SDN BASED IOT WITH EXTEND MULTI NETWORK INFORMATION METHOD

S.Sridhar Dhanabalan

M.Phil, Research Scholar,

Mail ID–sridhardhanabalans@gmail.com

CMS College of Science & Commerce, Chinnavedampatty-641049

Bharathiar University, Coimbatore, Tamil Nadu, India.

Sujatha Padmakumar

Associate Professor (PhD),Department of Computer Applications

Mail ID –sujathapadmakumar4@gmail.com

CMS College of Science & Commerce, Chinnavedampatty-641049

Bharathiar University, Coimbatore, Tamil Nadu, India

Abstract— Software-Defined Networking (SDN) is an emerging networking paradigm that gives hope to change the limitations of current network infrastructures. SDN is one of the upgrading one for the improvement of network traffic. The growing interest in the Internet of Things (IoT) has resulted in a number of wide-area deployments of IoT subnetworks, where multiple heterogeneous wireless communication solutions coexist from multiple access technologies such as cellular, WiFi, ZigBee, and Bluetooth, to multi-hop ad-hoc and MANET routing protocols, they all must be effectively integrated to create a seamless communication platform. Managing these open, geographically distributed and heterogeneous networking infrastructures, especially in dynamic environments, is a key technical challenge. For this we planned to design a software-defined approach for the IoT environment to dynamically achieve differentiated quality levels to different IoT tasks in very heterogeneous wireless networking scenarios with MNIA mechanism.

Keywords*: Software Defined Network, IOT, Network Management and Distributed Network.*

I. INTRODUCTION

Software-Defined Networking (SDN) is an emerging networking paradigm that gives hope to change the limitations of current network infrastructures. SDN is being circulated to different networking system. It is the ability to program network performance in open way using languages, systems, computers that are ordinary. Networking is being change into software discipline. Even in the major implication networking is becoming the part of computing. To enhance the performance and for better traffic management SDN can be applied on the fog devices [1]. SDN is the rising prototype that makes deportment of vehicular switches programmable and allowed to be controlled by central element known as controller. The Software Defined Networking (SDN) paradigm introduces a centralized and programmable way of designing networks and was designed to face the shortcomings of traditional networks, such as manual configuration and maintenance of every single device in the network, high latency in path-recovery due to distributed approach, etc [2].

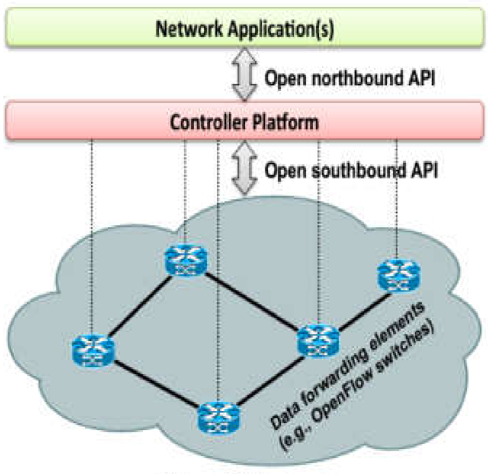

Figure 1: View of SDN

SDN separates the data plane from the control plane, improving the programmability of the network by external applications [3]. In an SDN-based network, the intelligence is centralized in a network controller, which determines how traffic flows will be forwarded in the network; while network devices (switches, routers) simply forward the packets, following the per-flow rules installed by the controller.

The Internet of Things (IoT) is the interconnection of extraordinarily identifiable installed processing gadgets inside the current Internet framework [4]. Ordinarily, IoT is required to offer propelled network of gadgets, frameworks, and administrations that goes past machine-to machine interchanges (M2M) and spreads an assortment of conventions, spaces, and applications. The IoT Multi-networks environment is to accomplish heterogeneous IoT tasks with various requirements. As a step towards this ambitious goal, we have developed MNIA, a reflective self observing and adapting middleware exploiting the Observe- Analyze-Adapt loop, to realize and manage dynamic and heterogeneous multi-networks in pervasive environments [5]. In particular, MINA achieves a reasonably accurate, centralized global view of the currently available multi-network environment and takes advantage of this global view for adapting it, e.g., by reallocating application flows across paths. More importantly, MINA adopts state-of-the-art Software- Defined Networking (SDN) technologies to achieve flexible resource matching and efficient flow control in industrial deployment environments. To this purpose, we propose a novel IoT multinetwork controller, based on a layered architecture that makes easier to flexibly and dynamically exploit IoT networking capabilities for different IoT tasks described by abstract semantics [6]. Moreover, we modify and exploit the Network Calculus to model the available IoT multi-network and we propose a genetic algorithm to optimize its exploitation through differentiated dynamic management of heterogeneous application flows.

