International Journal of Science and Research (IJSR)
ISSN: 2319-7064
ResearchGate Impact Factor (2018): 0.28 | SJIF (2019): 7.583

Low Latency Placement for Effective Fog based Infrastructure

Dr. Varsha Jotwani

Rabindranath Tagore University, Bhopal, India

Abstract: ***With the emerging technology in world wide web as our day to day activities are dependent on various computing factors, leading to the great significance the way applications are developed and deployed. With that volume of data and its impact which is increasing exponentially, for effective network utilization which is affected so there is a need of logical approach. A various IOT application comprises of running in a coordinated way. Internet of things are basically now a part of modern technological world ,with the increase volume of data in our application there is increase IOT technology devices which make daily work in a easier way. Due to complexity of network structure it is not convenient to implement centralized cloud system because it delays to handle the congestion of network. Hence for disperse network fog computing which is used for functioning of data which is on the verge of the network where the disperse network process the data. It generate NP hard problem which is related to fog devices in which placement of application module is great challenge so there is need of better approach for the placement of application module in fog infrastructure by reducing the latency of application.***

Keywords: Latency, fog computing, host machine, virtual function

1. Introduction

Fog computing is closed feature of the cloud services which is very near to the edge of network. Geographically distributed network nodes are utilized by fog computing to process the data and take the decision very quickly. The distinct devices such as router, gateways (smart gateways), edge servers, access points are applied as the fog nodes to process the data at the edge [1].

The reason for latency reduction of the traditional cloud computing architecture comprises of two forms networking and data. From the networking aspect the traffic move towards the data centre from each subsystem of large scale IOT system, due to data deluge the network suffer traffic overload which delay in transferring of data. Additionally, data itself an impediment for computing and networking. In Internet of things communication and information technology are highly influenced by low latency and high reliability of transferring of data ranging from wireless based system, cellular devices ,video and cloud based services.

Fog computing purpose is to provide extra feature to the Cloud services and its services to the edge of network thus increases the need of application segment which are having attributes of latency sensitive and functioning of real-time data and dispatching is quite challenging one. In this new kind of prototype, computing of data is fully based around the

Cloud sites and the network based elements which manages the prerequisite of Quality of Service (QoS).

There is a great combination between the fog and cloud, especially when it comes to indulging of the needs of latency sensitive applications. However, the devices nearer to the edge of network like routers, access points ,gateways etc. are generally not having that potentially strong computation which could host all the units of an application or in the IoT ecosystem, and hence the objectives needs to be develop in a way which keeps these constraints in mind, i.e., emphasize from fog layer towards the cloud and tries to place the first segment on the available resources on fog layer, thereafter repeating the same procedure towards the Cloud.

The reasons for small latency sensitivity of the standard cloud computing framework are divided into two forms that is data and networking. From the networking point of view the traffic diverted towards the data centre from each segment of huge scale of IOT system. The networks near the centre can therefore get affected by the traffic overload due to data diversion, which result in causing a network related including critical delay. For many IOT application cloud based framework cannot ensure the about the latency sensitivity requirement. Pre processing concept is involved to cut down the data. This is however, attained by eliminating the redundant data which contains an unavoidable kind of latent semantic and context generated from the data source. Virtual network function is a kind of network functions which able to get converted into software application.

Virtual network function control specific network operation that use numerous virtual machine that work in the top of hardware networking communication service. The goal for introducing of this methodology to overcome of numerous problem which affects network services like cost of energy, investment of funds, which is required to increase the working of hardware based appliances.

Network function virtualization faces the challenging aspect in which main task to place virtual network function to virtual machine that is in the fog environment and take out the best outcome of the whole scenario with the minimum workload on the network. Network function is implemented in fog module to get the optimum result.

