Dynamic Service Function Chain Embedding for NFV-Enabled IoT: A Deep Reinforcement Learning Approach

Chethan Lokesh Mariyaklla

Summer term 2020

The report explains briefly the contents of the research paper [1] about incorporating Deep Reinforcement Learning technique to solve the complex problem of SFC embedding in NFV enabled IoT networks.

1 Introduction

Modern technological developments in communication and information sectors have paved a way for making the Internet of things(IoT) to reality. To support a vast number of heterogeneous devices and applications of IoT, we have to adapt to new methods like Network Function Virtualization(NFC) where network functions are deployed as software called Virtual Network Functions(VNF) on the physical node instead of using the proprietary hardware. VNFs required by a particular application are arranged in a chain to form Service Function Chain(SFC), computing, and networking resources allocated to these VNFs on the substrate net which is termed as SFC embedding.

IoT network applications differ in their requirements of VNFs required for its processing traffic. Hence the algorithms using the static methods for SFC embedding does not gives efficient results. Authors in [1] proposes handling the dynamic NFV enabled IoT network requirement of SFC embedding with deep reinforcement learning(DRL) methods. The network and computing resources of IoT networks can be allocated adaptively using DRL based learning methods improving resource efficiency and reducing the average service latency.

2 System Model

This section will introduce some of the key concepts used to create the model and what all the factors we are considering when addressing the problem. It also mentions some of the important assumptions made to reduce the problem complexity.

2.1 SFC Embedding in NFV enabled IoT

In NFV enabled IoT network services provided by the proprietary hardware like domain name system, firewalls, load balancers, content distribution gateways, network address translation are realized by software components called VNFs. Each network application will have its own set of VNFs logically related to each other forming a chain. Through application, programming interface each of these VNFs requests resources like computing, storage, and connection for their operation, and its allocation is called SFC embedding.

For better utilization of network resources and process, more requests VNFs with large resource requests are broken down into more than one sub-VNFs called Virtual Network Function Components(VNFCs) and are connected through virtual links forming a new request topology. The new connected virtual links form VNF-FG. The process of dividing VNF into VNFCs is not discussed by the authors in paper[1].

In IoT networks, the SFC embedding process includes orchestrator, service chain, virtualization layer, and IoT network layer[1] as we see in Fig 1.

The orchestrator is responsible for running an optimization algorithm and decide on managing the SFC of an IoT service. The service function chain shows VNFCs of firewall, load balancing, encryption, packet inspection, and decryption in the mentioned order required for processing data. The resources from mobile devices to edges servers are represented in the network layer and there virtualized infrastructure is represented in the virtualization layer. For better connectivity the Base stations are connected with the edge server using fiber links.

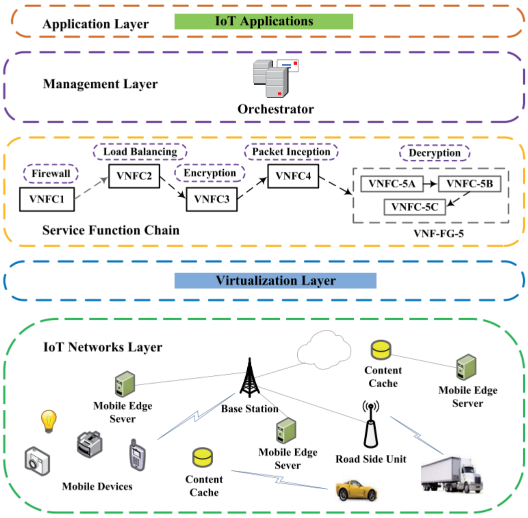

Figure 1: Service function chain embedding in the NFV-enabled IoT framework. Source: [1]

2.2 Problem Model

The VNFCs should be embedded in the substrate network considering the different arrived requests. With multiple SFC requests arriving in discrete time steps at the physical network, the layer is processed in a round-robin manner. With the SFC, VNFCs are allocated resources in order of their logical execution. The resource demand of each SFC is known only when the service request is received. Fixed embedding where the VNFCs are not preempted once allocated from the start of the service to the end of last VNFC completing its execution. Hence in NFV enabled IoT network, the response of the environment for an action from the agent depends on all the previous VNFC embedding.

