Dynamic Resource Provisioning Through Fog Micro Datacenter

Mohammad Aazam
Computer Engineering Department
Kyung Hee University, Suwon
South Korea
aazam@ieee.org

Eui-Nam Huh
Computer Engineering Department
Kyung Hee University, Suwon
South Korea
johnhuh@khu.ac.kr

***Abstract*— Lately, pervasive and ubiquitous computing services have been under focus of not only the research community, but developers as well. Different devices generate different types of data with different frequencies. Emergency, healthcare, and latency sensitive services require real-time response. Also, it is necessary to decide what type of data is to be uploaded in the cloud, without burdening the core network and the cloud. For this purpose, Fog computing plays an important role. Fog resides between underlying IoTs and the cloud. Its purpose is to manage resources, perform data filtration, preprocessing, and security measures. For this purpose, Fog requires an effective and efficient resource management framework, which we provide in this paper. Moreover, since Fog has to deal with mobile nodes and IoTs, which involves objects and devices of different types, having a fluctuating connectivity behavior. All such types of service customers have an unpredictable relinquish probability, since any object or device can quit resource utilization at any moment. In our proposed methodology for resource estimation and management, we have taken into account these factors and formulate resource management on the basis of fluctuating relinquish probability of the customer, service type, service price, and variance of the relinquish probability. Implementation of our system was done using Java, while evaluation was done on CloudSim toolkit. The discussion and results show that these factors can help service provider estimate the right amount of resources, according to each type of service customers.**

***Index Terms*—IoT; Cloud of Things; Fog computing; Edge Computing; Micro Data Center (MDC); resource management; Fog-Smart Gateway (FSG).**

I. INTRODUCTION

Connectivity has been revolutionized with the rapidly increasing wireless sensor networks (WSNs), healthcare related services, smart phones, and other pervasive means. With the advent of Internet of Things (IoT), devices, services, and people are ubiquitously connected almost all the time, also generating a lot of data. The IoT's objective is to provide a network infrastructure with interoperable communication protocols and softwares, to allow interaction and integration of physical as well as virtual sensors, computers, smart devices, vehicles, and dumb objects like: household items, food items, medicines, etc. [1].

The backbone of IoT communication is Machine-to-Machine (M2M), although, not limited to it. In M2M, two or more machines communicate with each other directly, without human intervention. IoT enables non-communicating devices become part of Internet and communicate through data communications means, like: bar-code reader, RFID, etc. With the advancements in smart phone technology, many objects would be able to be made part of IoT, through various smart phone sensors. By this, non-intelligent nodes, known as "things" become communicating and data generating objects of IoTs.

IoT based services are gaining importance rapidly. Since 2011, number of connected devices has already exceeded the number of people on Earth. Already, connected devices have reached 9 billion and are expected to grow more rapidly and reach 24 billion by 2020 [2]. With increasing number of heterogeneous devices connected to IoT and generating data, it is no more possible for a standalone IoT to perform power and bandwidth constrained tasks efficiently. IoT and cloud computing amalgamation is becoming very important [3] [4]. There comes a situation when cloud is connected with an IoT that generates multimedia data. Visual Sensor Network or CCTV connected to cloud can be examples of such scenario. Since multimedia content consumes more processing power, storage space, and scheduling resources, it will be very important to manage them effectively and perform efficient resource management in the cloud. Other than that, mission critical and latency sensitive IoT services require a very quick response and processing. In that case, it is not feasible to communicate through distant cloud, over the Internet. Fog computing plays a very vital role in this regard [5]. Fog Computing refers to bringing networking resources near the underlying networks. It is a network between the underlying network(s) and the cloud(s). Fog Computing extends the traditional Cloud Computing paradigm to the edge of the network, enabling creation of refined and better applications or services [6]. Fog is an Edge Computing and Micro Datacenter (MDC) paradigm for IoTs and wireless sensor networks (WSNs).

