School of Informatics

Informatics Research Review
Edge Computing Offloading in Internet of Things: Experimental Designs and Configurations

s2517285
January 2024

Abstract

The abstract is a short concise outline of your project area, **of no more than 100 words**.

Date: Saturday 13$^{\text{th}}$ January, 2024

Supervisor: My IRR Tutor

1 Introduction

Internet of Things (IoT) means that the objects around people can communicate with each other and cooperate to achieve the common goals, which has great potential for both private and business uses [1]. Most tasks handled by devices in IoT tend to be delay-sensitive, which also generates a mount of data nearly 49 EB [2]. However, the IoT devices are usually limited in terms of memory, battery life, and computing power [3, 4]. Hence, it is impossible to process all the application tasks in local devices and meanwhile satisfy all the performance requirements [2]. As a consequence, computation offloading is applied to solve this problem.

In tradition, cloud computation integrates with the Internet of Things since cloud server. Unlike IoT, the storage and computation power provided centrally by the cloud server is almost unlimited, which corresponds to the disadvantages of IoT [5, 6]. Hence, cloud computing can help IoT devices to complete their computation tasks with high performance. However, for IoT devices it is an obstacle to obtain stable and acceptable network performance to reach the cloud [7]. Additionally, the cloud have to be challenged by reliability problem since the devices may fail or become inaccessible [7]. Unfortunately, the extensive scale of the resultant system, stemming from interactions with a significant number of devices, renders the rising requirements for storage capacity and computational power in subsequent processing progressively difficult to meet [7]. Therefore, edge computing is introduced to address the issues of the IoT and the cloud.

Edge Computing (EC), also named Mobile Edge Computing (MEC), provides cloud computing capabilities within the Radio Access Network close to mobile users [8]. Comparing with the cloud computing, edge computing can compute in the real-time because the edge server are closer to the users [9]. Moreover, edge computing doesn't need to upload the data to the cloud computing center and reduce the load on the network bandwidth, which lowers the cost and the network bandwidth pressure [9]. Many algorithm has been designed to optimize the task offloading problem in IoT applications based on edge computing. Nevertheless, these algorithms shown high performance have not been systematically compared to come to a conclusion about the best algorithm. One of the reasons is the experimental designs and configurations for each algorithm is extremely different. Consequently, it is difficult to get an objective comparison.

As edge computing plays a significant role in coordinating the work between IoT devices, it is necessary to design optimization algorithms to enhance the functionality of IoT through the utilization of Edge Computing characteristics. Quantities of optimization algorithms have been proposed and implemented, however, it is impossible to compare their performance due to the difference between system models (edge computing models). There are several important components can be used to build different kinds of edge computing models for IoT devices, which may have a great impact on the performance of the algorithms on the edge computing. Hence, this review will summarize the designs or configurations for the IoT application based on edge computing. It is noted that the pattern mentioned only including cloud server, edge server and objective function.

To address the problem of differences in edge computing system models that result in incomparability, this article will attempt to answer the following questions:

1. What are the main differences between different system models? How those affect the performance of the algorithms?

2. Why the designers choose such configurations? What's the pros and cons?

3. Based on questions 1 and 2, what designs or configurations should be considered when applying optimization problems?

The section 2 will focus on the cloud computing component in the EC designs. Additionally, in section 3 the full offloading and partial offloading will be discussed. Last but not least, the section 4 will study on the different objective function chosen by the EC system models.

2 Literature Review

2.1 Cloud Computing components

Though the main conception discussed in this review is edge computing, it doesn't mean that the cloud components should be excluded from the edge computing models.

2.2 Offloading Strategies

Since the IoT devices have limited computation and energy resources, they can hardly satisfied the complicated tasks required by the IoT service. Therefore, the goal of the task offloading is to gain computation capability without using more energy-cost devices [10]. The offloading strategies can be separated into two categories: full offloading and partial offloading strategies. Full offloading strategy means offloading the task all to the edge computing server or cloud server. On the contrary, the partial offloading strategy is aimed at dividing tasks into two parts, one part is executed on the local machine while the other part is offloaded to the edge or cloud server [11].