Latency in itself is interruption or suspension in building a connection, it depend on various parameters which affect in placing communication, reason might be distance, weather, hardware configuration of hosting devices and users. Efficiency of a network is checked by the total latency generated in the connection. The whole network will get fail if it goes beyond a certain point. Proposed method include to bring down latency and which make technology applicable for constructing high definition technology, for conducting remote surgery, low- latency video streaming ,as well as to generate application in a potential way .It is beyond doubt that there would be need of further research to meet the challenges related to the evolving Fog-Cloud Architecture. The new paradigm requests a change in the way applications are developed and deployed, and to fill this gap, we introduce and formulate a strategy for efficient allocation of application modules in a combined Fog-Cloud paradigm, the main aim of which is to provide an efficient utilization of network resources and to minimize application latency. In this paper, we present a different approach to the problem from the application deployment perspective, which aims for the ultimate benefit of various players in the IoT ecosystem. The main objective is to provide an optimum use of network resources and leesen the latency of application. The new avenue requests a change in the traditional approach, procedure applications are constructed and deployed and to fill this gap to introduce and formulate a strategy for effectively allocation of application modules in a fog framework. Problem which will find in that one is described in this paper that there is requirement to reduce the latency for the fog application module in cloud based framework and second problem to optimum use of resources in the network. The paper is arranged as follow by the related work, problem description, and its proposed solution.

2. Related Work

Network Function Virtualization (NFV) develop as network framework and is an adequate technology in the networking area thus there have been lots of research is going on in this area. Researchers are continuously working to obtain the design or implement new techniques with only one objective to make this forthcoming technology in more efficient way.

In literature, several emerging studies reviewing the different area of virtual network function researches, e.g. scaling, allocation, task scheduling, placement, edge-based models, cloud-based models, latency optimizations. The challenge which is taken from [2], and has been well described in the paper. The paper has been described in proper way for better understanding by removing the complexity and making it into the simpler one to get it. The research works vary from defining the architecture [3] to assessing its confirmation in context of Internet of things [4]. In [5], authors suggest it research that to enable internet of technology fog computing is one of the important aspect. Researchers [6] suggested developing fog computing framework which comprises features of energy efficient and its management. The work done by the researchers [7] depicts the problem of workload allocation from an energy consumption point of view and [8] go through it into maximizing the service allocation in combined with fog cloud framework. These features work collectively and increase the efficiency over current infrastructure to develop and explore the full development of the new existence of application triggered by internet of technology and smart technology scenario. Comparatively to packet-switched network the problem of virtualized network function services in optical data centre network is more complicated one and this resultant into additional limitation to optical data centre [9].

3. System Framework

Network function virtualization is latest framework and is surrounded by initial problem is from [2], and it has been well described in the paper. The paper has been well described with figures and results thus making it simple and easy to understand. In our paper we bring a novel approach to solve the integer linear programming problem by applying feature of polynomial time to make it more result oriented. In this paper we are applying the same parameters which used by [2], here virtual network function are linked with the host computers and further users are connected to virtual network functions to use the network. To minimized the latency virtual network devices are linked with the host system.

Specification

Variable	Description
n_o	number of virtual network(num=1,2,.....n)
h_o	Number of virtual network hosting machine(ho=1,2,.....h)
U_i	Number of Users(ui=1,2.....Ui)
N_t	Total number of virtual network functions
H_m	Total number of virtual network functions hosting machine
U_i	Total number of users
Cap	Maximum capacity of hosting device
R_v	Requirement of virtual network function
Maxi	Maximum latency a virtual network function i can tolerate
L_a	Latency between the user of the ni virtual network function in case that virtual network function is located at hj.

Where n_v denotes virtual network function and h_o denotes number of hosting devices where $n_v >> h_o$. The set of users are represented as U= $\{u_1,u_2,u_3,....u_k\}$,the set of virtual network functions are represented as $N_v=\{n_1,n_2,n_3....n_i\}$ and $H_m==\{h_1,h_2,h_3....h_j\}$ as the set of hosting device. Every hosting device has its own capacity C_{ap} and similarly virtual network function have the requirement(R). Max latency is the maximum latency value a virtual network function can support. Latency matrix is designed to place the value for those latencies and is defined by L_a where latency is between user of the n_i ,virtual network function is placed at hosting machine h_j. Here x_{ij} =1, means that number of virtual network is assigned to host devices h_o, otherwise it will be 0.