II.LITERATURE REVIEW

Heebum Yoon [7], By effectively coupling IoT and Cloud together, we need to enable diversified IoT-Cloud services. To effectively support IoT-Cloud services, it is essential to maintain persistent data transport between IoT and Cloud. Also the emerging Software-Defined Networking (SDN) paradigm can assist flexible flow-centric networking of persistent data between IoT and Cloud. Thus, in this paper, we take an example of IoT-Cloud service realized over miniaturized IoTSDN- Cloud environment (named as SmartX-mini Playground) and propose the application of SDN-based flow steering to dynamically adjust the overlay data paths for IoT-Cloud services. More specifically, we attempt to combine the overlay data transport of Apache Kafka messages and the underlay flow-based networking coordinated by an ONOS (Open Networking Operating System) SDN controller. To support the persistent data delivery of IoT monitoring data, we implement ONOS northbound application called as ‘Steering Function’, whose implementation detail is depicted. It covers the role of delivering monitoring packets by manipulating host-to-host forwarding rules according to the single-to-multiple and multiple-to-single intents of ONOS SDN controller.

Hai Huang [8] The ever evolving technology of the Internet of Things(IoT) requires ubiquitous connectivity to billions of heterogeneous devices such as sensors, cameras, RFID devices, etc. However, due to the heterogeneity of devices and access protocols, IoT networks are becoming enormous and complex, which makes the management extremely difficult. Based on Machine-to-Machine(M2M) technics and the programmability feature of the network, brought by the Software Defined Network(SDN), devices in a network can be treated as objects, thus decoupling the control plane from the data plane. In this paper, we proposed a framework for managing the devices and configuring the network dynamically based on SDN. Finally, we apply the framework in our project and get a successful result.

Łukasz Ogrodowczyk, Barosz Belter [9] presents an innovative Software Defined Networking (SDN) based approach to deploying Internet of Things (IoT) applications for Smart Cities. The Poznan Supercomputing and Networking Center (PSNC) together with NoviFlow Inc. and Spirent have jointly developed a demonstration showing how programmable SDN infrastructure can be utilized to significantly simplify the onboarding and provisioning of end-toend IoT solutions for use in multi-tenant networks. The demo features a dynamic global view of the deployed IoT resources with their associated network connections, and the use of OpenFlow Experimenter-based extensions to trigger automated detection and onboarding of IoT devices, ("things") as well as the insertion of metadata into IoT device flows to automate service provisioning. To illustrate the use of the solution in a real-world setting, the Poznan Smart City use case is presented, showing how a single common SDN-based platform can be utilized to "slice" a city into multiple smart spaces running over a shared network and cloud infrastructure, and how OpenFlow-enabled network infrastructure can be used to automate the deployment of IoT devices for use in multi-tenant, cloud-based applications.

Do Sinh, Luong-Vy Le [10] The Internet of Things (IoTs) is a recent concept in which variety of smart devices such as smartphones, tablets, and sensors can interact with one another through the Internet. Hence, a plethora of data is generated and grown rapidly. They have posed various requirements of high throughput, high availability, lower latency, and high performance in heterogeneous connectivity environments. However, existing networks often struggle with such of limitations such as the complexity of control protocols and the internetworking of ahuge number of smart devices. These obstacles become substantial barriers to deploy IoT services from different providers on the same network. In this context, SDN/NFV not only are considered as the key innovations in technology and novel services but also are expected to be a new model of network infrastructure that meets the requirement of developing IoT applications. In this study, the authors focus on the following contributions: *i*) investigate the roles of SDN/NFV in deploying IoT services and introduce an SDN/NFV architecture for deploying IoT framework; *ii*) build applications to slice end-to- end multiple network segments to meet requirements of deploying IoT services from different providers; *iii*) build an application to recover an IoT service when it is out of order.

