Software Defined IoT Systems: Properties, State-of-the-Art, and Future Research

Pritish Mishra[1], Deepak Puthal[2], Mayank Tiwary[1], and Saraju P. Mohanty[3]
[1]SAP Lab, [2]Newcastle University, and [3]University of North Texas

***Abstract*—Internet of things (IoT) has revolutionized the modern way of life with the advent of *Intelligent* Systems. The traditional network architecture employed by the IoT domain is unable to define sufficient solutions for these challenges for a cost-effective and seamless workflow. The intelligent IoT system ensures the scalability in challenging or hostile environments. With the emergence of Software Defined Networking (SDN) domain offering programming ability of the control plane, many of these challenges seemed surmountable. This article presents a synergized overview of the challenges faced by the traditional domain and how they can be overcome by the upcoming domain of SDN-IoT. A thorough analysis of the practical adoptions showcase and feasibility of the solution in a real-time environment. The article examines the state of the art and highlights some of the key open-points in the domain, based on shortcomings of the current state of SDN-IoT, that can be taken up for future research.**

I. Introduction

The recent advancements made in the Internet of Things (IoT) domain paradigm are heralding remarkable transformations for improving the way of human life. The plethora of smart devices powered with their actuation and sensing abilities, environmental awareness and real-time analysis are making every aspect of modern society smarter and more efficient. For the purpose of achieving these goals, IoT devices must be able to inter-connect and concurrently provision services that are backed by the back-end data-storage systems, while processing and assimilating the huge data generated by the various sensors and actuators [1].

The fact that the IoT domain employs a huge number of devices that can collect information in real-time environment and on such a high frequency is perceived as its greatest strength. However, this also acts as a major bottleneck for the network. The complexity, heterogeneity and associated limitations of the various devices require complex and specific tools for management and for improving the performance of the network. The critical aspects associated with the performance and scaling of the network are often attributed to the characteristics of the devices and the proprietary architectures adopted by them. These issues cannot just be solved by the mere introduction of a "gateway" like structure, and then, the challenges also emerge in form of data-aggregation, reliability, privacy, security and trustworthiness. The capability and protocol mismatch between the IoT devices accentuates when there is a huge number of devices deployed in a complex environment with varying protocols and varying designs. These problems become even more acute in real-time scenarios like robotics and self-driven cars, where the exchange of real-time information also needs the applications to become scalable, efficient, seamless and cost-effective. Unfortunately, however, the current state of IoT technology alone cannot provision such requirements and overcome such challenges in an efficient manner [2].

Current state-of-the-art architecture for IoT devices aren't capable of supporting features like mobility, higher scalability and heavy traffic all at the same time along with the above mentioned functionalities. Moreover, with the number of connected IoT devices slated to grow exponentially over the next few years, it will become even more tedious to manage the mammoth amount of data generated by these devices, without the inherent features of elasticity and flexibility of the network. In absence of these features, such a flood of data and traffic could paralyze the entire network.

For overcoming such challenges faced by IoT paradigm, upcoming technologies like Software Defined Networking (SDN), Open-Flow architecture (based on SDN), and Network Function Virtualization (NFV) are gaining major traction. The technology domain of SDN, specifically, has been receiving a lot of attention from the research community and has also proven its mettle in large deployments of Data-center networks optimizing the needs of IT and network resources [3]. SDN architecture aims at making networks flexible and agile. The primary objective of SDN is to bring about an improvement in network control by empowering service providers and enterprises to provide a faster response to ever-evolving business requirements. In a software-defined network, the network administrator can control and monitor traffic without making any changes to the individual switches of the network. These switches are directed by the centralized SDN controller to provision network services based on their requirement, irrespective of the connections between the devices and the server.

The rest of this article is organized as follows: Section II discusses architecture of SDN-IoT. The state-of-art of SDN is presented in Section III. Section IV presents adoption of SDN in IoT applications. Some future research directions have been outlined in Section V. Summary and conclusions are briefed in Section VI.

II. SDN-IoT Architecture

An architecture for the implementation of an SDN-IoT framework is presented as per the design principles discussed as shown in Figure 1. As compared to the traditional IoT protocol stack layers (transport, security, storage, preprocessing, monitoring, physical), SDN-IoT protocol stack

will have facilities for managing security and transport layer with a centralized control plane. SDN-IoT is built upon the traditional IoT protocol stack, with improvisations of control plane in transport and security layer. However, such control plane integration includes real-time challenge of state integration cost. Further, an overview of the classification of individual layer properties of the SDN-IoT architecture is also presented in Figure 2.

A. Backbone Network/Device Layer

The lowest layer of the SDN-IoT architecture is known as the Device layer. This contains devices like sensors, actuators to collect huge amount of data in real-time and the data present in various formats for various IoT domains are further communicated to the controllers. Each such device network has many integration points like Gateways, Routers and an SDN controller which is centralized for this device network. There could be multiple such device networks for a single deployment of the IoT framework. These network-centralized controller further transmit the information to the Router present in the upper WAN layer. The controllers are centrally controlled by the components present in WAN layer. Thus, the controllers can communicate with one another too and can share the load, if required. Service-providers can deploy their SDN-based applications on this control-plane using Northbound APIs.

