security and privacy protection

The IoT systems contain significant amounts of private data and information, thus security and privacy protection issues are fundamental to ensure the normal operations of IoT systems. In the IoT dataflow systems, as dataflow is transmitted between different devices, the security protection must cover the entire process from

perspectives of sensing, transmitting, processing, and storage[43]. In this section, we mainly introduce solutions for security monitoring for IoT dataflow.

Recent research promotes the use of network-level solutions to detect and prevent attacks on intelligent household IoT devices by monitoring the network traffic in and out of IoT equipment. In Ref. [44], the authors demonstrated that stream-based monitoring can achieve most of the security benefits of dataflow and significantly reduce processing costs. At the same time, Ref. [45] developed a method for specifying the security policy of the IoT and then applied it to the network data plane traffic through a special intermediate box (called the M-box).

Nehme et al.[46] designed StreamShield to address the problems of security and privacy in IoT dataflow. They classified the security requirements in dataflow into two types, the data security punctuations and the query security punctuations. This stream-centric security model has advantages in aspects of flexibility, dynamicity, and the speed of enforcement.

## 4 Platform for the Management of IoT Dataflow

To better manage the streaming IoT dataflow, we need dedicated platforms that can efficiently address the challenges faced by flows of data and provide valuable functions, such as streaming processes, real-time analysis, visualization, storage, and query. To date, many researchers, institutions, and companies have been devoted to the development of specified tools or platforms, which slightly differ in the functions they focus on. In this section, we conclude popular platforms for continuous IoT dataflow management according to the platforms for general flows of data and for specified sensor data. In addition, a comparison is made by the data types they are mainly suitable for and the functions the platforms primarily concentrate on.

### 4.1 Platforms for general flows of data management in the IoT

Because streaming flows of data are full of potential value and need to be processed, analyzed, and managed well, leading companies, such as Apache, IBM, Microsoft, and so forth, have devoted themselves to the development of relatively integrated platforms to efficiently deliver potential services, such as financial analysis, network monitoring, and resource optimization. In this section, we overview platforms for general flows of data management in the IoT, which refers to the occasions in which platforms do not care too much about the types of dataflow, and the more significant element in the platform is the system architecture for managing demanding flows of data, including processing, analyzing, storing, visualizing, and so forth. In other words, the platforms concluded in this section are suitable for general dataflow in the IoT, whether they are in video, image, text, or sensor data, which would finally be converted to the forms that could be the right input.

The first category is for data streams with temporal characteristics, which refer to the streaming input with a representative time series or stamps. IBM Streams, as the name implies, is a specified platform for processing flows of data in motion[47, 48]. It can evaluate a wide range of data types, including video, image, text, even sensor data, and all data with time visibility. With operators, the platform can allow acquiring, analyzing, processing, and visualizing flows of data in time to help respond in time. In addition, IBM provides many employable toolkits, such as the object storage toolkit, which makes it possible to integrate IBM Streams with IBM Cloud Object Storage. Naiad is a data-parallel system built on the timely dataflow computation model. Generally, it is slightly difficult to perform complex processing of streaming data in a distributed system, such as multiple iterations or incremental calculations. By introducing the concept of timestamp, Naiad gives a very low-level model that can be used to describe arbitrarily complex streaming calculations[18]. We believe that the most confusing shortcoming of Naiad is that the interface is too abstract and difficult to understand and prompt. Furthermore, Amazon announced a serverless service named Amazon Timestream in 2018, which can provide a 100-fold faster query compared with the traditional relational databases[49]. Compared with the two platforms mentioned above, Amazon Timestream mainly focuses on the fast query and efficient storage, as well as simple trend prediction and analysis.

Apart from the abovementioned platforms, some platforms are designed for event streaming. For example, the Simple Scalable Streaming System (S4) announced by Yohoo! is a distributed stream processing engine whose input is a sequence flow of events[50]. Compared with the batch processing of stored big data, S4 is oriented to streaming and real-time processing, with rare manual intervention. However, S4 shows weak

reliability and may not be suitable for occasions that have demanding requirements for each data item. Another popular platform for the event streaming process is Apache Kafka, which supports processing, analyzing, storing, and managing streams of events[51]. It has been widely applied in areas, such as smart transportation systems, network monitoring, video processing, and so forth[52–55]. Nevertheless, the accuracy of data is slightly affected by the possibility of data being sent repeatedly.