In this paper, we present a service oriented resource management model for Fogs, which can help in efficient, effective, and fair management of resources for the IoTs. Our work is mainly focused on customer type based resource estimation. We have considered different traits and characteristics of customers in this regard, which makes our model more flexible and scalable.

978-1-4799-8425-1/15/$31.00 ©2015 IEEE

II. Related Work

Research on Fog computing is in its very beginning, therefore, no standard architecture is available regarding managing resource in the Fog. Already done studies mainly focus trivially on resource management in the clouds. The scenario of Fog computing or Cloud of Things (CoT) is not considered by any of the prior works.

Wang Wei et al. discuss [7] a brokerage service for reservation of requests. The authors suggest a brokerage service for on-demand reservation of resources, for IaaS clouds. Their work is limited to only on-demand jobs and they do not present anything beyond that. Park Ki-Woong et al. [8] discuss a billing system with some security features. The authors present a mutually verifiable billing system to resolve different types of disputes in future. Their work only focuses on the reliability of transactions made in purchasing and consuming resources. They do not focus on the overall resource management, specially for CoT. Rogers Owen et al. [9] present a methodology for resource allocation, but resource prediction related matters, along with service relinquishing issues are not considered. Their study is also only limited to standard cloud resource management. Yang Yichao et al. also present resource allocation algorithm, but in a simplistic way [10]. Deelman Ewa et al. present performance tradeoffs of different resource provisioning plans. They also present tradeoffs in terms of storage fee of Amazon S3 [11]. Their work does not take into account resource management tasks. Shadi Ibrahim et al. present the concept of fairness in pricing in respect of micro-economics [12], not discussing how pricing should be done for different types of services. Their work is only limited to micro-economics pricing aspect. Kan Yang et al. present [13] a dynamic auditing protocol for ensuring the integrity of stored data in the cloud. They present an auditing framework for cloud storage. Zhen Xiao et al. present [14] a resource allocation system that uses virtualization technology to dynamically allocate resources, according to the demands of the service. In their study, they present measuring the unevenness in resource utilization. IoT based environment is not considered in this study. D. Cenk Erdil, in [15], presents an approach for resource information sharing through proxies. Situations where clouds are distant and there is no direct control, proxies can be used to make resource information available to them. This study only focuses on the importance of resource information sharing. Rakpong et al. consider resource allocation in mobile cloud computing environment in their work [16]. They discuss about communication/radio resources and computing resources, but their work only focuses on decision making for coalition of resources, to increase service provider's revenue. Flavio Bonomi et al. present [6] basic architecture of Fog computing, which does not include its practical implications and resource management for IoT. Similarly, Salvatore J. Stolfo et al. present [18] data protection through Fog computing, but not going into resource management and related matters.

III. Fog Computing

Fog computing is a newly introduced paradigm, which extends the standard cloud computing to the edge. Therefore, it is also called Edge Computing. It is a Micro Datacenter (MDC), highly virtualized platform, responsible for providing computation, storage, and networking services between the end nodes in an IoT and traditional clouds [6]. In contrast to the standard cloud, which is more centralized, Fog computing is targeted for widely distributed applications. Figure 1 presents an overall architecture, where dedicated Fogs will be able to provide resources near the underlying networks or IoTs. Fog would be able to provide low latency and high quality streaming to mobile nodes and moving vehicles, through proxies and access points positioned accordingly, like, along highways and tracks. Similarly, resource and power constrained WSNs and virtual sensor networks (VSNs) would be able to take advantage from the presence of Fog. Because of being localized, i.e., residing closer to the underlying IoTs, Fog suits applications with low latency requirements, emergency and healthcare related services, video streaming, gaming, augmented reality, etc. For smart communication, Fogs are going to play an important role. Fog constitute of MDC, where processing, memory, virtual machines, and storage resources are available. Besides, Fog also contains gateway(s), able to handle the data communication in a smarter way, on the basis of the requirement of the higher level application and constraints of the underlying nodes. Such type of gateway is termed as Fog-Smart Gateway (FSG) [4], [5].