IoT has a variety of applications like smart building, augmented reality, sensor networks, and many more. Hence the data stream patter in the IoT network is stochastic in the model. SFC embedding is considered with transition probabilities from one state to another making it as finite Markov property problem[1] with which we can apply Reinforcement Learning patters to efficiently find a solution.

3 Problem Formulation

In this section different parameters considered for SFC embedding in IoT network for making it as a DRL problem are discussed.

3.1 System State and Action

The main objective of SFC embedding is to reduce the average processing delay of SFC. Same IoT nodes and data transfer link resources will be shared between VNFCs of different SFC making the process more complicated. In IoT network with N nodes share computing and memory resourcesi_c, i_m connected with linkb_{ij} where $i, j \epsilon \{1, 2....N\}$ represents the state space. The resource demand of VNFCs are represented as sum of different Gaussian functions and resource consuming state of substrate networks obtained by sampling at random points. The current state s is represented in vector form s, $s = \{\{1_c, 2_c, .., N_c\}, \{1_m, 2_m, .., N_m\}, \{b_{i,j}\}\}$[1].

With information on all the available resources and the requirements the central management should choose an action of allocating resources to the VNFCs based on the current state. In action space, $a\epsilon\{1, 2, .., N\}$ agent allocate resources to one VNFC at a time and state transition is observed of the state space. With this feedback the agent moves to the next VNFC of the service chain in the next time step.

3.2 Reward

The reward metric for the DRL in allocating the resources is the sum of processing delay of VNFCs in nodes of the network and transmission delay on links in NFV infrastructure[1]. For stability, purpose authors assume that user traffic between two active VNFCs will not be rerouted in later allocations. In the DRL algorithm, the agent main purpose is to accumulate maximum rewards over the long run rather than the immediate rewards which will result in processing the VNFCs with minimum average delay.

The proposed method deployment in the NFV enabled IoT network is with NFV-MANO(Management and Orchestration Node of NFV framework). NFV-MANO receives all the requests for the resource service requests with which it will frame the service function chain of VNFCs. Considering that the learning agent is present in NFV-MANO the resources are allocated to the suitable node in the substrate network, VNFCs are allocated and change of state is observed. Depending on the reward and the environment feedback agent will allocate the next VNFC.

4 Deep Reinforcement Learning Approach

RL model is made up of 4 main components: strategy, reward function, value function, and model of the environment. RL can be classified as model-based or model-free depending on whether an environment is available or not. In a defined state space and the agent takes an action based on the value function defined and status of the current state. With this action the state space observes a transition to a new state. RL model agents take action to maximize the cumulative reward obtained in the long run rather than the immediate. Q-learning is used in the training process of the agent where Q is the action-value function. In the proposed work authors have made use of the deep neural network, experience reply, and target network methods along with Q learning to analyze the high dimensions of state space and action space generated by the NFV enabled IoT network. Experience reply helps in reducing correlated data.

5 Simulation and Results

5.1 Simulation Environment

IoT network is heterogeneous, the resources for embedding is obtained from different terminal devices and edge servers. Due to this random nature it is very difficult to realize it in simulation. For this SFC embedding algorithm authors have considered random networks, small-world networks, and BA scale-free networks as simulation topologies. Random network topology simulates the irregularity nature of IoT substrate networks which is designed based on edge probability and will be helpful in performance evaluation. Small world network[3] contains large clustering coefficients and short average path length and BA scale-free network[2] will help stimulate the practical IoT scenarios with nodes are scale-free and connectivity or node-degree distribution uses power-law form[1].

5.2 Implementation

DRL based dynamic SFC embedding algorithm[1] is implemented using Python 3.6 with TensorFlow 1.4.0 for deep neural networks implementation and NetworkX library for simulating environments discussed in 5.1. Authors have not made the implementation public, hence it has to be done using the algorithm and simulation settings mentioned by them in [1].