Furthermore, we will discuss two more platforms with no explicit temporal features. One is Apache Storm, initially announced by Twitter and managed by the Apache community. It is a free and open-source distributed real-time computing system. By inputting an unbounded sequence of tuples, Storm can provide services, such as real-time data analysis, online learning, and continuous computing[56]. Another one is named STREAM project, a type of Data Stream Management System (DSMS) that is relatively similar to the DBMS and is aimed at managing the traditional static data[57]. DSMS offers efficient management for continuous data streams and can be easily extended with graphical interfaces to provide fast query and intuitive visualization.

### 4.2 Platforms for specified sensor data management in the IoT

Compared with the abovementioned platforms, there is also a specified type of dataflow management platform in the IoT for sensor data. Of course, sensors play significant roles in the IoT, which can sense and collect much time-series data in real-time and dynamically. Although the abovementioned platforms are also suitable for processing and managing sensor data, some researchers have devoted to the development of specified platforms to provide much more accurate and adaptive services. In this section, we present representative platforms for sensor data management and compare the functions they mainly focus on.

The WoTKit Platform released by Sense Tecnic Systems is one such service that enables processing multi-sensor data and responding in real-time[58, 59]. By introducing a dataflow program named pipe, the platform enables end users to find, control, visualize, query, and even store streaming sensor data. In 2017, Li et al.[60] proposed a Multi-sensor Data Real-time Monitoring Management System. It is designed for marine environments, where unmanned aerial vehicles equipped with multi-sensors are used for data acquisition, transmission, process, analysis, and management. The system demonstrates high feasibilities in certain flight experiments, while its robustness and compatibility remain uncertain. Considering the limitations of sensor data, Benabbas and Nicklas designed a Quality-aware Sensor Data Stream Management Tool for modeling sensor data semi-automatically and conducted processing and queries by pipelining[61]. Moreover, because smart sensors are heterogeneous in sources, types, data structures, etc., SStreaMWare serves as a middleware that allows various sensors to be represented in a unified data schema[62]. It finally provides an efficient management platform for sensor flows of data. Apart from SStreaMWare, similar proposals, such as Borealis, IrisNet, Hourglass, Fjords, and so forth, concentrate on the management of sensor dataflow in the IoT[63–66].

To elaborate on the state of the art of management platforms for IoT dataflow, we compare the above mentioned platforms. Table 1 lists the different data types and institutions the platforms belong to, as well as the functions they mainly provide. We hope it will provide valuable guidance for academia and industries.

## 5 Typical Application for IoT Dataflow Management

Dataflow management has been widely used in IoT applications because it provides better solutions for massive IoT dataflow with high speed, dynamicity, flexibility, etc. Especially in large-scale distributed IoT systems, the transmission of data flowing across various devices and applications need significant and efficient management. In this section, we analyze the typical applications of dataflow management in smart cities, smart transportation, and smart manufacturing scenarios.

### 5.1 Smart cities

Smart cities are the combination of cross-domain IoT applications to realize the intelligent, refined, and dynamic management of the city. It is important to establish a unified data center to gather flows of data from heterogeneous sensors and applications. By aggregating various dataflows, intelligent services can be generated, and predictions and recommendations can be provided to end users or decision makers in the smart city, such as for smart public security, smart emergency response, and smart government affairs[67].

Considering the different flows of data from various IoT devices, users, and applications, efficient