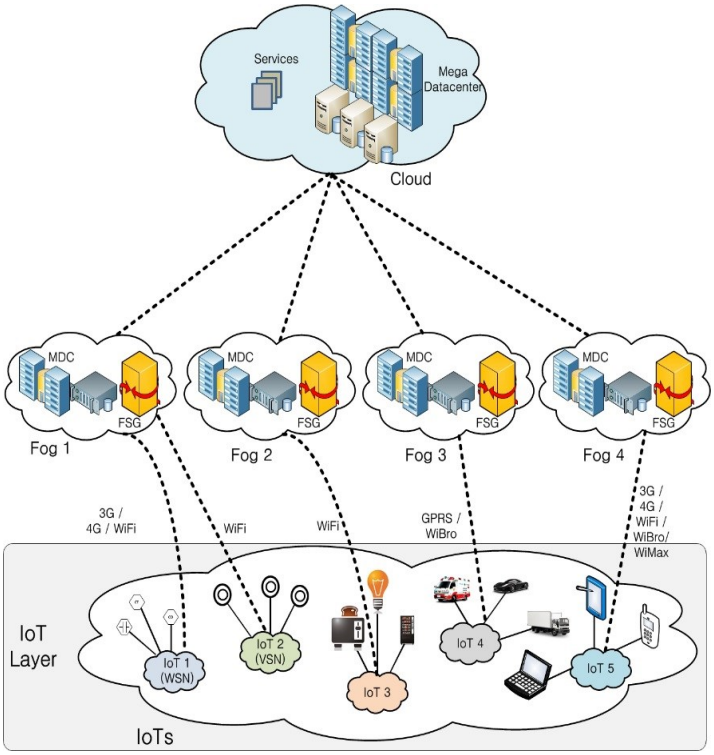

Figure 1. Fog MDC supported IoTs.

In the Cloud-Fog-IoTs architecture presented above, the underlying nodes and networks are not always physical. Virtual sensors and VSNs are also requirements for various services. Similarly, temporary storage of data, preprocessing, data security and privacy, content delivery services and other such tasks can be done easily and more efficiently in the presence of a Fog. Based on the feedback from application and depending upon the constraints of the node generating data, the Fog-Smart Gateway (FSG) decides the timings and type of data to be processed in the Fog and then sent to the cloud. FSG helps in better utilization of network and cloud resources.

Since Fog is localized, it provides low latency communication and more context awareness. Fog computing allows real-time delivery of data, specially for delay sensitive and healthcare related services. It can perform preprocessing and notify the cloud, before cloud could further adapt that data into enhanced services. With heterogeneous nodes, heterogeneous type of data would be collected. Interoperability and transcoding becomes an issue then. Fog plays a very vital role in this regard. Likewise, IoT and WSN federation, in which two or more IoTs or WSNs can be federated at one point, can be made possible through the Fog. This will allow creation of rich services.

IV. FOG'S RESOURCE MANAGEMENT MODEL

Sensors, IoT nodes, devices, and Cloud Service Customers (CSCs) contact Fog to acquire the required service(s) at best price. CSCs perform the negotiation and SLA tasks with Fog. Once the contract is agreed upon, the service is provided to the customer. In this regard, Fog not only provides services on ad hoc basis, but also, it has to estimate consumption of resources, so that they can be allocated in advance. Resource prediction allows more efficiency and fairness at the time of consumption. As mentioned, the requests can be made from objects or nodes as well as devices operated by people. Therefore, prediction and pre-allocation of resources also depend upon user's behavior and its probability of using those resources in future. For this purpose.