4. Issues and Challenges

Latency is that problem which results into interruption in establishing connection, this problem is affected by the various factors like weather, material and hardware used and their respective users. For a whole network to function

efficiently it is necessary to keep the total latency minimal as lesser the latency, better services and connection will be generated. The paper main purpose is to reduce the latency generated by the newly develop connection in topology. The whole procedure will be done by creating virtual network functions to that hosting device which gives lesser latency for the topology. This can be defined as a problem related to assignment are represented by the set of users which is represented by U_i , group of host machine virtual network function will be represented as H_m ,set of virtual network function N_v and a matrix of latency is 1,in this whole procedure we have to determine that appropriate allocation is required to match the overall network function in which it is expected to reduce the latency from all users U. This can be defined as a problem related to assignment that reduces the overall expected end to end latency matrix 1, now we have to determine the proper allocation of all network functions that reduces the overall expected two end latencies from all users to virtual network functions which are allotted or capacities of all the hosting devices gets filled. Number of allocation decides about the numerous factors like the necessity of virtual network functions, the host device capacity and mainly on latency between the hosting device and virtual network functions. The main goal of our protocol is to reduce the end to end latency value which is given by equation

$$mini \sum_{n \in Nv} \sum_{ho \in Hm} xij\ Lij \quad (1)$$

Subject to

$$\sum_{num \in Nv} xij * Rv < Ch, \forall ho \in H \quad (2)$$

$$\sum_{ho \in Hm} xij\ Lij < MLi, \forall num \in N \quad (3)$$

$$\sum_{ho} xij = 1, \forall\ num \in N \quad (4)$$

In First equation, servers which having sufficient capacity for virtual network can make it place easily. There is one limitation with the above equation which clears that if there is not sufficient capacity then virtual function can't be allocated.

In the second equation, it is required that latency perceptive virtual network function will not break the rule of the major latency necessity from their users, which ensure to check that whether end to end machine having a sufficient capacity to host virtual machine or not .

Third equation ensures that at least only one time there is allocation of virtual network to hosting machine, in fact hosting machine can host to two virtual network functions.

The main drawback of the above equation was related to allocation of the virtual network. If the end to end device doesn't have enough capacity to host a virtual network function or a virtual network function requirement is ability of hosting device, but two virtual network functions can establish a connection with anyone of the hosting device. Third equation assure that all the virtual network functions get their place once in their time, two hosting machine cannot be connected simultaneously with one virtual machine.

There is allocation constraint in this model which is drawback of these above formulation which said all virtual network functions should establish connection with at least one host device which fails in certain circumstances and thus protocol fails. In the above case hosting devices cannot connect to second virtual network function as it doesn't have the efficient capacity with both the virtual network function,thus model will break down. So to resolve this problem we will apply mixed integer linear programming protocol to find the maximum number of virtual network functions that can be linked maximal to the hosting machine.

$$\sum_{ienum} \sum_{ho \epsilon H} xij = V \quad (5)$$

$$\sum_{ho \epsilon H} xij \leq 1, \forall num \in N \quad (6)$$

Where 'V' is sum of virtual network functions that can be connected in maximize way and next equation is added one virtual function must connect to all one of the hosting machine.V is calculated by maximizing and getting the optimum result through equation 2, 3 and 6. The above Integer linear programming is also stated as NP hard problem and can be resolved by Integer Linear programming solver tool and this research based problem is very general form.

5. Proposed Algorithm

Stable matching algorithm will reduce the latency by pairing end to end machine with virtual network function node, here will adopt a concept of mathematical matrices which is based on preference. Both virtual network function and end to end machine will be given preference and those having lower latency connection wise they will be kept in priority and this procedure carry forward for other end to end machine and virtual network function respectively. If the hosting machine is on the priority list virtual network function node wants to build the connection and in similar way if virtual network function node is at top of the priority list then the end to end machine will like to connect accordingly. The algorithm runs till stability could not be achieved and it terminates, either when all the virtual network function nodes are linked to the hosting machine satisfying their matching preference with each other.