**Table 1 Comparison between different platforms for IoT dataflow management.**

| Platform | Data type | | | | Function | | | | Institution |
|---|---|---|---|---|---|---|---|---|---|
| | Video | Image | Text | Sensor data | Streaming process and real-time analysis | Visualization | Storage | Query | |
| IBM Streams[47, 48] | All kinds of data with real-time visibility | | | | ✓ | ✓ | ✓ | | IBM |
| Naiad[18] | Event streaming with timestamps | | | | ✓ | | | | Microsoft |
| Amazon Timestream[49] | Serverless time series | | | | ✓ | | ✓ | ✓ | Amazon |
| S4[50] | Event streaming | | | | ✓ | | | | Yohoo! |
| Apache Kafka[51] | Event streaming | | | | ✓ | | ✓ | | LinkedIn |
| Apache Storm[56] | Unbounded sequence of tuples | | | | ✓ | | | | Twitter |
| STREAM[57] | Continuous data streams | | | | ✓ | ✓ | | ✓ | Stanford University |
| WoTKit Platform[58, 59] | | | | ✓ | ✓ | ✓ | ✓ | ✓ | Sense Tecnic Systems |
| Multi-sensor Data Real-time Monitoring Management System[60] | | | | ✓ | ✓ | ✓ | ✓ | | Ocean University of China |
| Quality-aware Sensor Data Stream Management Tool[61] | | | | ✓ | ✓ | | ✓ | ✓ | University of Bamberg |
| SStreaMWare[62] | | | | ✓ | ✓ | ✓ | | ✓ | LIG Laboratory |

management is very significant, as it can provide specified access for data acquisition. The study in Ref. [40] designed an efficient management system in smart cities that can enable collecting, analyzing, and searching for large amounts of data and provide appropriate tools for data processing, visualizing, monitoring, etc. In addition, data-centric IoT dataflow management enables owners to track the flow and usage of data to effectively defend against external threats and destruction[68].

### 5.2 Smart transportation

Smart transportation collects real-time traffic information through numerous sensors in vehicles or surroundings to provide support for traffic monitoring, vehicle tracking, and travel planning. Effective dataflow management can help smart transportation systems handle real-time, high speed, and multiple types of data streams, and solve the problems of data loss, errors, and tampering caused by various emergencies in reality, which is the basis for an excellent smart transportation system.

RFID sensors, GPS on vehicles and mobile phones, video cameras, and other types of IoT sensors on the road collect and form the streaming dataflow of the smart transportation system altogether. The cameras produce continuous and high-speed dataflow, in which different operations, such as traffic statistics or vehicle identification, must be performed according to various requirements. GPS data collected from vehicles and mobile phones usually suffer from loss due to wireless communication problems; therefore the vehicle tracking system must work with other sensors to realize the tracking of vehicles. In addition, efficient management and monitoring of traffic have great significance, particularly when designing and optimizing personalized travel plans, in which real-time traffic dataflow management within a certain range counts considerably[69].

### 5.3 Smart manufacturing

For smart manufacturing, the smart machines in factories establish interactions with their surrounding environments. Ordinary machines are augmented with

intelligent abilities of self-perception, awareness, and autonomous learning, which enables them to process real-time data for self-diagnosis and prevent potential interruptions in the production process. In smart manufacturing, requirements for dataflow differ; for example, certain processes may generate data at high speed, such that efficient models are needed to process flows of data faster and more effectively.

Programming model prediction based on dataflow promotes service development and orchestration, and it is a representative method in data analysis for industrial manufacturing. In particular, many solutions must coordinate computing resources of the entire network; therefore, programming tools are usually more complicated because they ask developers to learn new protocols and APIs, and they create data processing components that are interconnected.

## 6 Conclusion

The pervasiveness of smart IoT enables many electric sensors and devices to be connected, at the same time, which generates a large amount of dataflow. Compared with traditional big data, the streaming dataflow presents representative challenges, such as high speed, strong variability, rough continuity, and demanding timeliness, which severely affect the management of IoT dataflow. In this paper, we emphasize the significance of efficient management for IoT dataflow. We first analyze the key challenges faced by IoT dataflow and initially overview the related techniques in dataflow management, spanning dataflow sensing, mining, control, security, privacy protection, etc. In addition, representative tools or platforms for IoT dataflow management are elaborated, which will provide significant guidance for further research. Moreover, typical applications for dataflow management in smart cities, smart transportation, and smart manufacturing are illustrated to demonstrate the promising significance for efficient IoT dataflow management. The management of IoT dataflow is an equally important area that merits in-depth discussions and further study.

### Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (No. 61872038).