We formulate the estimation of required resources as:

$$\Re = \sum_{i=0}^{n}\sum_{k=0}^{x}\begin{cases}\left(\mho_i * (P_L - \sigma^2)\right) * (\Omega_L), if\ n = 0\\ \left(\mho_i * (P_L - \sigma^2)\right) * (1 - \Omega_i), if\ x = 0\\ \left(\mho_i * \left((1 - \bar{\mathrm{x}}\,(P_i(L|H)_s)\right) - \sigma^2\right) * (1 - \Omega_i)\end{cases} \quad (1)$$

$$\Re \in \{CPU, storage, memory, bandwidth\}$$

$$P_i(L|H)_s = \begin{cases}\bar{\mathrm{x}}\,(\sum_{s=0}^{n} P(L|H)_s) & if\ n > 0,\\ 0.3 & if\ n = 0\end{cases} \quad (2)$$

$$\sigma^2 = \frac{1}{n-1} * \sum_{i=1}^{n}(x_i - \bar{\mathrm{x}}\,)^2 \quad (3)$$

Where $\Re$ represents required resources, $\mho_i$ is the basic price of the requested service. In most of the cases, $\mho_i$ is decided at the time contract is being negotiated. $\bar{\mathrm{x}}\,(P_i(L|H)_s)$ is the average of service oriented relinquish probabilities of a particular customer of giving up the same resource which it has requested now. In case the customer is requesting this service for the first time, the default value set for $\bar{\mathrm{x}}\,(P_i(L|H)_s)$ is 0.3. Because, the average of low relinquish probability (0.1 to 0.5, from complete range of 0.1 to 0.9) is 0.3. For simplicity, we have categorized customers into two types, one having low (L) giving up probability and the other having high (H) giving up probability. Where,

$$0 < L \leq 0.5\ ,\ 0.5 < H \leq 1 \quad (4)$$

$$\Omega_i = \begin{cases}\bar{\mathrm{x}}\,(\bar{\mathrm{x}}\,(\sum_{k=0}^{n} P(L|H)_k)), P(L|H)_{last}) & if\ n > 0,\\ 0.3 & if\ n = 0\end{cases} \quad (5)$$

σ^2 is the variance[1] of service oriented relinquish probabilities (SOP). CSCs, specially mobile users, can have a very fluctuating behavior in utilizing resources, which may lead to deception, while making decision about resource allocation. That is why, in our model, we have taken into account variance of relinquish probabilities, which helps determining the actual behavior of each customer.

Ω represents history of overall relinquish probabilities, i.e., average overall probability (AOP). Here, it should be noted that $P_i(L|H)_s$ determines probability of that particular service which customer is requesting currently, while Ω is overall probability, including all activities a particular customer has been doing. Last activity of the user in this regard tells about its most recent probability. That is why, it is given more importance and the average is taken again, by adding last relinquish probability. In case of a new user, when there no historical data for that user, this value is set at low relinquish probability 0.3.

V. IMPLEMENTATION RESULTS AND EVALUATION

In this section, we present implementation results of our service model, along with the discussion on each result. We defined our service model through algorithm to evaluate the effectiveness in CoT business. Our main objective is to observe the influence of performance factors on the systems and test the feasibility of our method.

A. Evaluation Setup

We have considered different parameters to estimate the required resources for different types of users. Table 1 shows the

[1] http://mathworld.wolfram.com/Variance.html

setting of basic parameters. Since implementation on real test-beds limits the extent to the scale of the test-bed, which consequently makes it difficult to reproduce the result and analyze in varied scenarios, we chose simulation instead.

