A Dynamic Algorithm for Computational Offloading in Fog Computing

Tri Nguyen Dang, S. M. Ahsan Kazami, Choong Seon Hong
Department of Computer Science and Engineering, Kyung Hee University, 446-701, Korea
Email: {trind, ahsankazmi, cshong}@khu.ac.kr

Abstract

The objective of this research article is to address the problem of computational offloading in Fog Computing (FOG). Similar to Cloud Computing, FOG provides data, compute, storage and application services to end-users. With Fog node, many external devices can offloading, perform computation and save energy. In this paper, we propose a dynamic algorithm to obtain a scheme that supports computational offloading among Fog users (FUEs). By applying the proposed distributed model to FOG, it enhances the performance and achieves fast convergence. First, we analyze the structural property of the problem and propose the optimization problem. We then design a dynamic computational offloading mechanism that can achieve convergence and lead to the optimal solution. Numerical results show that the effectiveness of our scheme in terms of convergence speed and achieving the optimal solution.

1. INTRODUCTION

The popularity and vast acceptance of Internet of Things (IoT) [1] has envisioned the idea of overloading tasks to Cloud Server and achieving the stability of network. A large number of IoT devices are being deployed daily. Furthermore, it will keep increasing every day which also increases the data traffic in network significantly. This means the current deployment and approach for Cloud computing cannot satisfy the ever increasing demands of the IoT. Therefore, Fog Computing has been introduced in [2], [3], [4], [5]. FOG is a distributed computing infrastructure [3]. The idea is that one or multitude of end-users collaborates with each other to carry out substantial amount of storage, computation, communication and management [5]. In Cloud computing, if end-user needed to process data, it will be transferred to Cloud server for processing. But under practical situation, a lot of IoT device can be connected to the Internet, which will increase the demand and traffic to the Cloud server [1]. This can cause two problems, first the probability that demand will be lost is high and second Cloud sever may not be able to serve the demand due to overloading. The Fog computing bring a novel paradigm that data will be analyzed and processed in milliseconds by using edge devices or Fog node in the network.

2. NETWORK MODEL

In this section, we introduce the system model of computational offloading via Fog computing. In the Fig. 2, inside the physical Fog node can virtualize number of Virtual Fog Node (VFN) included Computation and Storage.

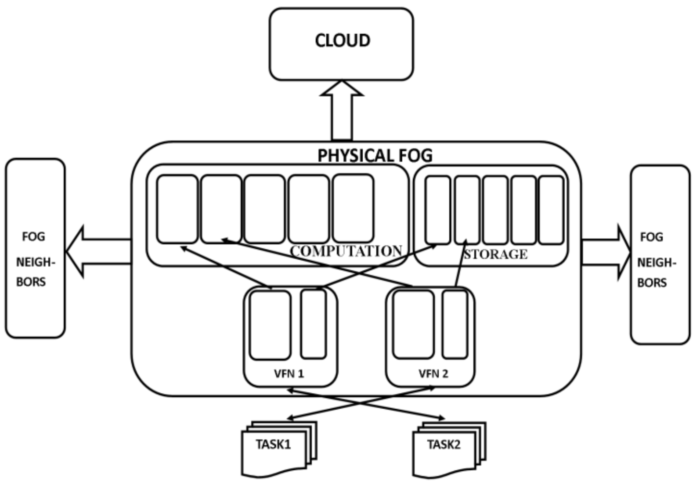

Figure 1: System model

We consider a set of N mobile devices $\mathcal{N} = \{1, 2 \ldots, N\}$. For each user, $n \in N$ has a computation intensive task I_n that needs to be completed. Such that $I_n = \{S_n, C_n^r\}$ in which S_n the size of input data and C_n^r is the numbers of CPU cycles required to execute that task. For example user 3 has a task $I_n = \{8, 2\}$, it means that the size of data input is 8*(MB)* and the number of CPU cycles required for execution is 2(*GHz).* Consequently, vector $\mathcal{I} = \{I_n : \forall n \in N\}$ represents all tasks to be completed of $\mathcal{N}$. On the other side, we consider there is exists a set of K virtual Fog nodes $\mathcal{K} = \{1, 2 \ldots, K\}$ which are available for execution of any application from the FOG users. For each $k \in \mathcal{K}$, we already know about required resource information's (e.g., Memory, Storage, and CPU cycles). Let $k = \{M_k, H_k, C_k\}$ where M_k is the memory size, H_k is storage capacity, and C_k is the computation capability. For example $\{4,16,4\}_5$, mean user *5* has the memory size $M_5 = 4(GB)$, capacity of storage $H_5 = 16(GB)$, and the numbers of CPU cycles $C_5 = 4(GHz)$. To guarantee performance of offloading model, we assuming that any user $k \in \mathcal{K}$ is strong enough to execute the entire task in $\mathcal{I}$. Furthermore different device differ by the capability and performance. The computation time can be denoted as $T_k^n = \dfrac{C_n^r}{C_k}$. Consequently, we formulated the optimization problem as follows:

$$\underset{x}{\text{minimize}} : \sum_{k=1}^{K} T_k^n x_{n,k}$$

$$\text{subject to}:$$

$$\sum_{k=1}^{K} x_{n,k} \le 1, \forall n \in N$$

$$\sum_{n=1}^{N} x_{n,m} = 1, \forall m \in M$$

The optimization is combinatorial then we using constraint relaxation and transform to convex optimization. To solve this problem, we are using augmented Lagrangian with augmented coefficient ρ. Algorithm 1 states the proposal solution in detail.

