On a Novel Deep-Learning-Based Intelligent Partially Overlapping Channel Assignment in SDN-IoT

Fengxiao Tang, Bomin Mao, Zubair Md. Fadlullah, and Nei Kato

The rapid increase of IoT devices and the subsequent massive surge of the IoT data traffic are expected to place a huge strain on the SDN-IoT. The authors focus on this issue and point out the importance of assigning suitable channels to each SDN-IoT switch to avoid potential network congestion. In particular, they consider how to exploit POC assignment in the SDN-IoT.

Abstract

Recently, SDN has emerged as a promising technology to cost-effectively provide the scale and flexibility necessary for IoT services. In this article, we consider the wireless SDN for IoT, referred to as SDN-IoT, which is anticipated to smartly route traffic and use underutilized network resources to deliver IoT data to the cloud/Internet. However, the rapid increase of IoT devices and the subsequent massive surge of the IoT data traffic are expected to place a huge strain on the SDN-IoT. In this article, we focus on this issue and point out the importance of assigning suitable channels to each SDN-IoT switch to avoid potential network congestion. In particular, we consider how to exploit POC assignment in the SDN-IoT. However, our investigation reveals that the conventional fixed POC assignment algorithms are not viable for the highly dynamic large-scale SDN-IoT. Therefore, in this article, we propose a novel deep-learning-based intelligent POC assignment for the wireless SDN-IoT where the IoT data traffic dynamically changes. In particular, we envision two deep-learning-based strategies to predict the future IoT traffic load and to adaptively assign POCs according to predicted traffic load, respectively. Computer-based simulation results demonstrate that with the envisioned deep learning methods carried out at the SDN-IoT controller, our proposal achieves high accuracy of traffic load prediction and quick convergence of the channel assignment process. Additionally, in contrast with the conventional POC assignment algorithms, our proposal significantly improves the network performance.

Introduction

Recently, the Internet of Things (IoT) [1] has emerged as a global network infrastructure to connect a huge number of diverse heterogeneous devices by using heterogeneous (wired/wireless) communication techniques. Due to the high mobility and wide coverage of these devices, different types of wireless radio access technologies have been widely used in IoT. However, the complexity of heterogeneous communication technologies and device infrastructures has resulted in many critical issues, such as task and space sharing among different devices, network load balance, and so forth.

To better suit the heterogeneous large-scale IoT, software defined Networking (SDN) [2] has been proposed as a novel solution to connect distributed heterogeneous devices into a centralized sharing working system. This is referred as the SDN-IoT [3, 4]. In SDN-IoT, as shown in Fig. 1, various devices are widely deployed in the sensing plane. All sensing data collected by the sensing plane are forwarded through switches in data plane and then delivered to the gateway. Using the control plane, SDN-IoT separates the network control logic from the underlying routers and switches to the central controller, which usually has high computation capacity. Thus, the controller is able to control the whole network, for example, by computing packets forwarding paths and managing the channel resource, while the switches in the data plane are just responsible for forwarding the massive IoT data.

The wireless SDN-IoT meets the requirement that huge numbers of heterogeneous devices work cooperatively in one large-scale network. However, with the increasing number of devices, the traffic load of switches may become significantly heavy, and multiple channels need to be appropriately allocated to links. To address these issues, partially overlapping channels assignment (POCA) may be a promising technique to assure communication quality of service (QoS) in the wireless network [5]. In addition, heterogeneous devices have different policies in data sensing and collection, which result in uneven bursty traffic arrival at switches. For such situations, how to adaptively assign channels to fit such bursty traffic becomes a significant research challenge.

As depicted in Fig. 2, conventional POCA algorithms [5–10] only focus on the current (i.e., last time slot) traffic load, which works well with the assumption of stable traffic loads. However, once the traffic pattern suddenly changes in the next time slot, a channel in good condition may be assigned to a wrong link with a heavy load in the last time slot, but idle in the next one. On the other hand, the link with high load in the next time slot may be assigned a channel in poor condition due to its idle state in the last time slot. The wrong channel assignment decision significantly wastes the channel resource, and this leads to decreased network throughput and high packet loss rate.

Digital Object Identifier: 10.1109/MCOM.2018.1701227

The authors are with Tohoku University.

Furthermore, conventional POCA algorithms do not consider the dynamics of traffic patterns and perform the channel management in a static manner. Thus, they carry out the channel assignment only once in the initial part in a distributed fashion, and have the problems of high computation complexity and long iteration time. In a dynamic IoT environment, the channels need to be reassigned once the network traffic condition changes. However, when the channel assignment is being processed, the network transmission must be suspended until new channels are available. Thus, the high computation complexity and long iteration time of conventional algorithms may lead to long suspension time of network transmission.

Therefore, in order to improve network transmission performance, two main problems need to be solved. One is the dynamic traffic load prediction problem, and the other is the problem of how to achieve the quick convergence of a channel assignment algorithm to reduce transmission suspension time.