B. WAN Layer

WAN layer can be considered as the internet layer for this architecture. It consists of devices like Gateways, Routers that form the data-plane of the layer and these routers are managed by an ISP SDN controller that is centralized and forms the control-plane for this layer. The routers/gateways are primarily responsible for forwarding of data in the network. Besides data forwarding, they can cache some local data or process information based on instructions from SDN controller.

Controllers, on the other hand, not only handle data forwarding but also manage data processing. These controllers can perform efficient management of equipment like configuration of gateways/routers, virtual network components, defining policies for data processing in subsequent devices, etc. The control plane here is programmable and centralized, thus, offering an operator the opportunity to tune the network in whatever it wants to.

C. Datacenter Layer

While the data processing and network management was being performed by the lower layers, Datacenter layer can be considered as analogous to a persistence layer for application services. The data generated by all the IoT devices are ultimately stored in this layer. This data can further be fetched by the applications and services executing in the Application layer for further processing and utilization. This layer is continuously fed with data channeled through routers or controller (as is the defined logic) and for a real-time environment, this raises challenges like high availability, disaster recovery of data, handling huge traffic in the communication channels, etc. Most of the data centers employ various techniques like Multiple Availability Zone, ZDM functionalities to handle such use-cases. Modern data-centers have become cloud-based and modern IoT frameworks tend to offload the responsibility on such external cloud providers to manage their data.

D. Application Layer

The most visible part of the architecture is the Application Layer, where all applications consumed by an end-user are deployed. These applications interact with the data using the Datacenter layer, communicate with devices present in Device Layer through the controller in WAN layer and finally assimilates and processes this information to provision services like smart homes, self-driving cars, smart appliances, etc.

III. The State-of-the-Art

IoT network employs a huge number of devices and often, these devices are deployed in a constrained environment. Each of this device usually has its own proprietary architecture. These critical aspects combined with expectation of metrics like QoS, scalability, low latency, energy management, etc. bring forth a lot of challenges in the traditional IoT architecture. This section further classifies the challenges faced by the IoT domain based on the layers of architecture of a simple IoT deployment as shown in Figure 3. The layers expressed in IoT architecture Figure 3 have been abstracted from the layers shown in SDN-IoT framework in Figure 2. SDN-IoT stack facilitates management of security and transport layer with a centralized control plane. Device layer is analogous to Physical layer, WAN is analogous to Network Layer and Datacenter layer is analogous to Data-link layer. Care should be taken to understand that though the concepts of analogous layers are similar, since the layers belong to two different frameworks, specific details may vary.

The domain of SDN has been widely advocated as a silver bullet for most of the issues faced by the IoT domain. The basic question that arises is what is so different in the underlying architecture of SDN that makes it such an effective solution for these challenges.

The major novelty in the architecture of SDN is the decoupling of the Data-plane (forwarding plane) and the Control-plane (control logic used by the network), which were traditionally strongly coupled with each other. Furthermore, the Control Plane resides in a controller which is logically centralized, thus, simplifying configuration and evolution of network and ease of enforcement of policy [4]. Network programmability is another critical aspect brought forth by the SDN paradigm that has resulted in not only improving the performance of the network, but has also eased significantly the management of network, handling of data and control, optimizing usage of network resource etc. This new approach allows a network operator to define programs for the controller, to in-turn manage the data-plane. Armed with functionalities like these, SDN when coupled with IoT can address many challenges pertaining to the domain in a simplified manner. In this section, in addition to the challenges faced by the