III. PROBLEM STATEMENT

The ever evolving technology of the Internet of Things (IoT) requires ubiquitous connectivity to billions of heterogeneous devices such as sensors, cameras, RFID devices, etc. However, due to the heterogeneity of devices and access protocols, IoT networks are becoming enormous and complex, which makes the management extremely difficult. Based on Machine-to-Machine (M2M) techniques and the programmability feature of the network, brought by the Software Defined Network (SDN), devices in a network can be treated as objects, thus decoupling the control plane from the data plane. In this paper, we proposed a framework for managing the devices and configuring the network dynamically based on SDN. Finally, we apply the framework in our project and get a successful result [11].

- Problems with end-to-end IP networking to resource
- Low adaptation and/or mapping functions

- Difficult to Manage a large number of devices with variety of IoT protocols
- It provide Capability mismatch between devices

IV. PROPOSED METHOD

The data collection component collects network/device information from the IoT Multinetworks environment and stores it into databases. This information is then utilized by the layered components in the left side. The controller also exposes the Admin/Analyst APIs, which enable the control processes to be governed not only by the controller itself but also by humans or external programs. Note that while the controller is logically centralized, to improve scalability it can be instantiated multiple times in different locations, e.g., in a per-domain per-service fashion [12]. IoT Multinetworks that define what is required; this leaves open the choice of what applications/services, devices and communication networks should be exploited to accomplish the required task. A simple example might be to determine how many vehicles currently there are in a recharging station. Services are concrete software/hardware entities that help in the realization of a task. A task may be realized by a single service (capture video from recharging station) or a workflow of services that together realize the task (capture video and count vehicles). A task/service mapping specifies which devices and applications should be used to complete the task.

Benefits

- The benefits of employing SDN techniques in IoT environments is becoming recognized in multiple domains beyond the smart transportation setting discussed earlier by both researchers and industry practitioners.
- It provides a robust control and communication platform.
- It hides the details of lower layers (network/devices) so that tasks can be accomplished in a more flexible way.

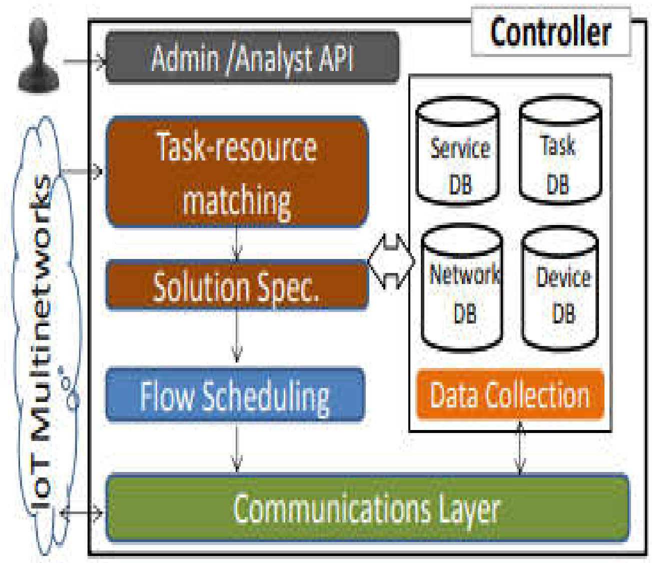

Figure 2: Proposed Architecture

In this paper, we propose a novel IoT Multinetworks controller architecture to overcome these limitations. As shown in Fig. 1, the data collection component collects network/device information from the IoT Multinetworks environment and stores it into databases. This information is then utilized by the layered components in the left side. The controller also exposes the Admin/Analyst APIs, which enable the control processes to be governed not only by the controller itself but also by humans or external programs. the highest level of abstractions in IoT Multinetworks that define what is required; this leaves open the choice of what applications/services, devices and communication networks should be exploited to accomplish the required task [13, 14]. A simple example might be to determine how many vehicles currently there are in a recharging station. Services are concrete software/hardware entities that help in the realization of a task. A task may be realized by a single service (capture video from recharging station) or a workflow of services that together realize the task (capture video and count vehicles).