In this vein, a deep-learning-based intelligent POCA algorithm is proposed in this article. Our proposal consists of two parts. First, we utilize deep learning to predict the future traffic loads of switches according to the history of traffic data. Then the central controller of SDN-IoT can further adopt the deep learning technique to allocate the channel resource according to the traffic load prediction. The centralized control mechanism in SDN-IoT can ensure the traffic load prediction accuracy, while the high computation ability of the central controller in SDN expedites the POCA process.

The remainder of the article is organized as follows. The following section surveys the relevant research work. Then we present the formal problem statement and also describe our considered deep learning system model. Following that, we describe our proposed deep-learning-based channel assignment. The performance of our proposal is then evaluated. The final section concludes the article.

Related Research Work

POCA has been demonstrated as an efficient technique to achieve better performance than the non-overlapping channel assignment method [7]. With the high performance of partially overlapping channels (POCs), a game-theory-based POCA algorithm was proposed in our earlier work [9]. Furthermore, in order to deal with the dynamic network topology, an anti-coordination game-based POCA (ACPOCA) was proposed [10]. In the ACPOCA, the channel assignment algorithm can converge fast because of the adoption of only local information in the game. However, recent studies [11, 12] show that the convergence speed can be further improved by using the powerful deep learning approach, which is presented in this work. The deep learning technique is a widely used intelligent machine learning technique first proposed by Hinton *et al.* in [13]. Our previous work in [12] envisioned the first proof of concept of using deep learning architectures for substantially improving the heterogeneous network traffic control. The survey in [14] further presented the wide usage of deep learning in the networking area. The work in [11] employs software defined routers (SDRs) to improve the routing strategy with deep learning. However, those previous works all consider a baseline routing method for training the deep learning structures in a supervised manner. In [15], a novel deep-learning-based traffic control strategy was proposed that can learn by itself to quickly adapt to the changing network environment without the baseline algorithm and human supervision.