TABLE 1: KEY PARAMETERS' SETTING FOR EVALUATION

Parameters	Range
Default SOP	0.3
Default AOP	0.3
Relinquish probability (P)	0.1 ~ 0.9
Service Price (U_i)	100,150, 200,250,…,500
User characteristic (L or H) default	L > 0 && <= 0.5, H > 0.5 && <= 1
Variance range	0 ~ 0.16
Minimum virtual resource value (VRV)	3
Number of registered services	10

B. *Resource estimation for an absolutely new customer*

When CSCs having different traits are requesting for a particular service, the Fog has to analyze what number of resources have to be allocated for that service, based on the type of customer. For low relinquish probability CSCs, priority in resource allocation is given. For those customers, who are absolutely new and Fog has no past record for them, default probability value is used. In other words, the default case is on the assumption that new customer will be 'somewhat' loyal. That is why, relinquish probability is set to 0.3. While perfectly loyal customer would be having a probability of 0.1. Since cloud resources are precious and it is not advisable to take risk, thence, instead of assigning 0.1 probability value, we have assigned 0.3, which is the average probability of low relinquish, as explained earlier with the model. Figure 2 shows the unit of resources, we call it virtual resource value (VRV), being estimated for new customers, for different types of registered services. This unit is then mapped to actual resources (memory, CPU, storage space, etc.), according to the type of service being offered and policies of a particular CSP. For example, a USD 100 cloud storage collaboration service is more I/O intensive. It requires more CPU as well as storage space. The CSP will map 9 to level one of its resource allocation actual mappings. In case the USD 100 service is related to database queries, then only I/O is intensive, not storage, because it requires read-only process. The CSP will perform mappings accordingly. This is how different units of resources are mapped to actual resources, based on the type of service. Similarly, for a USD 500 service, 45 units of resources are reserved.

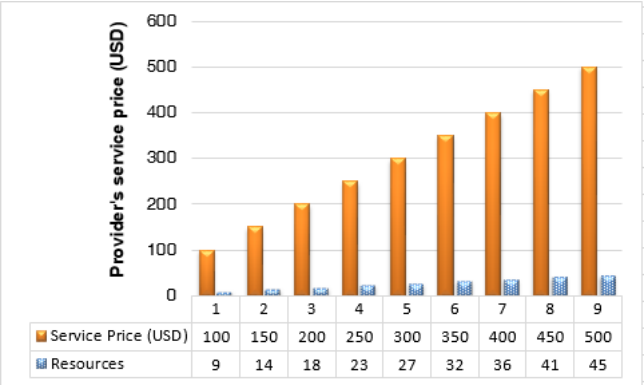

Figure 2. Resource estimation for new CSCs, for different requested services.

Illustrative Scenario:

Figure 3 shows the illustrative scenario, as an example of how mapping can be performed by the CSP, according to its resource pool and the type of service being provided. For a video on demand (VoD) service, S1, VRV 9 is mapped to corresponding resource pool level (RPL). Then according to the type of service being provided, the mapping is performed to the actual resource pool. Among the available resources for the service 1, CSP allocates 10% of CPU, 8% of memory, and data rate of 200Kbps. Storage is not required for this service, therefore, it is 0%. The guarantee of allocation of these resources is 80%, which means, at least 80% of resources from the mapping are guaranteed. This is only an example. This mapping would vary according to the type of service and available resource pool of CSP.