A task/service mapping specifies which devices and applications should be used to complete the task. The task-resource matching component of the controller maps the task request onto the existing resources in the multinetwork. The task-resource matching component then further refines each of the resource solution. In our example of locating and tracking vehicles, for the solution that consists of a road camera and a server for image processing, the refinement yields that the video stream coming from the raod camera is sent to the server and it also determines the image processing techniques that will be employed at the server side [15]. Once a solution is selected, the service solution specification component of the controller maps the characteristics of the devices and services involved in that solution to specific requirements for devices, networks, and application constraints (e.g., minimum throughput).

The Flow Scheduling component takes these requirements and schedules flows that satisfy them. Scheduling and coordination of the resources in IoT Multinetworks are complex due to the heterogeneity of the networks and various QoS requirements of flows [16]. Finally the controller triggers the necessary communications in the IoT Multinetworks, e.g., a command like "routing the video data sent from Camera via Ethernet" will be sent to the devices along the path. Solution Spec is used with the help of Exploiting neighbor changes method and it uses the least squares (LSQ)-based algorithm to transmit the data in continuous flow. The basic idea of our LSQ-based algorithm is to localize the jammer according to the changes of a node's hearing range. To simplify the algorithm description, we start by assuming the node hearing range The packet sent is computed according to the changes of a node's in network and it provides a continuity of the data using SDN controller with the help of IOT.

V. EXPERIMENTAL RESULT

In this section, we conduct an extensive simulation study to evaluate the performance. environment for designing network protocols, and it enables creating and animating different network scenarios, under which the performance of the protocols can be analysed. We customized Qualnet with SDN features by injecting a OpenFlow-like protocol in IP layer. In every network scenario, there is only one node serving as the controller and the remaining nodes are all controlled devices.Each node is half-duplex thus cannot transmit and receive at the same time. We adopt the protocol interference model; i.e., if multi nodes are within each other's transmission range, There are in total 10 end-to-end flows with randomly selected source-destination pairs in the network.

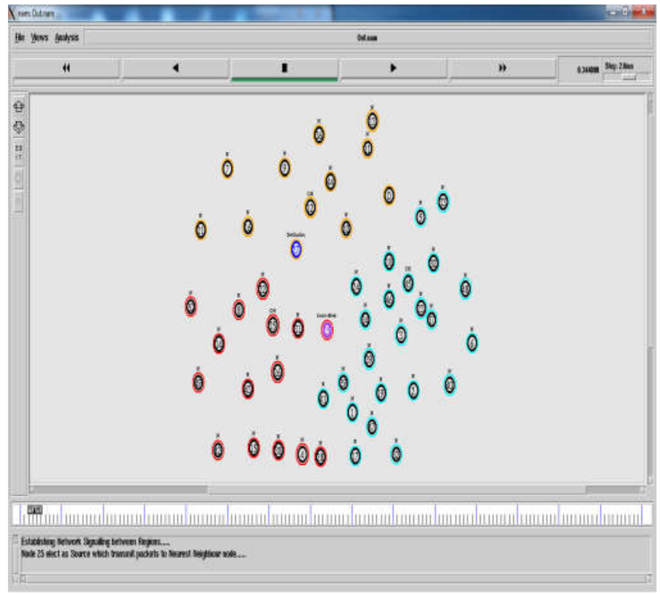

Fig 3: Node initialization

This information might be conveyed over a wired or remote connection, or may go through a specific system customer. we initially choose two paths for each flow as parents. Under this specific network topology, we choose the path generated by load balance algorithm as one of the parents; then, we determine the other parent by exchanging the current core route with the alternative one.

Fig 4: Packet rate transmission

Packet delivery ratio is calculated by ratio of packet received by the destination nodes to those generated by the source nodes. Hence the node density increases, the packet delivery ratio of dynamic path selection

Fig 5: Throughput Rate

Sum is calculated by sending and receiving packet at the instance. In the notable time how many processes of segments of information it can be measured is calculated.