Procedure 1:

1) Matching of virtual network function and end to end machine until a fixed state has been achieved.
2) Begin with all virtual network function VNF (n_o)ϵ N_t and hosting machine (h_o) ϵ H_m .
3) Begin with an overall latency and count as 0.
4) Virtual network functions have necessity and maximum latency and end to end machine have capacity.
5) While (In this scenario if there exist a free virtual network function VNF(n) who has not made the pair with hosting machine h_o and calculation is less then M)do
6) H_o is a priority based hosting machine.
7) If (H_o is free and competency constraint and latency constraint is satisfied)then (n_o,H_o) become fixed

8) Increment the value of Calculation, Calculation = Calculation+1
9) Update the competency concept = Competency of H_o – Requirement of n_o
10) Increment the Overall latency = Overall Latency + latency between n_o & h_o.
11) Else(few match (n',H_o) already exist
12) If (h prefers n_o to n_o') then
13) (n_o,H_o) become fixed
14) n' become free
15) Calculation remain same
16) On updating will get the competency + need of n'- need of n.
17) Update overall latency = overall latency – latency between n' and h + latency between n & h.
18) Else
19) (n_o',H_o) remain fixed.
20) End If
21) End While
22) Print Overall Latency
23) End of method

In beginning initialization will be done with all virtual network function and hosting machine are available and overall latency will be count as nil. The algorithm will run until certain condition and all met as shown in line 5, then a virtual network function if the condition is specified in line 7, are met than the virtual network function and hosting machine is engaged. The calculation, competency and overall latency are then edited. Then from line 13, if the hosting machine prefer to choose virtual network function n_o over the currently engaged n_o', the hosting device will be engaged with n_o and n' will get free. Again calculation, competency and overall latency are updated , if the hosting machine doesn't give preference to the selected virtual network function n_o over the currently occupied n_o', then the pair remains engaged. The proposed procedure in worst case has a complexity of θ(n*m) where n is the number of virtual functions and m is the number of host device while $n_o >> h_o$.

6. Simulation Result and Evaluation

The integer linear programming concept has been applied in Gurobi. In the branch of optimization, the research problem consider as problem of assignment. In simulation, there is a requirement to find out the matching of virtual network function with hosting machine and through this assignment will determine point to point latency. We are taking ten different runs for particular situation. For input the data, the value which will apply in which end device capacity would be taken virtual circuit function necessity and a maximum latency of virtual circuit function and latency between the virtual circuit and end machine. For latencies between the virtual circuit function and the end machine, we take random number between 15-40 as the latency depends on various aspects such as distance, the material used and the performance of end machine and virtual circuit function. In the same manner will apply instance value between 40-80, requirements and a maximum latency of virtual network function have also been taken in random form between 1-15 and 35-60 respectively. For all kind of simulation figures presented in this paper, begin with 20 virtual network function and 5 hosting devices.Here we change both the number of host machine and virtual network function as it start working on more scenario and to verify the function and efficiency of the proposed algorithm. So in the end the different instances that are implemented 20, 30, 50 and 100.For virtual network function random number is 5,10,15,20 for host machine which are then implemented to form different instances and implement them to compare outcome for both Gurobi and heuristic approach.

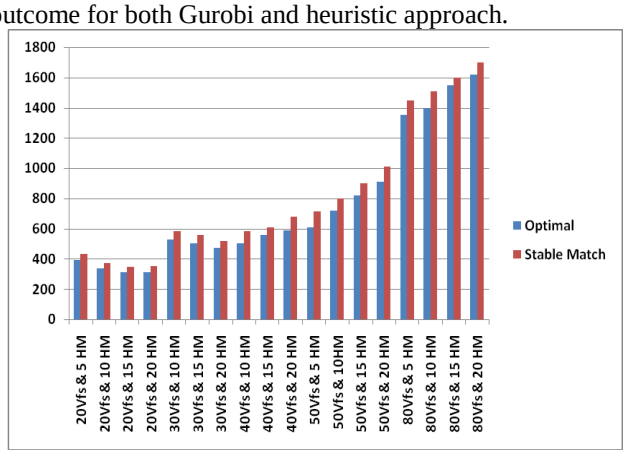

Figure 1: Comparison chart of Latency Outcome between optimal and stability match