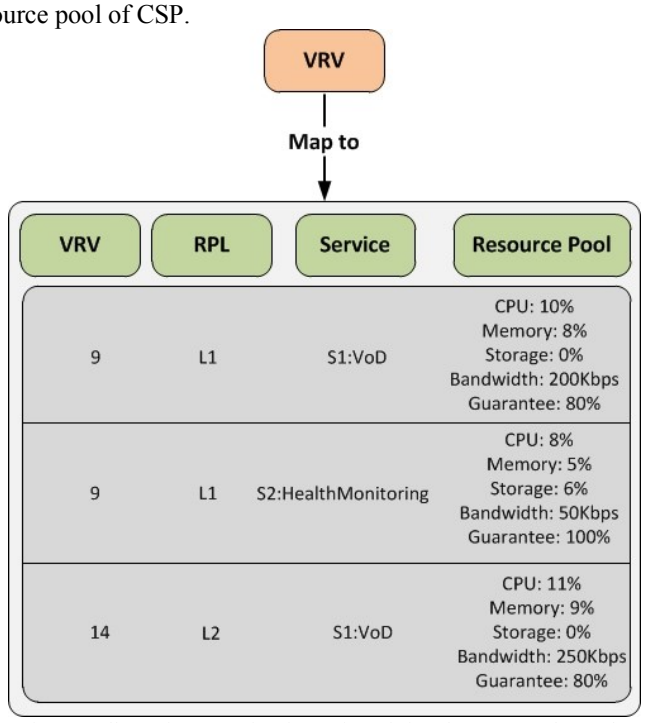

Figure 3. Illustrative scenario of mapping of virtual resource value to the resource pool, according to the type of service.

C. *Resource estimation for an existing customer, requesting service S for the first time*

In the scenario when a CSC has already been a customer of CSP before, but requested a particular service *S* for the first time, resources are estimated differently. In this case, the record of general characteristic of CSC exists, but on for service *S*, there is no historical data. Therefore, Fog allocates resources keeping in view the available record, but assuming that the CSC is going to be somewhat loyal in utilizing current service *S*. Main idea is to incorporate available historical data as much as possible, so that the CSC is dealt accordingly, with fairness and CSP and Fog have minimum possible risk.

Figure 4 shows that resources are predicted on the basis of available Average Overall Probability (AOP), keeping Service Oriented Probability (SOP) to 0.3 (somewhat loyal). In case of CSC 1, when AOP is 0.1, maximum possible resource units are allocated. For this case, 27 resources are allocated. Resources are decreased as the relinquish probability increases. For a CSC having 0.9 (90%) AOP, 3 units of resources are reserved. By this, Fog makes it sure that CSC is treated according to its reliability and Fog itself and the CSP are not deprived of the profit they deserve. Also, chances of resource wastage are minimized.

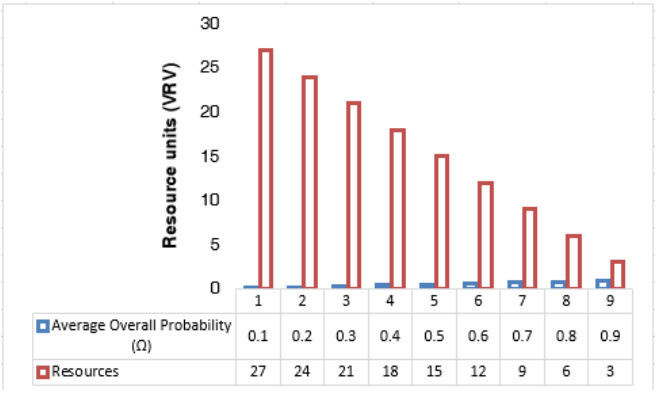

Figure 4. Resource estimation for existing CSC, requesting service *S* for the first time.

D. *Resource estimation for an existing customer*

For the returning/existing customers, Fog already has a historical record of its past activities and probabilities (AOP and SOP) with which CSC has been consuming resources. When characteristic of a particular customer is known, it is more reasonable and rational to determine and allocate resources accordingly. In this way, Fog and CSP will be able to reserve right amount of resources and would be having least number of chances to lose profit. Figure 5 shows five different types of CSCs, having different SOPs and AOPs, requesting a particular service *S*. In this example, the result is presented for service price USD 100. The unit is greater for *L* customers, while it is smaller for *H* customers, because of their behavior. Since there are more chances of an *H* customer to relinquish the service, hence, more priority and quality is provided to the more loyal customer, having *L* probability. In case of CSC 1, having SOP = 0.1 (bold font in the figure) and AOP = 0.3, 52 units (VRV) of resources are reserved for USD 100 service. In case of CSC 2, SOP = 0.2 and AOP = 0.4, 47 unit of resources are reserved. CSC 2 gets less resources as compared to CSC 1, because of its higher SOP and AOP. Comparing CSC 2 with CSC 5, both have same AOP. But CSC 5 has SOP = 0.5, therefore, it gets less resources (27). This shows that both these types of probabilities have their impact and final decision is made accordingly, which makes it sure that a CSC who has generally been loyal, but not so in case of some particular service, or vice versa, gets treated in view of that.