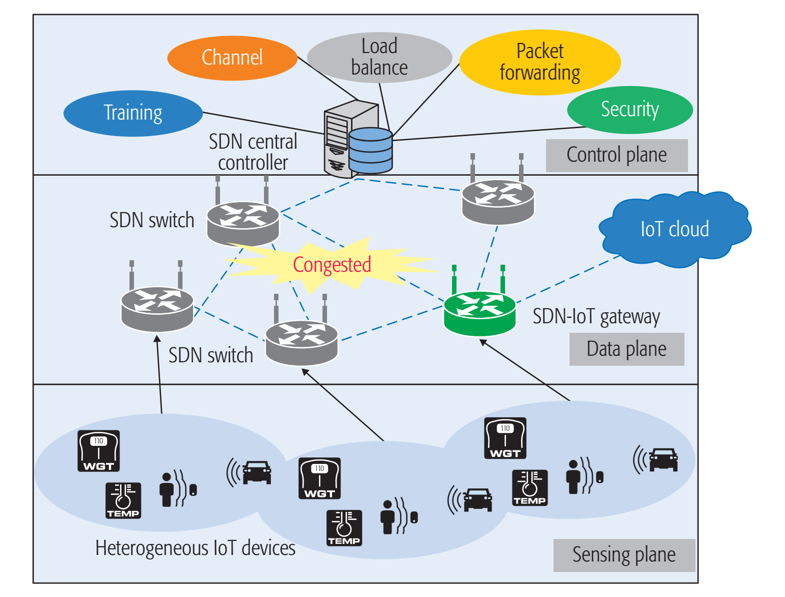


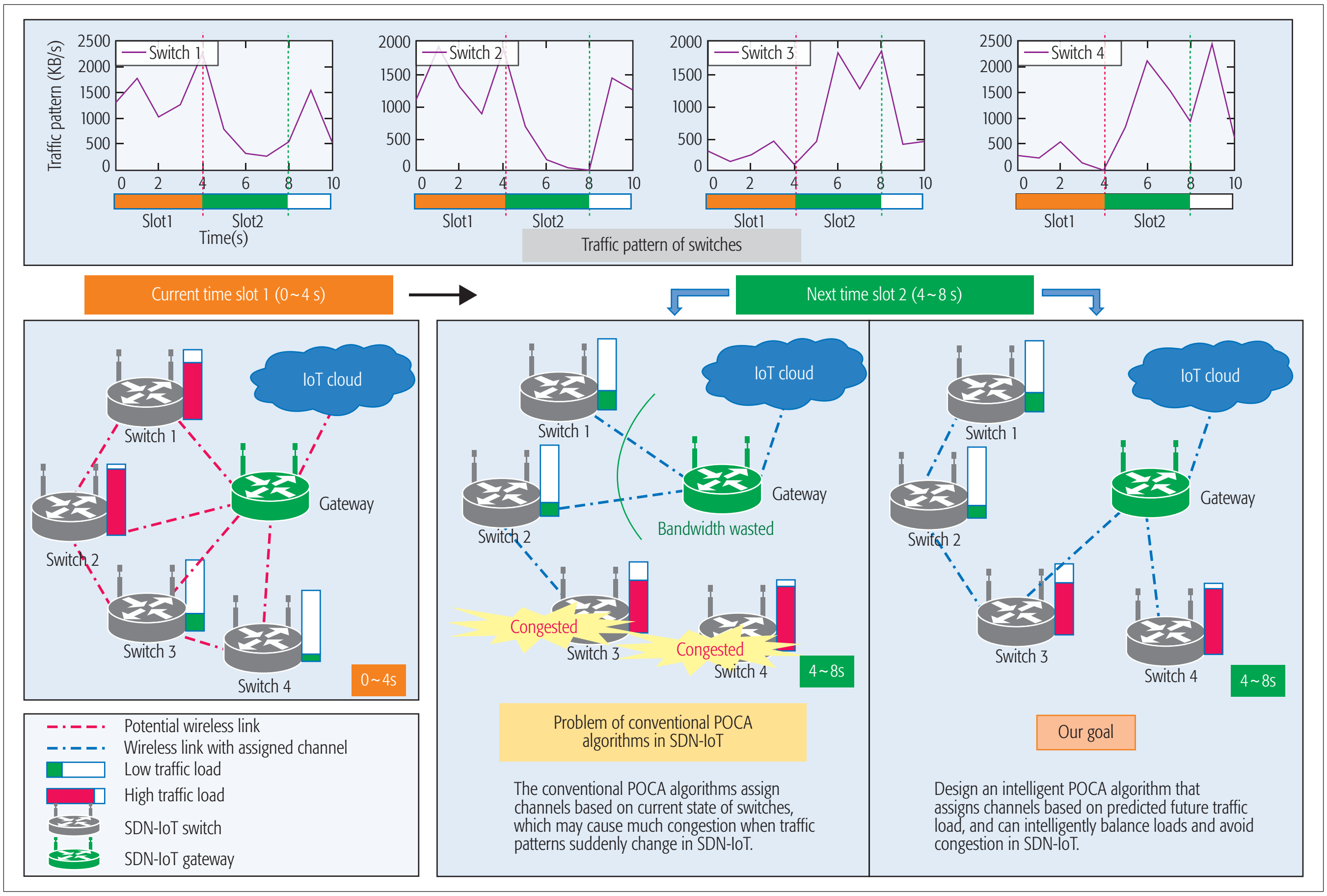
Figure 1. Architecture of the considered SDN-IoT.

However, the aforementioned research works consider deep learning in network packet forwarding. To the best of our knowledge, for the first time, in this article, we exploit deep learning in intelligent POCA.

Problem Statement and the Considered Deep Learning System Model

In this section, we first present the problem, and then formulate our considered deep learning system model. Our considered wireless SDN-IoT is constructed with a heterogeneous structure, as shown in Fig. 1, whereby POCA can be used to improve performance, whereas in SDN-IoT, there are a huge number of heterogeneous devices deployed in the sensing plane. They sense and collect data using different policies and send the sensed data to the switches through complex data flows, which causes highly dynamic and massive IoT data traffic and uneven traffic load distribution in SDN-IoT.

The conventional POCA algorithms considered channel assignment based on current traffic loads. As shown in Fig. 2, with conventional POCA algorithms, good condition channels are assigned to the switches with current high traffic loads. However, with the highly dynamic data traffic in SDN-IoT, the traffic load may suddenly change in the next time slot. At this moment, according to conventional POCA algorithms, the channels in good condition may be assigned to the links that experienced heavy traffic loads in the last time slot, but become idle in the following time slot. This leads to a significant capacity waste



Traffic pattern (KB/s)
Switch 1
Switch 2
Switch 3
Switch 4
2500
2000
1500
1000
500
0
0 2 4 6 8 10
Slot1
Slot2
Time(s)
Traffic pattern of switches
Current time slot 1 (0~4 s)
Next time slot 2 (4~8 s)
IoT cloud
Gateway
Switch 1
Switch 2
Switch 3
Switch 4
0~4s
Bandwidth wasted
Congested
Congested
4~8s
Problem of conventional POCA algorithms in SDN-IoT
The conventional POCA algorithms assign channels based on current state of switches, which may cause much congestion when traffic patterns suddenly change in SDN-IoT.
Our goal
Design an intelligent POCA algorithm that assigns channels based on predicted future traffic load, and can intelligently balance loads and avoid congestion in SDN-IoT.
Potential wireless link
Wireless link with assigned channel
Low traffic load
High traffic load
SDN-IoT switch
SDN-IoT gateway

Figure 2. The problem of conventional POCA algorithms and our research goal.

of channel resource and can further lead to serious traffic congestion in switches that have heavy traffic loads. Therefore, the conventional POCA algorithms are not suitable for the highly dynamic SDN-IoT.