Table 1: Time complexity for different instances of Virtual network function

Virtual Network Function	Hosting Machine	Optimal Match	Stability Match
20	5	4.49	3.54
	10	6.19	4.45
	15	7.69	5.79
	20	8.61	7.36
30	10	3.17	2.12
	15	2.91	2.07
	20	5.86	4.91
40	10	6.03	4.31
	15	7.15	5.91
	20	8.23	6.22
80	10	7.81	7.11
	15	7.41	5.52
	20	7.89	6.45

Figure 1. depicts the differences between the integer linear programming given by Gurobi and the outcome by our proposed approach for various instances for virtual nodes and end to end device. In this we have taken all the results from various combinations that we applied as an input value and it is clear proposed approach is functioning well and is giving comparable result.

Table 1. represent the time complexity where we have taken 20, 30, 40 and virtual nodes respectively and different types of end to end machine that is hosting device .From these we concluded that our algorithm is near to optimal match that means it range from 8% to 9%.

7. Conclusion and Future Work

In this paper, algorithmic procedure has been adopted to solve an assignment problem to minimize the latency. Problem which is defined in research paper [1]had some

technical aspect issues, thus it was somewhat generalized to overcome problem through it .Our proposed algorithm is to get the result of stable matching and to do comparison between integer linear programming results .Gurobi solver is used for solving the integer linear programming related problem. On performing experiment it is clear that heuristic approach is working effectively which is giving result between 8% to 9% which is more than optimal latency, thus proposed algorithmic approach is solving the problem in polynomial time. In future, there is a need to bring the modification in framework of assignment related problem, in which virtual network function automatically gets assigned to hosting machine

References

[1] Aazam, M. and Huh, E.N., 2014, August. Fog computing and smart gateway based communication for cloud of things. In Future Internet of Things and Cloud (FiCloud), 2014 International Conference on (pp. 464-470). IEEE.
[2] R. Cziva and D. P. Pezaros, "On the Latency Benefits of Edge NFV," 2017 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), Beijing, 2017, pp. 105-106.doi: 10.1109/ANCS.2017.23
[3] I. Stojmenovic, "Fog computing: A Cloud to the ground support for smart things and machine-to-machine networks," 2014 Australasian Telecommunication Networks and Applications Conference, ATNAC 2014, pp. 117–122, 2015.
[4] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the Suitability of Fog Computing in the Context of Internet of Things," IEEE Transactions on Cloud Computing, vol. pp, no. 99, pp. 1–1, 2015.
[5] M. Yannuzzi, R. Milito, R. Serral-Graci`a, D. Montero, and M. Nemirovsky, "Key ingredients in an IoT recipe: Fog Computing, Cloud computing, and more Fog Computing," in 2014 IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD). IEEE, 2014, pp. 325–329.
[6] M. A. Al Faruque and K. Vatanparvar, "Energy Management-as-a-Service Over Fog Computing Platform," IEEE Internet of Things Journal, vol. 3, no. 2, pp. 161–169, 2016.
[7] R. Deng, R. Lu, C. Lai, and T. H. Luan, "Towards power consumption delay,Tradeoff by workload allocation in Cloud-Fog Computing," in 2015 IEEE International Conference on Communications (ICC), June 2015, pp. 3909–3914.
[8] V. B. C. Souza, W. Ramrez, X. Masip-Bruin, E. Marn-Tordera, G. Ren, and G. Tashakor, "Handling service allocation in combined Fog-Cloud scenarios," in 2016 IEEE International Conference on Communications (ICC), May 2016, pp. 1–5.
[9] V. Nikam, J. Gross and A. Rostami, "vNF service chaining in optical data center networks," 2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Berlin, 2017, pp. 1-7. doi: 10.1109/NFV-SDN.2017.8169845