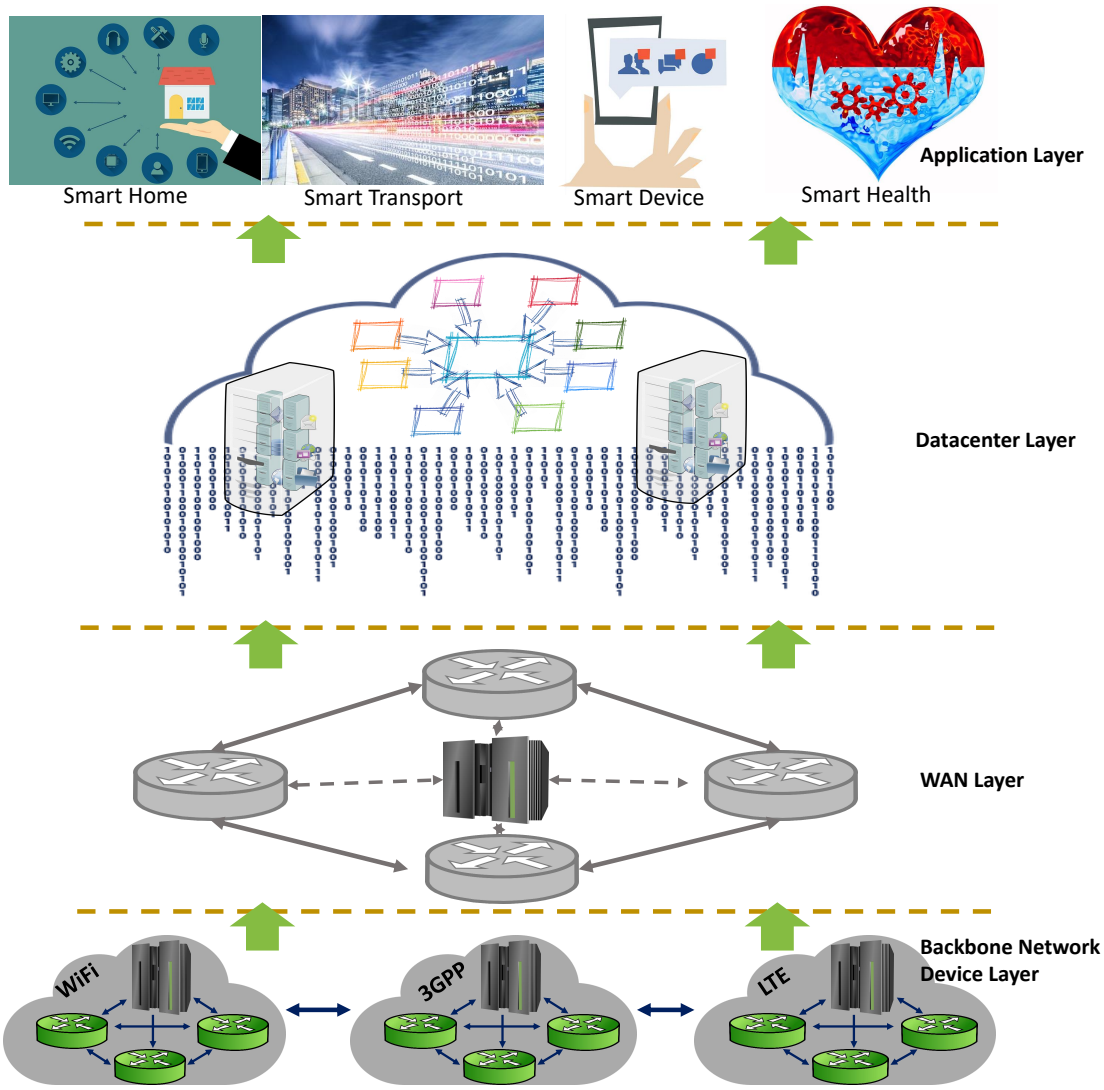

Fig. 1. Architecture of SDN-IoT model

IoT domain, the work also justifies how the challenges can be overcome by introducing SDN-IoT architecture as shown in Figure 4. This section showcases how introducing SDN into the IoT framework helps in tackling security, scalability and routing challenges in Datacenter and WAN layer, load balancing, fault tolerance, latency issues in Network layer and finally, environment constraints, trust issues and authentication concerns in Application Layer.

A. Data-Link & Physical Layer

1) Secure Architecture: For an IoT deployment in a real-time environment, there are a huge number of devices connected to the network, each implementing a different security protocol to protect itself from the various security vulnerabilities. Since these devices are heterogeneous in nature, the threats perceived by each device is different and so also, is the associated security protocol. Thus, it becomes virtually impossible to define a uniform security protocol to encapsulate the entire network. Furthermore, due to this reason, the threats become accentuated at the gateways and other integration points. This drives a need for the IoT-SDN framework to ensure the security of the generated data by the virtualization of network services and components. The improved visibility of the centralized SDN controller also helps in visualizing the security threats for the entire network from a common standpoint and hence, a common centralized protocol can be defined. SDN framework can also be perceived to address any future threats, since the framework can evolve owing to its adaptable nature.

2) Scalability: For a domain like IoT, scalability is probably the most common and undoubtedly the most critical of challenges. The sheer number of miniaturized devices employed (e.g. actuators, sensors, etc) and the amount of data generated by these devices over a period of time magnifies the issue of scalability. The network must be able to handle this amount of traffic and data in a real-time environment, where devices keep getting added and the data-generation keeps becoming voluminous. The Open Level Control (OLC) plane architecture employed by SDN boosts up the scalability