### References

[1] B. Safaei, A. Mohammadsalehi, K. Talaei, S. Zarbaf, and A. Ejlali, Impacts of mobility models on RPL-based mobile IoT infrastructures: An evaluative comparison and survey, *IEEE Access*, doi: 10.1109/ACCESS.2020.3022793.

[2] H. S. Ning, D. G. Belanger, Y. L. Xia, V. Piuri, and A. Y. Zomaya, Guest editorial special issue on big data analytics and management in internet of things, *IEEE Internet of Things Journal*, vol. 2, no. 4, pp. 265–267, 2015.

[3] M. Strohbach, H. Ziekow, V. Gazis, and N. Akiva, Towards a big data analytics framework for IoT and smart city applications, in *Modeling and Processing for Next-Generation Big-Data Technologies*, F. Xhafa, L. Barolli, A. Barolli, and P. Papajorgji, Eds. Switzerland: Springer International Publishing, 2015, pp. 257–282.

[4] D. Puthal, R. Ranjan, S. Nepal, and J. J. Chen, IoT and big data: An architecture with data flow and security issues, in *Cloud Infrastructures, Services, and IoT Systems for Smart Cities*. Brindisi, Italy: Springer, 2017, pp. 243–252.

[5] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, Deep learning for IoT big data and streaming analytics: A survey, *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2923–2960, 2018.

[6] J. Singh, J. Powles, T. Pasquier, and J. Bacon, Data flow management and compliance in cloud computing, *IEEE Cloud Computing*, vol. 2, no. 4, pp. 24–32, 2015.

[7] D. Carney, U. Çetintemel, M. Cherniack, C. Convey, S. Lee, G. Seidman, M. Stonebraker, N. Tatbul, and S. Zdonik, Monitoring streams: A new class of data management applications, in *Proc. 28$^{th}$ Int. Conf. Very Large Data Bases*, San Francisco, CA, USA, 2002, pp. 215–226.

[8] John Rydning David Reinsel, John Gantz, Data age 2025: The evolution of data to life-critical don't focus on big data; focus on the data that's big IDC white paper, http://www.innovation4.cn/library/r21572, 2020.

[9] J. C. Yang, C. F. Ma, J. B. Man, H. F. Xu, G. Zheng, and H. B. Song, Cache-enabled in cooperative cognitive radio networks for transmission performance, *Tsinghua Science and Technology*, vol. 25, no. 1, pp. 1–11, 2020.

[10] M. M. Rathore, A. Ahmad, A. Paul, and G. Jeon, Efficient graph-oriented smart transportation using internet of things generated big data, in *Proc. 11$^{th}$ Int. Conf. Signal-Image Technology and Internet-Based Systems* (*SITIS*), Bangkok, Thailand, 2015, pp. 512–519.

[11] D. Nallaperuma, R. Nawaratne, T. Bandaragoda, A. Adikari, S. Nguyen, T. Kempitiya, D. De Silva, D. Alahakoon, and D. Pothuhera, Online incremental machine learning platform for big data-driven smart traffic management, *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 12, pp. 4679–4690, 2019.

[12] J. Li, M. Siddula, X. Z. Cheng, W. Cheng, Z. Tian, and Y. S. Li, Approximate data aggregation in sensor equipped IoT networks, *Tsinghua Science and Technology*, vol. 25, no. 1, pp. 44–55, 2020.

[13] D. Kim, J. Son, D. Seo, Y. Kim, H. Kim, and J. T. Seo, A novel transparent and auditable fog-assisted cloud storage with compensation mechanism, *Tsinghua Science and Technology*, vol. 25, no. 1, pp. 28–43, 2020.

[14] L. Santos, C. Rabadão, and R. Gonçalves, Flow monitoring system for IoT networks, in *New Knowledge in Information Systems and Technologies*, Á. Rocha, H. Adeli, L. P. Reis, and S. Costanzo, Eds. Switzerland: Springer, 2019, pp.

420–430.

[15] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy, Mining data streams: A review, *ACM SIGMOD Record*, vol. 34, no. 2, pp. 18–26, 2005.

[16] C. Wickramaarachchi and Y. Simmhan, Continuous dataflow update strategies for mission-critical applications, in *2013 IEEE 9th International Conference on e-Science*, Beijing, China, 2013, pp. 155–163.