Can we design a novel POCA algorithm in which the intelligent controller can adaptively assign best channels to suit the dynamic network data traffic? To design such an intelligent algorithm, how to suit the dynamic traffic load is the main problem. Thus, a method of future traffic load prediction is proposed. Then the POCs can be intelligently assigned to the suitable SDN-IoT switches based on the predicted traffic loads. In order to explain this problem more clearly, we analyze the data generation mechanisms of devices in the sensing plane of SDN-IoT. Considering the heterogeneous structure of SDN-IoT, we divide the sensing devices into two groups according to their sensing manners: periodic and event-driven sensing. For periodic sensing IoT devices, such as temperature, humidity, and light sensing devices, they sense data periodically, and then integrate and transmit them to the central controller. Moreover, even the same type of IoT devices may have different policies (e.g., sensing cycle, volume of sensing data). For example, a kind of temperature sensing IoT device may collect 3 kB temperature once every 30 s, while a humidity sensing device may collect 10 kB humidity data once a minute. Then, in the data plane of the SDN-IoT, each switch collects data from the IoT devices, and sends them to the gateway with multihop transmission. Thus, the data traffic flows of switches are constructed with collected data traffic coming from the IoT devices in the sensing plane and relayed data traffic coming from other SDN-IoT switches. Such mixed traffic makes the traffic load prediction much more complex, which is impossible to estimate using existing algorithms.

Furthermore, in conventional POCA algorithms, the transmission suspension time caused by the channel assignment process (iteration times) is overlooked. However, in the highly dynamic SDN-IoT, the channels frequently change, and the transmission suspension time during the channels assignment process may significantly degrade the network performance and cannot simply be ignored. Thus, how to decrease the iteration times during the channel assignment is an important problem.

In the next sections, a deep-learning-based approach is proposed to solve both the traffic load prediction and channel assignment problems. We utilize the deep convolutional neural network (deep-CNN) as our deep learning structure. As shown in Fig. 3, the deep-CNN consists of two main components: feature extraction and classification. The feature extraction component constructed of multiple convolution layers is adopted to filter the features of input. Considering the possible feature loss, the pooling layers used to decrease the feature size are omitted in this research. After extracting the features, the classification process is carried out to give the final output through the classification component,

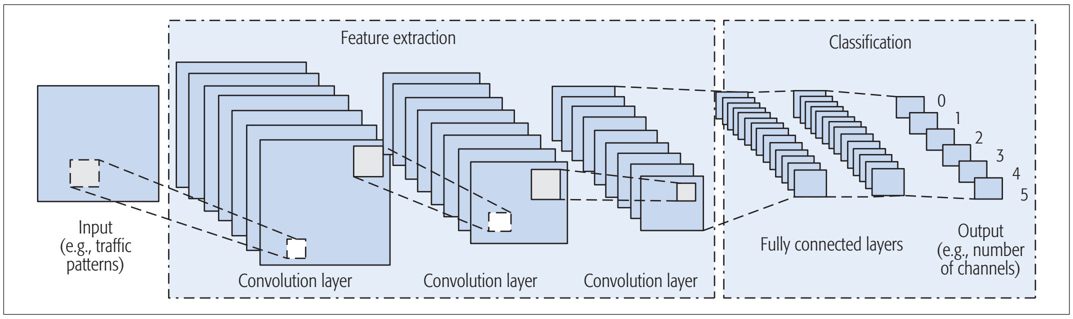


Figure 3. The considered deep convolutional neural network model.

which is constructed of many fully connected layers. This part will utilize the extracted features of the input to form an *N*-dimensional vector, which represents the output of the final classification.

The Deep-Learning-Based Intelligent POCA Algorithm

To solve the POCA problem in SDN-IoT, a deep-learning-based intelligent approach is proposed in this research. Our proposal consists of two deep learning processes. The first one is an online deep-learning-based traffic load prediction method, which uses past IoT traffic patterns to predict future traffic loads. The second process is executed based on the predicted future traffic to adaptively assign suitable channels to links of SDN-IoT switches. This adaptive channel assignment is conducted by offline deep learning using the training data generated by running the existing channel assignment algorithm in [10]. The entire process is illustrated in Fig. 4. The details of the process are provided in the remainder of this section.

Cold Start State

Prior to the intelligent channel assignment, the initial training data of the deep learning process needs to be collected. Before enough training data is collected, the network is permitted to run the conventional POCA algorithm [10]. We refer to this initial training data collection process as the cold start state, which includes the updating phase and offline training phase.

Updating Phase: In the updating phase, there are two steps: conventional channel assignment and data collection. In the first step, all switches do channel assignment with a conventional POCA algorithm such as ACPOCA [10]. For the active links assigned to channels, each switch records the assigned channel numbers of their connected links and the traffic loads in the last time slot as a set of training data. In each updating phase, switches collect and transmit the new set of training data to the central controller. The central controller formats the received traffic loads of all switches as a decimal vector, and assigns the channel numbers of all links as a collection of binary vectors. This kind of data format is better for use in the offline training phase. The updating phase is executed periodically during the packet forwarding process. After running the updating phase many times, enough data are collected for the next offline training process.