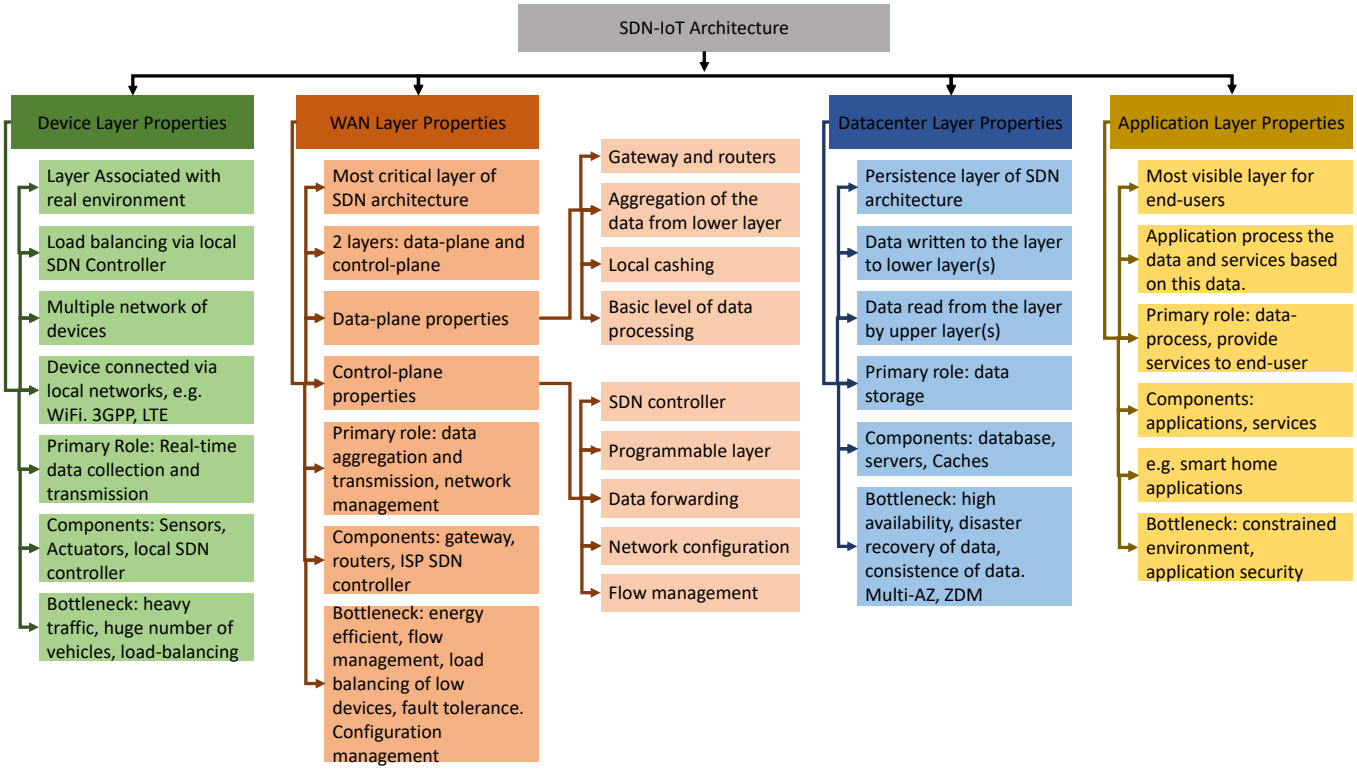

Fig. 2. Classification of properties for each layer of SDN-IoT architecture

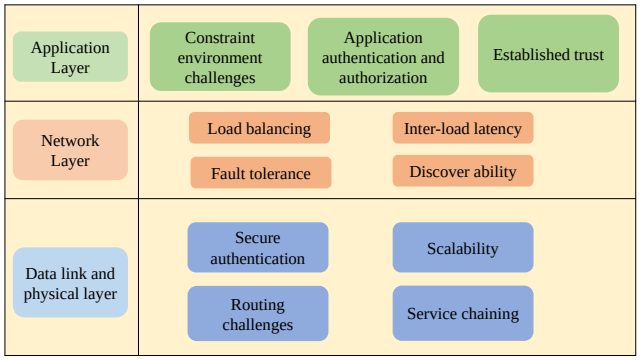

Fig. 3. Classification of challenges faced by traditional IoT architecture

of SDN-IoT framework for a heavy traffic and data-flow without causing any cascading changes in the central hardware's software, tools and protocols. This framework ensures the combination of distributed and centralized architectures simultaneously.

3) Routing challenges: The framework employed by the IoT domain is largely decentralized. Thus, each device and the associated integration points (like aggregators, gateways, routers, etc.) have their own logic of routing in place for handling the flow of data. So, if a new component is added or there is reboot or reconfiguration of the entire network, such separate implementations of routing rules makes the management of the network extremely difficult. The centralized vantage point of the SDN controller facilitates easier management and supervision of the entire network, thus providing a unified vision of the network components and their topology. Thus, it can perform network control operations like routing and control of QoS. It can also formulate the optimized routing rules and can maintain these rules into a centrally-maintained, globally-distributed set of flow tables. Thus, the component devices like sensors are not concerned with the decisions related to routing. Prominent industrial networks in wireless senosor networks (WSN), like WirelessHART and ISA100.11a have been benefited from such a centralized routing framework.

4) Service-chaining: A shorter and simpler service-chain benefits the IoT framework by improving the efficiency and increasing the capacity of the network without requiring a radical change in the hardware, thus easing the processing of spinning up of new IoT applications. It enables operators to dynamically configure their software, eliminating the need to make corresponding modifications to the network at a hardware level. Introduction of the concept of an orchestrator facilitates adding of new services without any upgraded support for the new services (as was needed in the traditional IoT network model). The orchestrator also provisions services and ensures connectivity of network and eliminates the need for overprovisioning because any additional resource can be introduced/added on-the-fly, as and when the necessity arises. Such software-programmed networks enable dynamic and rapidly-evolving IoT applications by enabling service-provisioners for efficient delivery of services.