[17] V. Bhatnagar, S. Kaur, and S. Chakravarthy, Clustering data streams using grid-based synopsis, *Knowledge and Information Systems*, vol. 41, no. 1, pp. 127–152, 2014.

[18] D. G. Murray, F. McSherry, R. Isaacs, M. Isard, P. Barham, and M. Abadi, Naiad: A timely dataflow system, in *Proc. $24^{th}$ ACM Symp. Operating Systems Principles*, Farminton, PA, USA, 2013, pp. 439–455.

[19] M. Sandstede, Online analysis of distributed dataflows with timely dataflow, arXiv preprint arXiv: 1912.09747, 2019.

[20] B. Ellis, *Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data.* Indianapolis, IN, USA: John Wiley & Sons, 2014.

[21] H. S. Ning, *Unit and Ubiquitous Internet of Things*. Boca Raton, FL, USA: CRC Press, 2013.

[22] List of automation protocols–Wikipedia, the free encyclopedia, https://en.wikipedia.org/wiki/List_of_automation_protocols, 2020.

[23] X. M. Zeng, X. S. Chen, G. L. Shao, T. He, and L. Wang, DTA-HOC: Online HTTPS traffic service identification using DNS in large-scale networks, *Tsinghua Science and Technology*, vol. 25, no. 2, pp. 239–254, 2020.

[24] N. K. Giang, M. Blackstock, R. Lea, and V. C. M. Leung, Developing IoT applications in the fog: A distributed dataflow approach, in *Proc. $5^{th}$ Int. Conf. Internet of Things* (*IoT*), Seoul, the Republic of Korea, 2015, pp. 155–162.

[25] Y. Teranishi, T. Kimata, H. Yamanaka, E. Kawai, and H. Harai, Dynamic data flow processing in edge computing environments, in *Proc. IEEE $41^{st}$ Annu. Computer Software and Applications Conf.* (*COMPSAC*), Turin, Italy, 2017, pp. 935–944.

[26] F. Paganelli, S. Turchi, and D. Giuli, A web of things framework for RESTful applications and its experimentation in a smart city, *IEEE Systems Journal*, vol. 10, no. 4, pp. 1412–1423, 2016.

[27] S. Sagar, M. Lefrançois, I. Rebaï, M. Khemaja, S. Garlatti, J. Feki, and L. Médini, Modeling smart sensors on top of SOSA/SSN and WoT TD with the semantic smart sensor network (S3N) modular ontology, in *ISWC 2018: $17^{th}$ Internal Semantic Web Conf.*, Monterey, CA, USA, 2018, pp. 163–177.

[28] L. Sciullo, C. Aguzzi, M. Di Felice, and T. S. Cinotti, WoT store: Enabling things and applications discovery for the W3C web of things, in *Proc. $16^{th}$ IEEE Annu. Consumer Communications and Networking Conf.* (*CCNC*), Las Vegas, NV, USA, 2019, pp. 1–8.

[29] R. De Virgilio and R. Torlone, A general methodology for context-aware data access, in *Proc. $4^{th}$ ACM Int. Workshop on Data Engineering for Wireless and Mobile Access*, Baltimore, MD, USA, 2005, pp. 9–15.

[30] A. Kos, S. Tomazic, J. Salom, N. Trifunovic, M. Valero, and V. Milutinovic, Big data processing: Data flow vs. control flow (new benchmarking methodology), in *Proc. 2014 Int. Conf. Identification, Information and Knowledge in the Internet of Things*, Beijing, China, 2014, pp. 56–59.

[31] Veeramanikandan, S. Sankaranarayanan, J. J. P. C. Rodrigues, V. Sugumaran, and S. Kozlov, Data flow and distributed deep neural network based low latency IoT-edge computation model for big data environment, *Engineering Applications of Artificial Intelligence*, vol. 94, p. 103785, 2020.

[32] W. S. Gan, J. C. W. Lin, H. C. Chao, and J. Zhan, Data mining in distributed environment: A survey, *WIREs Data Mining and Knowledge Discovery*, vol. 7, no. 6, p. e1216, 2017.