Offline Training: In the offline training phase, the collected traffic loads and assigned channels are formatted as training data to train the deep-CNN introduced earlier. Because of the large scale of the network, it is difficult to use only one deep-CNN to train all switches at a time [12]. Hence, we use several deep-CNNs to separate the computation tasks, and each deep-CNN is in charge of just one link. The trained weight matrices of each switch are recorded as $\{WM_{i,j} | i \leq M, j \leq E_{max}\}$, where M denotes the number of switches and E_{max} denotes the maximum number of active links of each switch. For each deep-CNN, to better fit the deep-CNN structure, we format each group of training data as input and output vectors. We utilize tl^i_k to denote the traffic load in time slot k of switch i. Then the total traffic loads of all switches can be donated as $tl_k = \{tl^1_k, tl^2_k , ..., tl^M_k\}$. To further improve the training performance, we convert the decimal number of the assigned channel to binary vector format. The maximum number of available channels is denoted as C_{max}, and the binary channel vector is formatted as $L_c = \{l_1, l_2, ..., l_{C_{max-1}}, l_{C_{max}} | l \in \{0, 1\}$. This kind of binary vector means that if channel i is assigned to this link, the value of the ith element in the vector is 1, otherwise 0. Therefore, the whole training data can be simply formatted as $(x_{input}, y_{output}) = (t_{lk}, L_c)$ and utilized in the training process. The offline training is conducted only in the cold start state. In the offline training, the stable POC assignment weight matrices are obtained from the trained deep-CNNs by employing training data from existing channel assignment algorithms. This is distinguished from the weight matrices obtained in the online training phase of intelligent running state, which is used to predict future traffic loads. We call the deep-CNN weight matrices gotten in the cold start state as trained POC assignment weight matrices (AWMs). Then the trained AWMs are saved and set as active in the updating phase to

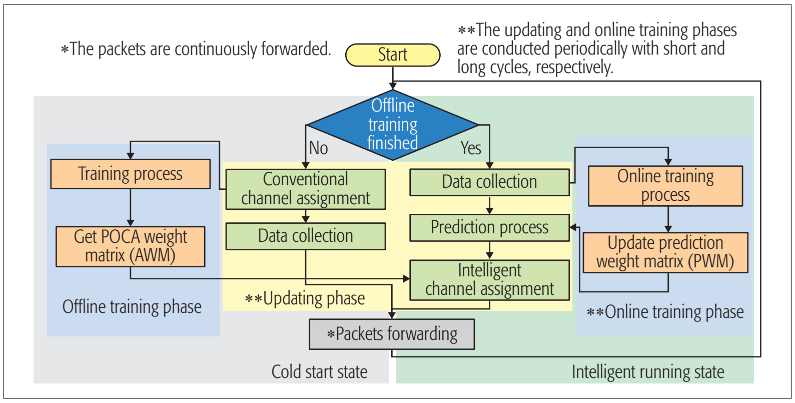

*The packets are continuously forwarded.
Start
**The updating and online training phases are conducted periodically with short and long cycles, respectively.
Offline training finished
No
Yes
Training process
Get POCA weight matrix (AWM)
Offline training phase
Conventional channel assignment
Data collection
**Updating phase
Data collection
Prediction process
Intelligent channel assignment
Online training process
Update prediction weight matrix (PWM)
**Online training phase
*Packets forwarding
Cold start state
Intelligent running state

Figure 4. The proposed deep-learning-based intelligent POCA algorithm.

The online training phase is necessary for adjusting the deep-CNNs to adapt to the new environment. Specifically, each switch continuously records the traffic load data, and the training phase is processed periodically with the newly collected training data. Therefore, the PWMs are periodically adjusted.

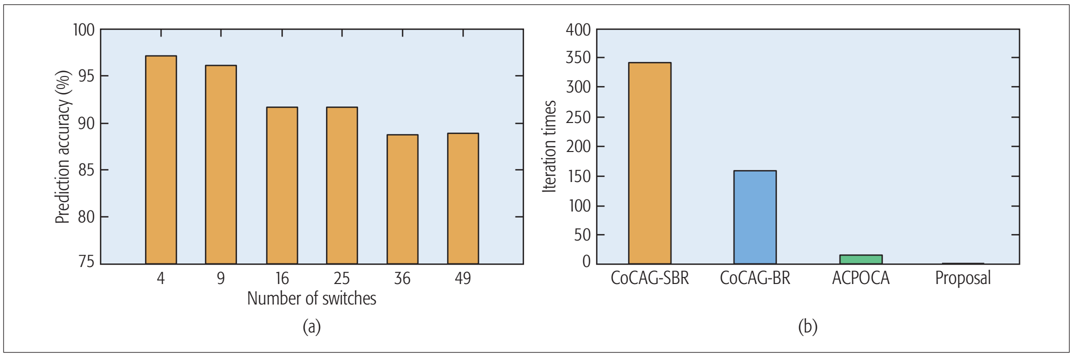


Figure 5. The prediction accuracy of our proposal and iteration times of POCA algorithms: a) the prediction accuracy with different numbers of SDN-IoT switches of our proposal; b) the iteration times of our proposal and conventional POCA algorithms.

perform channel assignment during the intelligent running state.