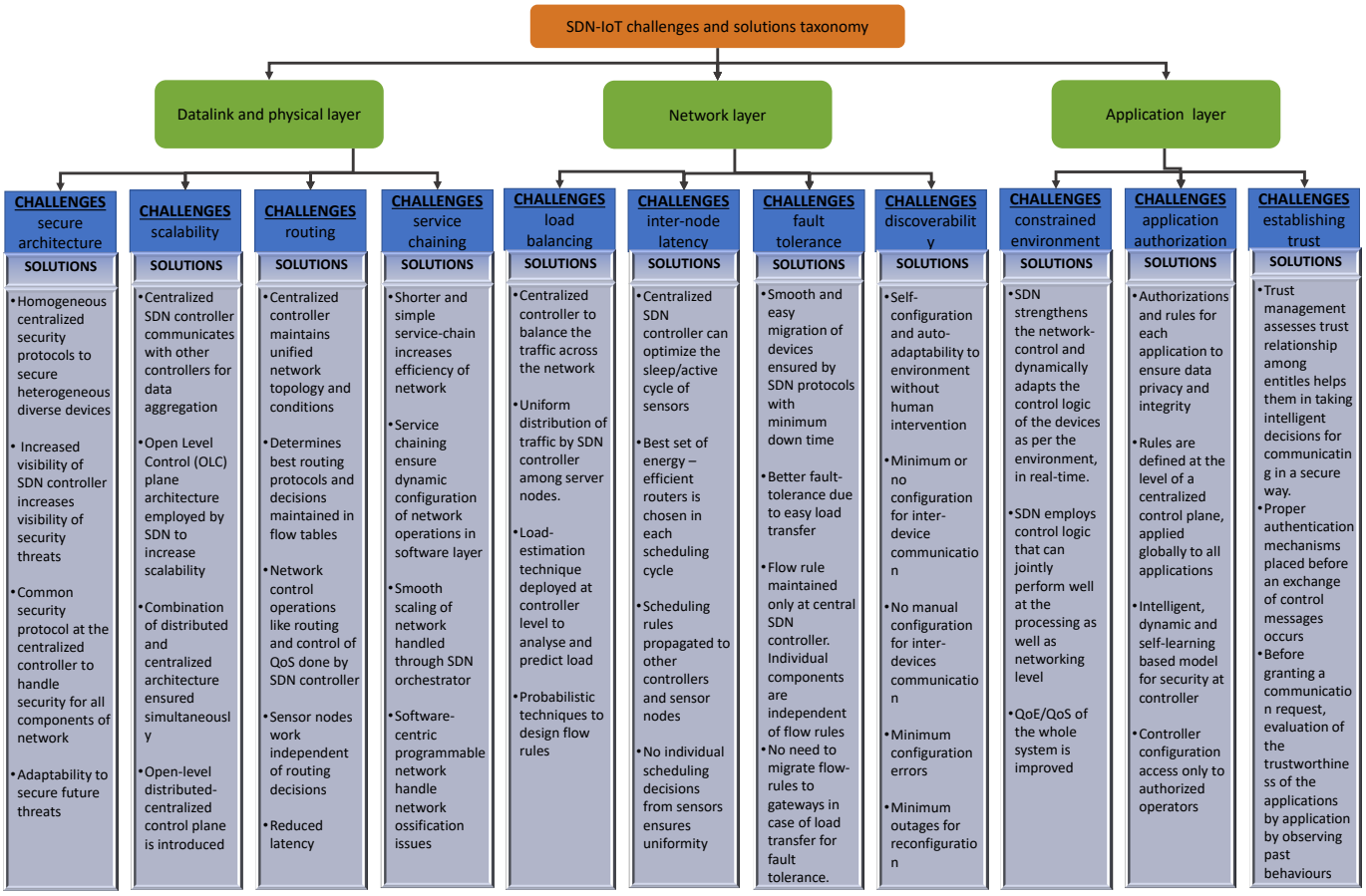

Fig. 4. Taxonomy of challenges faced by IoT and solutions offered by SDN-IoT architecture.

B. Network Layer

1) Load balancing: Optimized load balancing solutions can cause an extended lifetime for IoT devices by reducing the energy consumption over the period of time. Conserving network bandwidth, decrease in energy consumption and reducing redundant data packets could be some major performance improvements brought in by better load-balancing approaches. A general feature brought forth by SDN is to distribute the network traffic amongst multiple virtual machines or servers deployed within a cluster to prevent overloading of any particular host and thereby improving the performance. Centralized control can also help to balance the traffic across the network. A centralized load-estimation technique deployed at the controller level can analyze and determine the predicted load and can design the flow routes for the network. Probabilistic techniques can help in predicting such a load on the IoT nodes.

2) Fault tolerance: Fault tolerance in IoT networks can become very critical, especially for integration points and aggregators. For example, if an integration point like gateway goes down and the entire traffic of the component needs to be migrated to another component, it is very tedious to incorporate these routing rules for the new component. This can cause larger downtime for the entire network and outage for the section till the migration is complete. SDN framework ensures smooth migration in such cases owing to the nature of the centralized implementation of routing rules and protocols at the controller level. One of the key challenges in IoT sustainability is high mobility of sensor nodes/devices. Due to the physically-distributed and centralized control-plane of SDN caters to this issue very easily. There have been numerous approaches proposed in the SDN domain that also ensure easy device migration with minimum downtime.

3) Inter-node latency: Owing to the large number of devices and the varying architectures of each device, latency is also a major concern in the traditional IoT model. However, an efficient scheduling algorithm implemented and executed at the central SDN controller can optimize the sleep/active cycles of sensors and can choose best set of energy-efficient routes in each scheduling cycle. Energy consumption often spikes in IoT by exploitative duty cycles, i.e., large chunks of time spent by nodes in OFF state, or transmission power control, i.e., nodes transmitting at the power level that best suits the current transmission conditions. Thus, such a unified smart scheduling framework can reduce latency significantly and can ensure huge savings of energy.