[33] S. Teerapittayanon, B. McDanel, and H. T. Kung, Distributed deep neural networks over the cloud, the edge and end devices, in *Proc. 2017 IEEE $37^{th}$ Int. Conf. Distributed Computing Systems* (*ICDCS*), Atlanta, GA, USA, 2017, pp. 328–339.

[34] A. A. Matos and J. Cederquist, Information flow in a distributed security setting, arXiv preprint arXiv: 1901.01111, 2019.

[35] S. Nakamura, T. Enokido, L. Barolli, and M. Takizawa, Capability-based information flow control model in the IoT, in *Innovative Mobile and Internet Services in Ubiquitous Computing*, L. Barolli, F. Xhafa, and O. K. Hussain, Eds. Switzerland: Springer, 2020, pp. 63–71.

[36] D. Basin, M. Harvan, F. Klaedtke, and E. Zălinescu, MONPOLY: Monitoring usage-control policies, in *Runtime Verification*, S. Khurshid and K. Sen, Eds. San Francisco, CA, USA: Springer, 2012, pp. 360–364.

[37] D. Basin, F. Klaedtke, and S. Müller, Policy monitoring in first-order temporal logic, in *Computer Aided Verification*, T. Touili, B. Cook, and P. Jackson, Eds. Edinburgh, UK: Springer, 2010, pp. 1–18.

[38] X. W. Zhang, J. P. Seifert, and R. Sandhu, Security enforcement model for distributed usage control, in *Proc. 2008 IEEE Int. Conf. Sensor Networks, Ubiquitous, and Trustworthy Computing*, Taichung, China, 2008, pp. 10–18.

[39] M. Harvan and A. Pretschner, State-based usage control enforcement with data flow tracking using system call interposition, in *Proc. 2009 $3^{rd}$ Int. Conf. Network and System Security*, Gold Coast, Australia, 2009, pp. 373–380.

[40] P. Bellini, F. Bugli, P. Nesi, G. Pantaleo, M. Paolucci, and I. Zaza, Data flow management and visual analytic for big data smart city/IoT, in *Proc. 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation* (*SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI*), Leicester, UK, 2019, pp. 1529–1536.

[41] Apache, Apache nifi, http://nifi.apache.org, 2020.

[42] T. F. J. M. Pasquier, J. Singh, D. Eyers, and J. Bacon, Camflow: Managed data-sharing for cloud services, *IEEE Transactions on Cloud Computing*, vol. 5, no. 3, pp.472–484, 2017.

[43] Y. Z. Wu, Y. Q. Lyu, and Y. C. Shi, Cloud storage security

assessment through equilibrium analysis, *Tsinghua Science and Technology*, vol. 24, no. 6, pp. 738–749, 2019.

[44] A. Sivanathan, D. Sherratt, H. H. Gharakheili, V. Sivaraman, and A. Vishwanath, Low-cost flow-based security solutions for smart-home IoT devices, in *Proc. 2016 IEEE Int. Conf. Advanced Networks and Telecommunications Systems* (*ANTS*), Bangalore, India, 2016, pp. 1–6.

[45] T. L. Yu, V. Sekar, S. Seshan, Y. Agarwal, and C. R. Xu, Handling a trillion (unfixable) flaws on a billion devices: Rethinking network security for the internet-of-things, in *Proc. 14$^{th}$ ACM Workshop on Hot Topics in Networks*, Philadelphia, PA, USA, 2015, pp. 1–7.

[46] R. V. Nehme, H. S. Lim, E. Bertino, and E. A. Rundensteiner, StreamShield: A stream-centric approach towards security and privacy in data stream environments, in *Proc. 2009 ACM SIGMOD Int. Conf. Management of Data*, Providence, Rhode Island, USA, 2009, pp. 1027–1030.

[47] IBM, IBM streams, https://www.ibm.com/cloud/streaming-analytics, 2020.

[48] M. Hirzel, H. Andrade, B. Gedik, G. Jacques-Silva, R. Khandekar, V. Kumar, M. Mendell, H. Nasgaard, S. Schneider, R. Soulé, et al., IBM streams processing language: Analyzing big data in motion, *IBM Journal of Research and Development*, vol. 57, nos. 3&4, pp. 7:1–7:11, 2013.