Intelligent Running State

After the offline training is finished, the network goes into intelligent running state. In this state, instead of the complex conventional POCA algorithm, the central controller can quickly and intelligently assign channels to links with trained AWMs based on the predicted future traffic loads.

Updating Phase: Similar to the updating phase in the cold start state, the updating phase of intelligent running state runs periodically. However, different from the cold start state, there are three steps in the updating phase with intelligent running state. In the first step, the switches record the traffic loads of the last *N* time slots. The traffic load of switch *i* in the last time slot *k* is recorded as tl^i_k. Then the past traffic loads TL^i of switch *i* are formatted as a length-*N* vector, $TL^i = \{tl^i_k, tl^i_{k-1}, \ldots, tl^i_{k-N+1}\}$. Therefore, the central controller collects and formats the traffic load series of all switches as traffic load matrix, $TL = \{TL^1, TL^2, \ldots, TL^M\}$.

After the data collection step, the deep-learning-based future traffic prediction is processed. In this step, the prediction weight matrices (PWMs) of each deep-CNN obtained in the online training phase are adopted to predict the future traffic loads. The prediction process is just a simple forward propagation process to activate the output. The output of all deep-CNNs is summarized and recorded as TLP_{k+1} = {tlp1k+1, tlp2k+1, ..., tlpMk+1}.

Finally, the central controller conducts the adaptive deep-learning-based channel assignment based on the predicted traffic loads TLP_{k+1}. With the corresponding trained AWM for each link, TLP_{k+1} is used as input to trigger the forward propagation process and obtain the activated output, which is a binary vector. The central controller further reconverts the binary vector into a decimal channel number and assigns the corresponding channel to this link.

Online Training: In the online training phase of the intelligent running state, we also use the deep-CNN as our deep learning structure. In this phase, we use the training data collected in the updating phase to train the deep-CNN to get PWMs. Similar to the offline training process, we also separate the training tasks and distribute them to several deep-CNNs to alleviate the computation load. Each deep-CNN is only used to predict the traffic load of one switch. Thus, the central controller only uses the future traffic load of one switch as the output of the corresponding deep-CNN. For example, the training data of deep-CNN of switch *i* is $(x_{input}, y_{output}) = (TL, tl^i_{k+1})$. Then the central controller trains all the deep-CNNs to obtain all the stable PWMs.

If the generation policy of the input traffic always acts according to a certain pattern, the training and prediction process based on only the existing training data is reasonable. However, in the practical IoT environment, the generation policy of the input traffic may change because of some reasons, such as SDN-IoT switch/IoT device failure or new sensing tasks assigned to existing devices. In such situations, the training process should also be adjusted accordingly. The online training phase is necessary for adjusting the deep-CNNs to adapt to the new environment. Specifically, each switch continuously records the traffic load data, and the training phase is processed periodically with the newly collected training data. Therefore, the PWMs are periodically adjusted.

Performance Evaluation

In this section, we compare the performance of our proposal with conventional POCA algorithms to assign channels to SDN-IoT links. In the conducted simulation, we consider the network to be deployed in a square place. The maximum width and length are equal and proportional to the number of switches and devices in the SDN-IoT. Consider the deployment of SDN-IoT switches to be randomly initialized, and assume that the distance between switches is larger than 10 m. Different types of devices are deployed in the control area of each switch, and the average number of devices in the coverage of each switch is 100. The percentages of event-driven sensing devices and periodical sensing devices are 20 and 80 percent, respectively. We assume that the event-driven sensing devices collect sensing data at random, while the periodic sensing devices operate with 10 different collection policies. The gateway is positioned at the farthest place from user devices. The spectrum configuration is the same as that in [9]. Multiple channels and multiple radios are assumed to be equipped on each SDN-IoT switch, while the network is operated with IEEE 802.11g wireless technology. The maximum bandwidth of each link is set to 8 Mb/s.