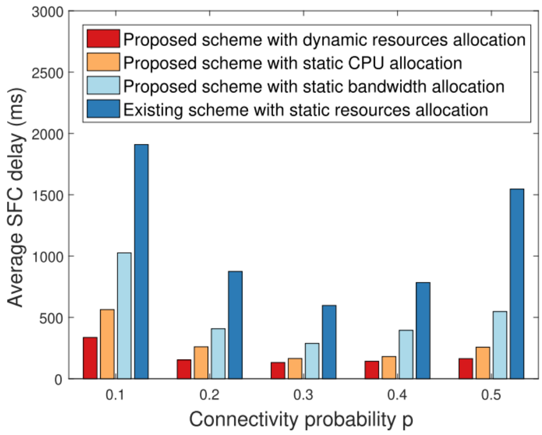

Figure 2: The average delay of a SFC in random network topologies with different node connectivity probabilities.

5.3 Results

The proposed scheme is compared with static resource allocation using a heuristic algorithm, static computing resource allocation, and static bandwidth allocation. The performance of the algorithm is as shown concerning the average delay in the random network with an increase in connectivity probability(Fig. 2). Connectivity probability is directly proportional to the number of edges hence the delay increases after 0.3. Small world network results show average delay with a count of neighboring nodes(Fig.3). An increase in nodes will increase the number of edges affecting positively at first but after the count 3 delay increases. Fig. 4 shows the evaluation of the BS scale-free network and results show the average delay of SFC embedding increases with the increase in the number of initial nodes. In all the schemes evaluated the proposed dynamic resource allocation with the DRL method gives the best results.

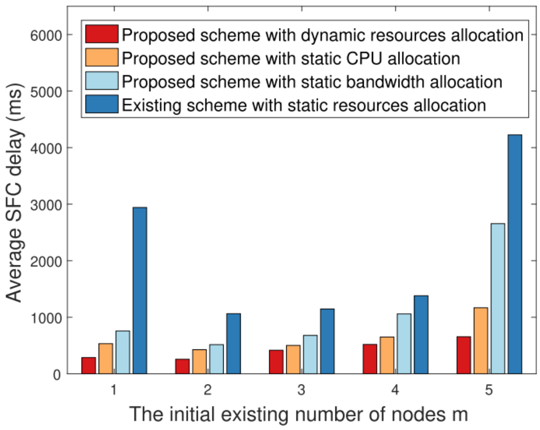

Figure 3: The average delay of a SFC in BA scale-free network topologies with different initial existing nodes.

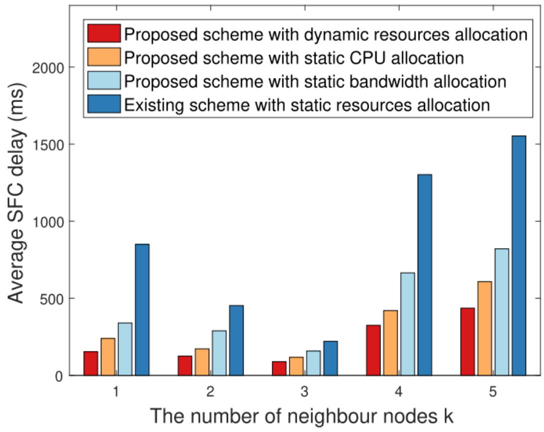

Figure 4: The average delay of a SFC in small-world network topologies with different number of neighbor nodes

6 Conclusion

For NFV enabled IOT networks the SFC embedding is a complex task because of the heterogeneous and dynamic property of IoT. Authors of [1] have proposed DRL based solution to the SFC embedding problem by dividing the VNF into small VNFCs. The state space, action space, and rewards have been discussed with experience reply and target network for DQL to enhance the convergence performance. Results have been evaluated by considering different topologies which shows the proposed scheme gives better performances.

7 Future Work

As proposed by the authors, convolutional neural networks can be adopted to NFV enabled IoT network scenario and check whether there will be any improvement in the performance of SFC embedding.

References

[1] X. Fu, F. R. Yu, J. Wang, Q. Qi, and J. Liao. “Dynamic Service Function Chain Embedding for NFV-Enabled IoT: A Deep Reinforcement Learning Approach”. In: *IEEE Transactions on Wireless Communications* (2019).

[2] A.-L. Barabási and E. Bonabeau. “Scale-free networks”. In: *Scientific american* 288.5 (2003), pp. 60–69.

[3] M. E. Newman. “The structure and function of networks”. In: *Computer Physics Communications* 147.1-2 (2002), pp. 40–45.