3. EVALUATION

In this section, we provide simulation results from our proposed Algorithm 1. In this algorithm, we first randomly matched any entries of set N and K. We then loop at most H step to update the objective value and variable x, whether *(n, k)* is matched or not. The algorithm will be stop if the time is greater than predefined time or the objective value has converged.

Algorithm 1: Minimize Computation Offloading Time (MCOT)

1. Initial information for any $n \in \mathcal{N}$ and $k \in \mathcal{K}$
2. Randomly match between $n \in \mathcal{N}$ and $k \in \mathcal{K}$
3. For each time slot $t \in \{1,\ 2,\ \ldots,\ H\}$

$$x_n^{k+1} = \arg\min_{x \in \mathbb{R}^N} \{T_n^k x_n^{i,} + y_n^{kT}(x_n - z^k) + \frac{\rho}{2} \| x_n - z^k \|_2^2\}$$

$$z^{k+1)} = \frac{1}{N} \sum_{n=1}^{N} (x_n^{k+1} + \frac{1}{\rho} y_n^k)$$

4. Update the objective value.
5. **If** $T(x^k) = T(x^{k+1})$
6. Break and return results.

For simulation setup, our setting of constraint is provided in the follow table:

Notation	Value
S	Uniform[120, 420] KB
C	Uniform[100,1000]Megacycles
C_n	{0.5,0.8,1.0}GHz
C_k	{10.0,15.0,20.0} GHz
K	15
N	10

Table 1: Simulation information

In simulation, we analyze the algorithm by Python and had shown the numerical results. We have used three different algorithms for performance comparison: Locally processing mean the system working without offloading,

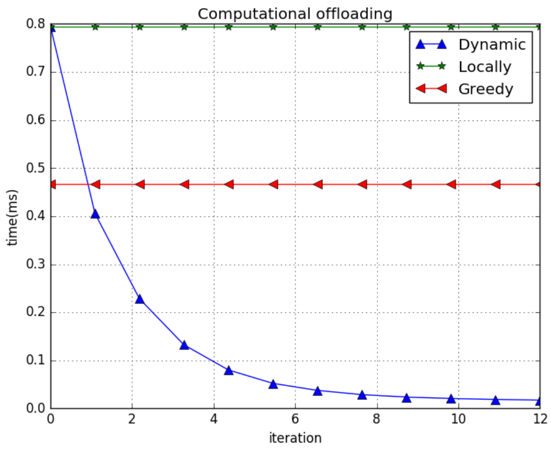

Figure 2: Simulation result

Greedy offload denotes the computation offloading using greedy algorithm, and the last one is our proposed dynamic offloading.

The simulation result shown that the proposed algorithm has achieved minimum computation time and convergence is extremely fast under 10 iteration. Furthermore, our proposals have shown that the objective value is minimal and better than all others which was our main goal. The system information when we ran this simulation, CPU is *Intel Core i3-4150* 3.5(*GHz)*, and the capacity of RAM is 8.0*(GB).*

4. CONCLUSION

Computational offloading can help devices save energy, time, and reduce the data traffic in the network. This study shows us that by applying a dynamic offload algorithm to determine whether the task will be allocated or not depending upon network condition can significantly enhance the performance. Simulations have shown that proposed scheme significantly enhances the performance and achieves very fast convergence. Finally, a detailed study into the actual operating environment is required, as a future course, to validate our proposal and its performance

5. ACKNOWLEDGEMENT

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (B0190-16-2017, Resilient/Fault-Tolerant Autonomic Networking Based on Physicality, Relationship and Service Semantic of IoT Devices) *Dr. CS Hong is the corresponding author.

REFERENCES

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," in *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347-2376, Fourthquarter 2015.

[2] Stojmenovic, "Fog computing: A cloud to the ground support for smart things and machine-to-machine networks," *Telecommunication Networks and Applications Conference (ATNAC), 2014 Australasian*, Southbank, VIC, 2014, pp. 117-122.

[3] S. Yi, Z. Hao, Z. Qin and Q. Li, "Fog Computing: Platform and Applications," *Hot Topics in Web Systems and Technologies (HotWeb), 2015 Third IEEE Workshop on*, Washington, DC, 2015, pp. 73-78.

[4] Stojmenovic and S. Wen, "The Fog computing paradigm: Scenarios and security issues," *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*, Warsaw, 2014, pp. 1-8.

[5] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," in *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347-2376, Fourthquarter 2015.