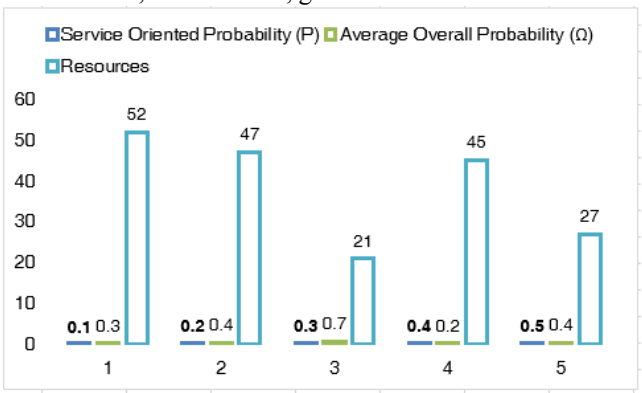

Figure 5. Resource estimation for different types of CSCs, for $100 service.

E. *Resource estimation with variable AOP variance*

As mentioned earlier that with IoT devices and mobile nodes, service relinquish probability is very fluctuating. Due to this, variance in AOP is made part of user characteristic, while determining resources. This section presents the effect of variability in AOP variance. In this part, we fixed the SOP to 0.3 and service price to $100, to assess the effect of AOP and its variance. Figure 6 shows that for case 1, when AOP is 0.4 and variance in AOP (shown in bold font) is 0.16, resource estimated for USD 100 service are 32 VRV. Case 4 and 5 having same AOP=0.1, the effect of variance is evident. For case 4, variance is 0.04. Estimated resources are 59 VRV. For case 5, having variance 0.05, the resources are decreased with the same ratio the variance increases. In this case, estimated resources are 58 VRV.

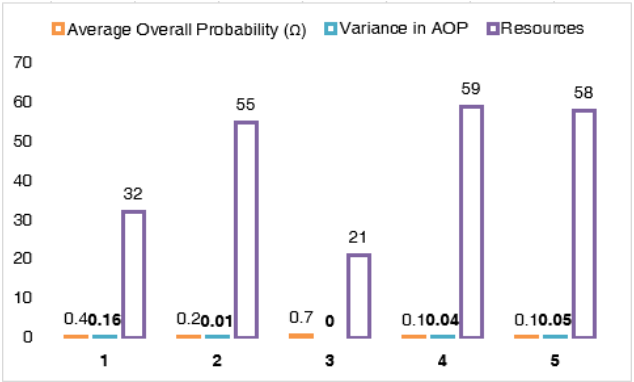

Figure 6. Effect of variance on overall resource estimation.

VI. CONCLUSION AND FUTURE WORK

Rapidly increasing IoT-based services has triggered the need of more sophisticated ways for handling heterogeneous devices, having fluctuating connectivity and data generating behavior. Energy and resource constrained IoT nodes require to be offloaded. Furthermore, healthcare, emergency, and multimedia services require quick response with minimum latency. With IoT-Cloud communication, it becomes very difficult to achieve that, having cloud reachable through a shared, unreliable core network. Resources are to be brought up closer to the nodes. Fog computing provides the solution by bringing cloud resources to the edge of the underlying IoTs and other end nodes. But with heterogeneous devices being part of IoT, it is not predictable that how much resources would be consumed and whether the requesting node, device, or sensor is going to fully utilize the resources it has requested. Due to this uncertainty, the probability of resource utilization, known as Relinquish Probability in our model, is incorporated while performing resource estimation. Our model presents user characteristic based resource management for Fog, taking into account the type of service, overall service relinquish probability, and service oriented relinquish probability. We have also included variance in relinquish probability to know the exact deviation and irregularity factor in give-up probability. This methodology helps determine the right amount of resources required, avoiding resource wastage and profit-cut for the CSP as well as the Fog itself. Every involved entity is treated rationally.