4) Discoverability: Discoverability is another critical aspect of IoT networks which is one of the most essential factors for a successful IoT application deployment. Any issues with the discoverability aspect can cause configuration errors and thus, contribute to long duration of outages. With the number of devices growing in the IoT sector and owing to the hostile nature of the environment in which these devices are deployed

in, there is a need to drive the design of an intelligent, self-configurable network that can eliminate the need of any human intervention or manual configuration and is adaptable to the changing environment. SDN-IoT perfectly fits the picture since it enables applications and devices to operate with minimal to zero configuration cost.

C. Application Layer

1) Constrained Environment: IoT applications are generally deployed in a real-time environment where there is constraint exerted on almost every resource. Each application further has its own constraints on account of the diverse architecture employed. SDN strengthens the network-control and dynamically adapts the control logic of the devices as per the environment, in real-time. SDN and the extended modules of WSN domain bring in new possibilities of supporting application-oriented requirements paired with a control logic that can jointly perform at the processing as well as networking level, thereby improving the QoE/QoS of the whole system.

2) Application Authentication & Authorization: Alongside the huge number of devices employed by the IoT framework, also associated are a number of applications with these devices. The applications read data from the devices like sensors, actuators, etc., process and analyze this data and are expected to consume the data in some way or the another. However, which data should accessible to which application and how to prevent sensitive information from getting leaked to hostile applications and segregation of concerns to separate the visibility of data in a multi-user kind of scenario are some key challenges. The SDN-IoT framework must ensure that the applications are properly authenticated to prevent hostile applications from getting access to the network or the data. It must also maintain authorizations and roles for each application to ensure data privacy and integrity. These rules are defined at the level of a centralized control plane such that it can be applied globally to all applications. SDN also employs an intelligent, dynamic and self-learning based model for security which can provide access protocols and rules to ensure only authorized operators are permitted to modify the device configurations.

3) Established Trust: For a dynamic and open network framework setup like SDN, trust management is critical to ensuring defense against attacks and gain confidence of users for using the applications. Trust management assesses trust relationship among entities and helps them in taking intelligent decisions for communicating and collaborating in a secure way. In the context of SDN, it is quintessential that an element of trust exists between the deployed IoT applications and the SDN controller for a trustworthy communication. SDN introduces a dynamic trust model where malicious or hostile applications can be prevented from harming the network and proper authentication mechanisms are placed before an exchange of control messages occurs. Thus, protection of the controller is ensured by leveraging trust to perform evaluation of the trustworthiness of the applications by observing the behavior before granting a communication request.

IV. SDN Adoption For IoT Systems

Considering the aspect of enhanced manageability offered by the SDN framework, various works have probed the applicability of the domain in providing IoT applications in various fields. SDN model can be adopted into an IoT system at various levels, such as, access networking, data center, cloud services, management of traffic for devices, etc. Every such network environment presents challenges pertaining to optimization for SDN adoption and meeting the specific demands of the field. It is indeed remarkable how IoT applications span an immensely broad range of services and devices, ranging from resource-constrained sensors in WSN to smart home devices, self-driving cars, etc. In this diverse landscape, SDN provides an enhanced management of various IoT environments owing to its programmability and flexibility. That being said, specific challenges of IoT domain require unique enhancements in SDN model, as can be visualized in the following exemplary scenarios of IoT.

Several case studies have been performed to advocate the adoption of SDN for communications in vehicles, citing the improvement in network utilization and swift automated configuration of network provided by the framework. For ensuring the mobility of vehicles and to meet the latency demands for vehicular safety applications, optimized strategies of dynamic routing protocols employed by SDN are required. There have implementations of SDN-IoT architecture designed specifically to accommodate the features and constraints of WSNs, like reduction in operating cost of control traffic. TinySDN architecture has been presented for enabling various controllers in software-defined WSNs deployed in TinyOS-enabled devices [5]. One unique feature of this implementation deals with the presence of diverse controllers deployed within the same WSN and needs to reduce the aggregate latency.

Adoption of SDN framework for Wi-Fi-based and cellular-based IoT networks have also been proposed in [6], wherein, OPENSDWN, a unique Wi-Fi model utilizes the SDN network to provide programmability of data path and enables differentiation of services and acute control of transmission, thus ensuring that critical applications are efficiently prioritized. Another flexible SDN-based architecture has been developed for 5G cellular access networks to meet the performance and functional goals of such next-generation services and futuristic IoT devices [7].

Another stateful implementation of SDN, SDN-WISE [8], that has been designed for IEEE 802.15.4, can improve programmability of sensors as finite state machines, in-order to be able to perform a variety of in-network functions. An improvised SDN controller has also been designed to ensure discrete QoS for separate IoT flow for various heterogeneous networking schemes [9]. For ensuring a smooth integration of SDN-WISE based sensors coupled with OpenFlow networks, the open-source framework of ONOS has now been extended [10]. Now, by anchoring on the new features introduced due to ONOS controller, a couple of new applications have come to efficiently leverage the SDN functionalities for sensor devices. While the application of SensorNodeForwarding installs requisite forwarding protocols, by monitoring the global