At first, we evaluate the prediction accuracy with different numbers of switches, and the results are shown in Fig. 5a. It can be found that the pre-

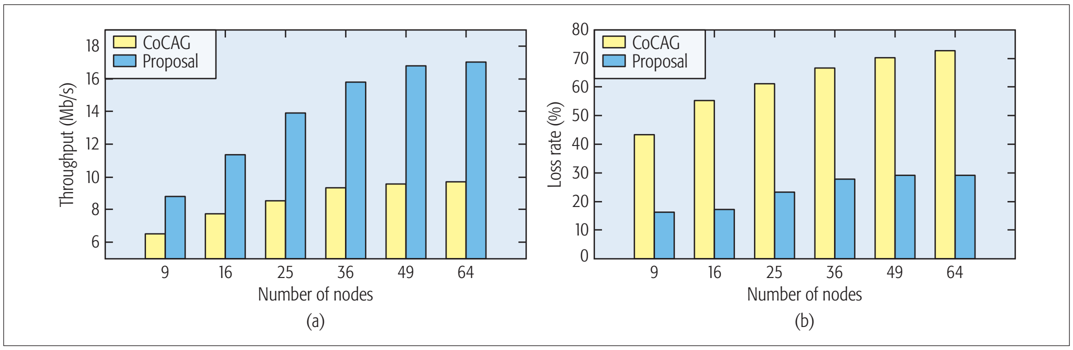


Figure 6. The network performance comparison of the proposal and conventional algorithm in terms of throughput and packets loss rate: a) the throughput compared with our proposal and CoCAG; b) the packet loss rate compared with our proposal and CoCAG.

The proposed intelligent POCA algorithm is demonstrated to converge significantly faster than the conventional POCA algorithms and has a significantly higher channel assignment accuracy to deal with the dynamic traffic of SDN-IoT. Further simulation results show that our proposal significantly outperforms the conventional POCA algorithms in terms of network throughput and packet loss rate.

diction accuracy slightly decreases with increasing network scale. However, even when the network has more than 60 switches, the prediction accuracy can still achieve almost 90 percent, which is tolerable in our proposal.

In Fig. 5b, we compare the iteration times (convergence time) of our proposal with conventional algorithms, that is, Cooperative Channel Assignment Game with Best Response (CoCAG-BR), Cooperative Channel Assignment Game with Smoothed Better Response (CoCAG-SBR) [8], and ACPOCA. From the result, we can notice that the iteration time of our proposal is always 1, which significantly outperforms the conventional algorithms. This superior performance happens since our proposal gets the suitable channel of each link through the forward propagation process, which only needs to run once. Less iteration time means faster convergence time of the channel assignment process. Therefore, our proposal can achieve the fastest channel assignment and decrease transmission suspension time among the compared POCA methods.

With deep learning, the fast convergence of the POCA algorithm leads to less transmission suspension time and decreased transmission delay. Furthermore, the proposed deep-learning-based traffic prediction can intelligently measure future congestion and help POCA to adaptively balance channel resources. Then we compare the network performance, such as throughput and packet loss rate, of our proposal with conventional CoCAG [8].} In this part, we execute the channel assignment in our simulation once every 10 s. During the channel assignment process, all data transmissions are paused. Figures 6a and 6b demonstrate the throughput and packet loss rate comparison of our proposal with the benchmark CoCAG method when the number of SDN-IoT switches is increased from 9 to 64. It can be found that in both results, our proposal outperforms the conventional algorithms. However, both the throughput and packet loss rate slightly increase with the increasing network scale.

Conclusion

Due to the rapid increase of sensing data and quick response requirement in the IoT environment, high-speed transmission becomes extremely important. As POCs are widely used in various wireless networks to increase network capacity, assigning suitable POCs in the IoT becomes an emerging topic for high-speed transmission. However, the high dynamics of traffic loads render the conventional fixed POCA algorithm out of application. Currently, SDN-IoT is proposed to improve transmission quality. Considering the high computation capacity of SDN-IoT, we propose a novel deep-learning-based POCA algorithm to intelligently allocate channels to each link in the wireless SDN-IoT. In our proposal, the future traffic load is first predicted based on historical traffic load through deep-CNNs, and then the SDN-IoT controller conducts POCA according to the predicted future traffic loads with deep learning technology. The proposed intelligent POCA algorithm is demonstrated to converge significantly faster than the conventional POCA algorithms and has a significantly higher channel assignment accuracy to deal with the dynamic traffic of SDN-IoT. Further simulation results show that our proposal significantly outperforms the conventional POCA algorithms in terms of network throughput and packet loss rate.

References

[1] C. Brewster *et al.*, "IoT in Agriculture: Designing a Europe-Wide Large-Scale Pilot," *IEEE Commun. Mag.*, vol. 55, no. 9, Sept. 2017, pp. 26–33.

[2] A. Rostami *et al.*, "Orchestration of RAN and Transport Networks for 5G: An SDN Approach," *IEEE Commun. Mag.*, vol. 55, no. 4, Apr. 2017, pp. 64–70.

[3] Z. Qin *et al.*, "A Software Defined Networking Architecture for the Internet-of-Things," *Proc. 2014 IEEE Network Operations and Management Symp.*, Krakow, Poland, May 2014, pp. 1–9.

[4] K. Kalkan and S. Zeadally, "Securing Internet of Things with Software Defined Networking," *IEEE Commun. Mag.* DOI: 10.1109/MCOM.2017.1700714.