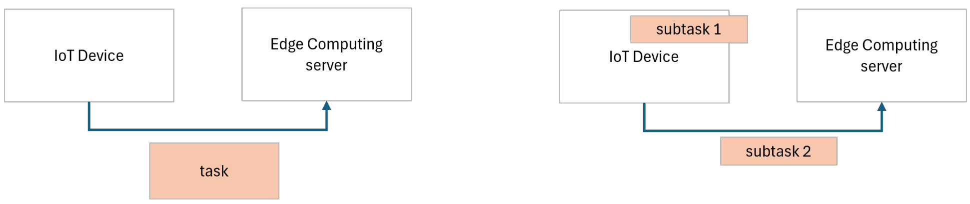

Figure 1: Full offloading strategy VS partial offloading strategy

Ning and te.al suggests that one factor that affects the choice of different strategies is the type of the applications [4]. For example, if the input data of the application is privacy information, the tasks should be partial offloaded. Another important factor that influences the strategy is the ability of offloading, due to some part of the tasks shouldn't be offloaded [4]. On the condition that the task is allowed to be divided, for the reason of optimizing user's energy conservation, partial offloading strategy has a higher priority [12]. However, the offloading becomes more complicated when partial offloading strategy is considered, for the reason that the task relevance, characteristics and segmentation have to be concentrated [13].

2.3 Objective Functions

2.4 Other Difference

3 Summary & Conclusion

References

[1] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.

[2] Jiwei Huang, Ming Wang, Yuan Wu, Ying Chen, and Xuemin Shen. Distributed offloading in overlapping areas of mobile-edge computing for internet of things. *IEEE Internet of Things Journal*, 9(15):13837–13847, 2022.

[3] Arash Heidari, Mohammad Ali Jabraeil Jamali, Nima Jafari Navimipour, and Shahin Akbarpour. Internet of things offloading: Ongoing issues, opportunities, and future challenges. *International Journal of Communication Systems*, 33(14):e4474, 2020. e4474 IJCS-19-0207.R3.

[4] Zhaolong Ning, Peiran Dong, Xiangjie Kong, and Feng Xia. A cooperative partial computation offloading scheme for mobile edge computing enabled internet of things. *IEEE Internet of Things Journal*, 6(3):4804–4814, 2019.

[5] Manuel Díaz, Cristian Martín, and Bartolomé Rubio. State-of-the-art, challenges, and open issues in the integration of internet of things and cloud computing. *Journal of Network and Computer Applications*, 67:99–117, 2016.

[6] Muhammad Asim, Yong Wang, Kezhi Wang, and Pei-Qiu Huang. A review on computational intelligence techniques in cloud and edge computing. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(6):742–763, 2020.

[7] Alessio Botta, Walter de Donato, Valerio Persico, and Antonio Pescapé. Integration of cloud computing and internet of things: A survey. *Future Generation Computer Systems*, 56:684–700, 2016.

[8] Milan Patel, Brian Naughton, Caroline Chan, Nurit Sprecher, Sadayuki Abeta, Adrian Neal, et al. Mobile-edge computing introductory technical white paper. *White paper, mobile-edge computing (MEC) industry initiative*, 29:854–864, 2014.

[9] Keyan Cao, Yefan Liu, Gongjie Meng, and Qimeng Sun. An overview on edge computing research. *IEEE Access*, 8:85714–85728, 2020.

[10] Wei Yu, Fan Liang, Xiaofei He, William Grant Hatcher, Chao Lu, Jie Lin, and Xinyu Yang. A survey on the edge computing for the internet of things. *IEEE Access*, 6:6900–6919, 2018.

[11] Chunli Ren, Guoan Zhang, Xiaohui Gu, and Yue Li. Computing offloading in vehicular edge computing networks: Full or partial offloading? In *2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC)*, volume 6, pages 693–698, 2022.

[12] Congfeng Jiang, Xiaolan Cheng, Honghao Gao, Xin Zhou, and Jian Wan. Toward computation offloading in edge computing: A survey. *IEEE Access*, 7:131543–131558, 2019.

[13] Xiaoheng Deng, Zihui Sun, Deng Li, Jie Luo, and Shaohua Wan. User-centric computation offloading for edge computing. *IEEE Internet of Things Journal*, 8(16):12559–12568, 2021.