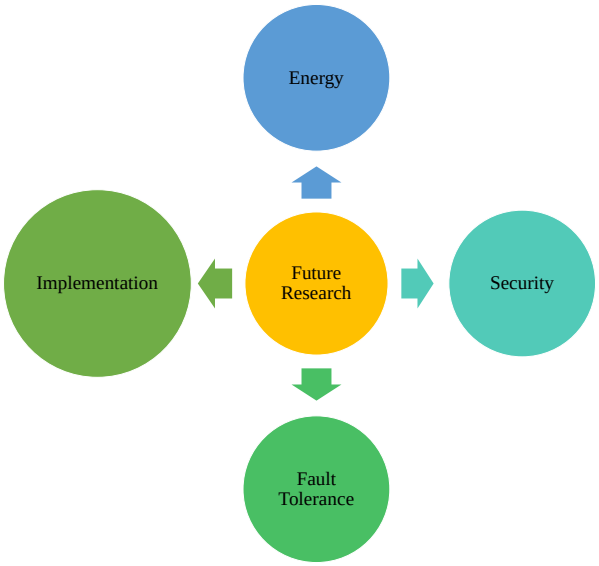

Fig. 5. Future research in SDN-IoT

topology (that consists of both SDN-WISE and OpenFlow nodes), on the other side, the SensorNodeDeviceManagement implements management of remote sensors, thereby increasing the flexibility of resource usage [11]. SDN-WISE showcases a valuable use-case for demonstrating efficient energy management by supporting both duty cycle and transmission power control using SDN-IoT.

A detailed analysis of performance evaluations of SDN-based implementation has been performed in [12], thus proving that, under quasi-static and static conditions, SDN beats the two conventional IoT network protocols like 6LoWPAN and Zigbee. This is independent of any size of network, size of payload, amount of traffic, etc. There have been numerous such adoptions of SDN in various fields of IoT applications that exert and drive home the importance and practicability of the topic.

V. Future Research

This section focuses on defining and identifying the most prominent open-research areas for the end-to-end deployment of SDN-based solutions in IoT network as shown in Figure 5. This discussion aims at performing a comprehensive analysis of research efforts encompassing SDN-based implementations at all levels of IoT, such as core networks, access networks, data-centers and aspects like secure architecture, load balancing, fault tolerance, scalability, energy management, etc. The scope of future research has been broadly organized in 4 critical aspects of SDN-IoT framework: Security, Energy management, Fault-tolerance/Sustainability and Real-time challenges. Concerns pertaining to each of these sections are discussed in the sub-sections.

A. Security Aspects:

For addressing the security challenges, there have been many efforts made for defining security policies. A security framework based on OpenFlow model, OpenSec [13], has also been proposed to enable a security admin to define policies in a human-readable format. However, there are many distinct features in IoT domain that require extension of definition of policy for considering contextual information to enforce more effective protection mechanisms. Security policies must also take into consideration any interaction between smart objects belonging to the same environment. This, thus, drives a need to enhance the current security policies to handle such nuances of the IoT network. Security mechanisms must also be able to design solutions for heterogeneous deployments of IoT. There has to be optimal selection of security protocols that can account for service quality, especially for applications executing in constrained environments and performing mission-critical operations. These protocols must be designed keeping in mind the scalability, reliability and latency aspects of the network.

B. Energy management

Developing a solution for efficient management of IoT networks doesn't have a very straightforward solution. Although SDN domain offers numerous possibilities in this area, owing to the reprogrammability of its devices, management challenges pertaining to energy management, fault tolerance, and load balancing still persist. While there have been efforts to construct such a solution for management, none of the proposed solutions currently address the challenges that need to be tackled in this area. Additionally, most of the existing works lack a prototype implementation and real-time evaluation of the prototype.

C. Fault-tolerance & Sustainability

Another key area for future work could be the need for an in-depth investigation of how the centralized SDN controller in IoT network can perform automated recovery of faulty nodes present in the network. Similar investigation is also required for provisioning secure services employing SDN principles in IoT network. There have been works to address fault tolerance and load balancing [14], but majority of works aren't capable of handling this issue. There have been proposals on how the centralized SDN controller can enable efficient energy management of nodes in IoT network by load-balancing the traffic, however, there have been very few architecture implementations.

D. Practical Implementation & Performance Evaluation

There is not much concrete work till now that can provision a concrete, concise solution for the efficient management of an IoT network. Most of the existing works have proposed architectures that somehow lack the detailed workflow of functional components for management of IoT network. There have been very few works that have examined the proposed model thoroughly exerting its effectiveness. There have been works that compare these efforts in terms of metrics like fault tolerance, load balancing, energy management and secure infrastructure. A few of these works have also addressed the issue of providing minimum level of secure provisioning

of services, although there is still scope to improve this further [15]. There have been other works on securing provisioning of services in IoT framework comprehensively, but majority of works aren't capable of handling this issue with the proposed SDN framework. Thus, for the purpose of future works in this domain, works especially tackling challenges of energy management, fault tolerance, load balancing and security will remain prominent research areas.

VI. Conclusions

This article has presented a comprehensive overview of the SDN-IoT architecture and perspectives of its future. The work justifies the need of models in the current architecture and signifies how the new architecture can benefit the end-to-end performance. The practicability of the solution is also thoroughly highlighted. The open research areas are probed for guiding future works in this area from the worldwide reseachers. The SDN-IoT domain has been validated now in terms of the concept and its advantages no longer in question. Many practical implementations of the concept have also ensured few industrial adoptions. However, the domain could still benefit from more widespread industry adoptions and open-sourced contributions in tackling the prevalent open-issues.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] C. J. Bernardos, A. De La Oliva, P. Serrano, A. Banchs, L. M. Contreras, H. Jin, and J. C. Zúñiga, "An architecture for software defined wireless networking," *IEEE Wireless Communications*, vol. 21, no. 3, pp. 52–61, 2014.