[5] M. A. Hoque, X. Hong, and F. Afroz, "Multiple Radio Channel Assignment Utilizing Partially Overlapped Channels," *Proc. IEEE GLOBECOM 2009*, Honolulu, HI, Nov. 2009, pp. 1–7.

[6] W. Zhao *et al.*, "DAPA: Capacity Optimization in Wireless Networks through a Combined Design of Density of Access Points and Partially Overlapped Channel Allocation," *IEEE Trans. Vehic. Tech.*, vol. 65, no. 5, May 2015.

[7] A. Mishra *et al.*, "Partially Overlapped Channels Not Considered Harmful," *SIGMETRICS Perform. Eval. Rev.*, vol. 34, no. 1, Saint Malo, France, June 2006, pp. 63–74.

[8] H. Skalli *et al.*, "Channel Assignment Strategies for Multiradio Wireless Mesh Networks: Issues and Solutions" *IEEE Commun. Mag.*, vol. 45, no. 11, Nov. 2007, pp. 86–95.

[9] P. B. F. Duarte *et al.*, "On the Partially Overlapped Channel Assignment on Wireless Mesh Network Backbone: A Game Theoretic Approach," *IEEE JSAC*, vol. 30, no. 1, Jan. 2012, pp. 119–27.

[10] F. Tang *et al.*, "ACPOCA: Anti-Coordination Game Based Partially Overlapping Channels Assignment in Combined UAV and D2D Based Networks," *IEEE Trans. Vehic. Tech.*, June 2017, pp. 1–1.

[11] B. Mao *et al.*, "Routing or Computing? The Paradigm Shift Towards Intelligent Computer Network Packet Transmission Based on Deep Learning," *IEEE Trans. Computers*, vol. 66, no. 11, Nov. 2017, pp. 1946–60.

[12] N. Kato *et al.*, "The Deep Learning Vision for Heterogeneous Network Traffic Control: Proposal, Challenges, and Future Perspective," *IEEE Wireless Commun.*, vol. 24, no. 6, Dec. 2017, pp. 146–53.

[13] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, no. 5786, Nov. 2006, pp. 504–07.

[14] Z. M. Fadlullah *et al.*, "State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems," *IEEE Commun. Surveys & Tutorials*, vol. 19, no. 4, May 2017, pp. 2432–55.

[15] F. Tang *et al.*, "On Removing Routing Protocol from Future Wireless Networks: A Real-Time Deep Learning Approach for Intelligent Traffic Control," *IEEE Wireless Commun.*, vol. 25, no. 1, Feb. 2018, pp. 154–60. DOI: 10.1109/MWC.2017.1700244.

Biographies

Fengxiao Tang [S'15] received his B.E. degree in measurement and control technology and instrument from Wuhan University of Technology, China, in 2012 and his M.S. degree in software engineering from Central South University, Changsha, China, in 2015. Currently, he is pursuing a Ph.D. degree at the Graduate School of Information Sciences (GSIS), Tohoku University, Japan. His research interests are unmanned aerial vehicle systems, game theory optimization, and machine learning algorithms.

Bomin Mao [S'15] received his B.Sc. degree in telecommunications engineering and M.S. degree in electronics and telecommunications engineering from Xidian University, China, in 2012 and 2015, respectively. Currently, he is pursuing a Ph.D. degree at GSIS, Tohoku University. His research interests involve wireless networks, software defined networking, and quality of service, particularly with applications of machine intelligence and deep learning.

Zubair Md. Fadlullah [M'11, SM'13] is serving as an associate professor at GSIS, Tohoku University. His research interests are in the areas of 5G, smart grid, network security, intrusion detection, game theory, and quality of security service provisioning mechanisms. He was a recipient of the prestigious Dean's and President's awards from Tohoku University in March 2011, the IEEE Asia Pacific Outstanding Researcher Award in 2015, and the NEC Tokin award for research in 2016 for his outstanding contributions. He also received several best paper awards in conferences including IWCMC '09 and IEEE GLOBECOM '14.

Nei Kato [M'04, SM'05, F'13] is a professor at GSIS, Tohoku University. He became a strategic adviser to the president of Tohoku University in 2013 and the director of the Research Organization of Electrical Communication (ROEC), Tohoku University in 2015. He has been engaged in research on computer networking, wireless mobile communications, satellite communications, ad hoc, sensor, and mesh networks, smart grid, and pattern recognition. He has published more than 300 papers in peer-reviewed journals and conference proceedings. He currently serves as a Member-at-Large on the Board of Governors, IEEE Communications Society, and Chair of the IEEE Communications Society Sendai Chapter. He has also served on the IEEE Communications Society Award Committee (2015-2017), and as Editor-in-Chief of *IEEE Network* (2015–2017), Associate Editor-in-Chief of the *IEEE Internet of Things Journal* (2013–present), and an Area Editor of *IEEE Transactions on Vehicular Technology* (2014–present). He was a Distinguished Lecturer of IEEE Communications Society and IEEE Vehicular Technology Society. He is also a Fellow of the IEICE.