VI CONCLUSION

The growing interest in the Internet of Things (IoT) has resulted in a number of wide-area deployments of IoT subnetworks, where multiple heterogeneous wireless communication. They were facing geographically distributed and heterogeneous networking infrastructures problem.. To overcome this issue in this paper we structured a software-defined approach for the IoT environment to dynamically achieve differentiated quality levels to different IoT tasks in very heterogeneous wireless networking scenarios with Multi Network Information Architecture (MNIA) mechanism and variant of Network Calculus model is developed to accurately estimate the end-to-end flow performance in IoT Multinetworks, which is further serving as fundamentals of a novel multiconstraints flow scheduling algorithm under heterogeneous traffic pattern and network links.

VII REFERENCE

[1]. SDxCentral. (2017). "What's Software-Defined Networking (SDN)?". [online] Available at: https://www.sdxcentral.com/sdn/definitions/what-the-definition-of-software-defined-networking-sdn/ [Accessed Feb. 2017].

[2]. Yan, Q., Yu, F.R., Gong, Q. and Li, J., 2016. Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges. *IEEE Communications Surveys & Tutorials*, *18*(1), pp.602-622.

[3]. Zongjian He, Jiannong Cao, Xuefeng Liu, "SDVN: Enabling Rapid Network Innovation for Heterogeneous Vehicular Communication", IEEE Network, May/June 2016.

[4]. K. S. Yeo, M. C. Chain, T. C. W. Ng, and A. T. Do, "Internet of Things: Trends, challenges and applications," in *Proc. IEEE ISIS,* Mar. 2014.

[5]. K. Sood, S. Yu, and Y. Xiang, "Software-defined wireless networking opportunities and challenges for internet-of-things: A review," *IEEE Internet of Things Journal*, vol. 3, no. 4, pp. 453–463, Aug 2016.

[6]. R. Leela, N. Thanulekshmi, and S. Selvakumar, "Multi-constraint qos unicast routing using genetic algorithm (muruga)," Appl. Soft Comput., vol. 11, no. 2, Mar. 2011.

[7]. Heebum Yoon, "Dynamic Flow Steering for IoT Monitoring Data in SDN-coordinated IoT-Cloud Services", ©2017 IEEE.

[8]. Hai Huang, "An SDN_based Management Framework for IoT Devices", ISSC 2014 / CIICT 2014, Limerick.

[9]. Łukasz Ogrodowczyk, Barosz Belter, "IoT ecosystem over programmable SDN infrastructure for Smart City applications"

[10]. Do Sinh, Luong-Vy Le, "SDN/NFV - A new approach of deploying network infrastructure for IoT", ©2018 IEEE

[11]. Zhao Y, Zhang J, Gao L, et al, "Unified control system for heterogeneous networks with Software Defined Networking (SDN) //Communications and Networking in China (CHINACOM)", 2013 8th International ICST Conference on. IEEE, 2013, pp 781-784.

[12]. Hampel G, Steiner M, Bu T, "Applying software-defined networking to the telecom domain//Computer Communications Workshops (INFOCOM WKSHPS)", 2013 IEEE Conference on. IEEE, 2013, pp133-138.

[13]. Sydney.A, "The evaluation of software defined networking for communication and control of cyber physical systems," Ph.D. dissertation, Department of Electrical and Computer Engineering College of Engineering, KANSAS STATE UNIVERSITY, Manhattan, Kansas, 2013.

[14]. Y. Yiakoumis, K.-K. Yap, S. Katti, G. Parulkar, and N. McKeown, "Slicing home networks," in Proceedings of the 2nd ACM SIGCOMM workshop on Home networks, ser. HomeNets '11. New York, NY, USA: ACM, 2011, pp. 1–6. [Online].

[15]. R. Leela, N. Thanulekshmi, and S. Selvakumar, "Multi-constraint qos unicast routing using genetic algorithm (muruga)," Appl. Soft Comput., vol. 11, no. 2, Mar. 2011.

[16]. F. Xiang, L. Junzhou, W. Jieyi, and G. Guanqun, "Qos routing based on genetic algorithm," Computer Communications, vol. 22, no. 1516, pp. 1392 – 1399, 1999.