[3] K. Wang, Y. Wang, D. Zeng, and S. Guo, "An SDN-based architecture for next-generation wireless networks," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 25–31, 2017.

[4] D. Kreutz, F. M. Ramos, P. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.

[5] B. T. De Oliveira, L. B. Gabriel, and C. B. Margi, "TinySDN: Enabling multiple controllers for software-defined wireless sensor networks," *IEEE Latin America Transactions*, vol. 13, no. 11, pp. 3690–3696, 2015.

[6] J. Schulz-Zander, C. Mayer, B. Ciobotaru, S. Schmid, and A. Feldmann, "OpenSDWN: Programmatic Control overHome and Enterprise WiFi," in *Proceedings of the 1st ACM SIGCOMM symposium on software defined networking research*, 2015, p. 16.

[7] W. H. Chin, Z. Fan, and R. Haines, "Emerging technologies and research challenges for 5g wireless networks," *IEEE Wireless Communications*, vol. 21, no. 2, pp. 106–112, 2014.

[8] L. Galluccio, S. Milardo, G. Morabito, and S. Palazzo, "SDN-WISE: Design, prototyping and experimentation of a stateful SDN solution for WIreless SEnsor networks," in *IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 513–521.

[9] Z. Qin, G. Denker, C. Giannelli, P. Bellavista, and N. Venkatasubramanian, "A software defined networking architecture for the Internet-of-Things," in *IEEE network operations and management symposium (NOMS)*, 2014, pp. 1–9.

[10] P. Berde, M. Gerola, J. Hart, Y. Higuchi, M. Kobayashi, T. Koide, B. Lantz, B. O'Connor, P. Radoslavov, W. Snow *et al.*, "ONOS: towards an open, distributed SDN OS," in *Proceedings of the third workshop on Hot topics in software defined networking*, 2014, pp. 1–6.

[11] A.-C. G. Anadiotis, L. Galluccio, S. Milardo, G. Morabito, and S. Palazzo, "Towards a software-defined Network Operating System for the IoT," in *IEEE 2nd World Forum on Internet of Things (WF-IoT)*, 2015, pp. 579–584.

[12] C. Buratti, A. Stajkic, G. Gardasevic, S. Milardo, M. D. Abrignani, S. Mijovic, G. Morabito, and R. Verdone, "Testing protocols for the internet of things on the EuWIn platform," *IEEE Internet of Things Journal*, vol. 3, no. 1, pp. 124–133, 2016.

[13] A. Lara and B. Ramamurthy, "OpenSec: Policy-based security using software-defined networking," *IEEE transactions on network and service management*, vol. 13, no. 1, pp. 30–42, 2016.

[14] D. Wu, D. I. Arkhipov, E. Asmare, Z. Qin, and J. A. McCann, "UbiFlow: Mobility management in urban-scale software defined IoT," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 208–216.

[15] J. Li, E. Altman, and C. Touati, "A general SDN-based IoT framework with NVF implementation," *ZTE communications*, vol. 13, no. 3, pp. 42–45, 2015.

Biographies

Pritish Mishra (pritish.mishra@sap.com) is working as a core developer in SAP Cloud Platform at SAP Labs Bangalore, India. He has graduated from International Institute of Information Technology, Bhubaneswar, India. He has numbers of publications in the domain of SDN, IoT, distributed and cloud computing.

Deepak Puthal (deepak.puthal@newcastle.ac.uk) is a Lecturer at School of Computing, newcastle University, UK. His research interests include cyber security, Internet of Things, distributed computing, and edge/fog computing. He has received several reorganizations and best paper award from IEEE. He is an Associate Editor of the IEEE Transactions on Big Data, Computers & Electrical Engineering (Elsevier), International Journal of Communication Systems (John Wiley & Sons), IEEE Consumer Electronics Magazine and Internet Technology Letters (John Wiley & Sons).

Mayank Tiwary (mayank.tiwary@sap.com) is working as a core developer in SAP Cloud Platform at SAP Labs Bangalore, India. He has graduated from Biju Patnaik University of Technology, Odisha, India. He has numbers of publications in the domain of SDN, distributed and cloud computing.

Saraju P. Mohanty (saraju.mohanty@unt.edu) is a Professor at the University of North Texas. Prof. Mohanty's research is in "Smart Electronic Systems" which has been funded by National Science Foundations, Semiconductor Research Corporation, US Air Force, Mission Innovation, and IUSSTF. He has authored 300 research articles, 4 books, and invented 4 US patents. He has received 6 best paper awards and has delivered Keynotes at various International Conferences. He received IEEE-CS-TCVLSI Distinguished Leadership Award in 2018 for services to the IEEE, and VLSI research community. He has been recognized as an IEEE Distinguished Lecturer by the Consumer Electronics Society (CESoc) during 2017-2018.

He was the recipient of 2016 PROSE Award for best Textbook in Physical Sciences & Mathematics category from the Association of American Publishers for his Mixed-Signal System Design book published by McGraw-Hill in 2015. He is the Editor-in-Chief (EiC) of the IEEE Consumer Electronics Magazine (CEM). He served as the Chair of Technical Committee on VLSI, IEEE Computer Society during 2014-2018.