[49] Amazon Web Services, Amazon timestream, https://aws.amazon.com/timestream/, 2020.

[50] L. Neumeyer, B. Robbins, A. Nair, and A. Kesari, S4: Distributed stream computing platform, in *Proc. 2010 IEEE Int. Conf. Data Mining Workshops*, Sydney, Australia, 2010, pp. 170–177.

[51] N. Garg, Apache Kafka. Birmingham, UK: Packt Publishing, 2013.

[52] Å. Hugo, B. Morin, and K. Svantorp, Bridging MQTT and Kafka to support C-ITS: A feasibility study, in *Proc. 2020 21$^{st}$ IEEE Int. Conf. Mobile Data Management* (*MDM*), Versailles, France, 2020, pp. 371–376.

[53] R. Wiska, N. Habibie, A. Wibisono, W. S. Nugroho, and P. Mursanto, Big sensor-generated data streaming using Kafka and impala for data storage in wireless sensor network for $CO_2$ monitoring, in *Proc. 2016 Int. Workshop on Big Data and Information Security* (*IWBIS*), Jakarta, Indonesia, 2016, pp. 97–102.

[54] M. T. Tun, D. E. Nyaung, and M. P. Phyu, Performance evaluation of intrusion detection streaming transactions using apache Kafka and spark streaming, in *Proc. 2019 Int. Conf. Advanced Information Technologies* (*ICAIT*), Yangon, Myanmar, 2019, pp. 25–30.

[55] K. Yu, Y. Zhou, D. Li, Z. Zhang, and K. Q. Huang, A large-scale distributed video parsing and evaluation platform, in *Chinese Conference on Intelligent Visual Surveillance*, Z. Zhang and K. Huang, Eds. Beijing, China: Springer, 2016, pp. 37–43.

[56] A. Batyuk and V. Voityshyn, Apache storm based on topology for real-time processing of streaming data from social networks, in *Proc. 2016 IEEE 1$^{st}$ Int. Conf. Data Stream Mining and Processing* (*DSMP*), Lviv, Ukraine, 2016, pp. 345–349.

[57] A. Arasu, B. Babcock, S. Babu, J. Cieslewicz, M. Datar, K. Ito, R. Motwani, U. Srivastava, and J. Widom, Stream: The Stanford data stream management system, in *Data Stream Management*. Berlin, Germany: Springer, 2016, pp. 317–336.

[58] M. Blackstock and R. Lea, Toward a distributed data flow platform for the web of things (distributed node-RED), in *Proc. 5$^{th}$ Int. Workshop on Web of Things*, Cambridge, MA, USA, 2014, pp. 34–39.

[59] M. Blackstock and R. Lea, IoT mashups with the WoTKit, in *Proc. 2012 3$^{rd}$ IEEE Int. Conf. Internet of Things*, Wuxi, China, 2012, pp. 159–166.

[60] X. Li, Y. Han, F. J. Yu, and G. Chen, Multi-sensor data real-time monitoring and management system based on onboard UAV for ocean observation, in *Proc. 4$^{th}$ Int. Conf. Machinery, Materials and Information Technology Applications*, Xi'an, China, 2017, 174–181.

[61] A. Benabbas and D. Nicklas, Quality-aware sensor data stream management in a living lab environment, in *Proc. 2019 IEEE Int. Conf. Pervasive Computing and Communications Workshops* (*PerCom Workshops*), Kyoto, Japan, 2019, pp. 445–446.

[62] L. Gurgen, C. Roncancio, C. Labbé, A. Bottaro, and V. Olive, SstreaMWare: A service oriented middleware for heterogeneous sensor data management, in *Proc. 5$^{th}$ Int. Conf. Pervasive Services*, Sorrento, Italy, 2008, pp. 121–130.

[63] D. J. Abadi, Y. Ahmad, M. Balazinska, U. Çetintemel, M. Cherniack, J. H. Hwang, W. Lindner, A. S. Maskey, A. Rasin, E. Ryvkina, et al., The design of the borealis stream processing engine, in *Proc. 2005 CIDR Conf.*, Asilomar, CA, USA, 2005, pp. 277–289.

[64] P. B. Gibbons, B. Karp, Y. Ke, S. Nath, and S. Seshan, IrisNet: An architecture for a worldwide sensor web, *IEEE Pervasive Computing*, vol. 2, no. 4, pp. 22–33, 2003.

[65] J. Shneidman, P. Pietzuch, J. Ledlie, M. Roussopoulos, M. Seltzer, and M. Welsh, *Hourglass*: *An Infrastructure for Connecting Sensor Networks and Applications*. Columbia, MA, USA: Harvard University, 2004.

[66] S. Madden and M. J. Franklin, Fjording the stream: An architecture for queries over streaming sensor data, in *Proc. 18$^{th}$ Int. Conf. Data Engineering*, San Jose, CA, USA, 2002, pp. 555–566.

[67] R. Lea and M. Blackstock, City hub: A cloud-based IoT platform for smart cities, in *Proc. 2014 IEEE 6$^{th}$ Int. Conf. Cloud Computing Technology and Science*, Singapore, 2014, pp. 799–804.

[68] J. M. Bohli, A. Skarmeta, M. V. Moreno, D. García, and P. Langendorfer, SMARTIE project: Secure IoT data management for smart cities, in *Proc. 2015 Int. Conf. Recent Advances in Internet of Things* (*RIoT*) , Singapore, 2015, pp. 1–6.

[69] B. Jan, H. Farman, M. Khan, M. Talha, and I. U. Din, Designing a smart transportation system: An internet of things and big data approach, *IEEE Wireless Communications*, vol. 26, no. 4, pp. 73–79, 2019.

**Dawei Wei** received the BS degree from Shenyang University of Technology, China in 2015, and the MS degree from University of Science and Technology Beijing, China in 2008. He is currently pursuing the PhD degree at University of Science and Technology Beijing, China. His current research interests include IoT, protocol processing, machine learning, and optimization algorithm.

**Huansheng Ning** received the BS degree from Anhui University in 1996 and the PhD degree from Beihang University in 2001. He is currently a professor and vice dean at the School of Computer and Communication Engineering, University of Science and Technology Beijing, China, and the founder of Beijing Engineering Research Center for Cyberspace Data Analysis and Applications. He has published more than 100 journal/conference papers, and authored five books. He served as an associate editor of *IEEE Systems Journal* from 2013 to 2020 and was an associate editor of the *IEEE Internet of Things Journal* from 2014 to 2018. He was the steering committee member of the *IEEE Internet of Things Journal* from 2016 to 2020. His current research focuses on the IoT and general cyberspace.

**Feifei Shi** received the BS degree from China University of Petroleum in 2016 and the MS degree from University of Science and Technology Beijing in 2019. She is currently a PhD candidate at the School of Computer and Communication Engineering, University of Science and Technology Beijing. Her current research interests include IoT and artificial intelligence.

**Yueliang Wan** received the PhD degree from Beijing Institute of Technology, China in 2007. He is currently the director of the Research Institute with Run Technologies Company, Ltd., Beijing, China. He is also the founder of Beijing Engineering Research Center for Cyberspace Data Analysis and Applications. He focuses on the content security in Internet, multimedia analysis, and Internet search and mining. His research interests include Internet multimedia retrieval, privacy protection, and data center network.

**Jiabo Xu** received the BS degree from Yantai University, China in 2004, and the PhD degree from Xinjiang University, Urumqi, China in 2011. He is currently a professor and the dean of the School of Information Engineering, Xinjiang Institute of Engineering.

**Shunkun Yang** received the BS, MS, and PhD degrees from Beihang University, China in 2000, 2003, and 2011, respectively. He has been an associate research professor at Beihang University since 2016. He was also an associate research scientist at Columbia University from 2014 to 2015. His main research interests include reliability, testing and diagnosis for embedded software, CPS, IoT, Intelligent manufacturing, etc.

**Li Zhu** received the BEng degree in electronic information engineering from Beihang University, Beijing, China in 2011, and the PhD degree from Paul Sabatier University, Toulouse, France in 2018. He is currently working at Engineering University of the PAP, Xi'an, China. His current research interests include signal processing, control of robotics, and artificial intelligence.