In future, we would extend our model for varied scenarios, considering monetary matters, according to the type of CSC.

ACKNOWLEDGMENT

This work was supported by the ICT R&D program of MSIP/IITP, Republic of Korea. [14-000-05-001, Smart Networking Core Technology Development]. The corresponding author is Prof. Eui-Nam Huh.

This work was also supported by the IT R&D program of MSIP/IITP [2014044078003, Development of Modularized In-Memory Virtual Desktop System Technology for High Speed Cloud Service]. The corresponding author is Prof. Eui-Nam Huh.

REFERENCES

[1] Gerd Kortuem, Fahim Kawsar, Daniel Fitton, and Vasughi Sundramoorthi, “Smart Objects and Building Blocks of Internet of Things”, IEEE Internet Computing Journal, volume 14, issue 1, pp. 44-51, Jan.-Feb., 2010.

[2] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami, “Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions”, Technical Report CLOUDS-TR-2012-2, July 2012.

[3] Mohammad Aazam, Pham Phuoc Hung, Eui-Nam Huh, “Cloud of Things: Integrating Internet of Things with Cloud Computing and the Issues Involved”, in the proceedings of 11th IEEE IBCAST, Islamabad, Pakistan, 14-18 January, 2014.

[4] Mohammad Aazam, Pham Phuoc Hung, Eui-Nam Huh, “Smart Gateway Based Communication for Cloud of Things”, In the proceedings of 9th IEEE ISSNIP, Singapore, 21-24 April, 2014.

[5] Mohammad Aazam, Eui-Nam Huh, "Fog Computing and Smart Gateway Based Communication for Cloud of Things", in the proceedings of IEEE Future Internet of Things and Cloud (FiCloud), Barcelona, Spain, 27-29 August, 2014.

[6] Flavio Bonomi, Rodolfo Milito, Jiang Zhu, Sateesh Addepalli, "Fog Computing and Its Role in the Internet of Things", in the proceedings of ACM SIGCOMM, August 17, 2012, Helsinki, Finland.

[7] Wang, Wei, et al. "Dynamic cloud resource reservation via cloud brokerage”, 33rd IEEE ICDCS 2013.

[8] Park, Ki-Woong, et al. "THEMIS: A Mutually verifiable billing system for the cloud computing environment." Services Computing, IEEE Transactions on 6.3, 300-313, 2013.

[9] Rogers, Owen, and Dave Cliff. "A financial brokerage model for cloud computing." Journal of Cloud Computing 1.1, 1-12, 2012.

[10] Yang, Yichao, et al. "A sevice-oriented broker for bulk data transfer in cloud computing.", 9th IEEE International Conference on Grid and Cooperative Computing (GCC), Nanjing, Jiangsu, China, 01-05 November, 2010.

[11] Deelman, Ewa, et al. "The cost of doing science on the cloud: the montage example." Proceedings of the 2008 ACM/IEEE conference on Supercomputing, 2008.

[12] Shadi Ibrahim, Bingsheng He, Hai Jin, “Towards Pay-As-You-Consume Cloud Computing”, IEEE International Conference on Services Computing, Washington, USA, July 4-9, 2011.

[13] Kan Yang, Xiaohua Jia, "An Efficient and Secure Dynamic Auditing Protocol for Data Storage in Cloud Computing", IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 9, September 2013.

[14] Zhen Xiao, Weijia Song, and Qi Chen, "Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment", IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 6, June 2013.

[15] D. Cenk Erdil, "Autonomic cloud resource sharing for intercloud federations", Future Generation Computer Systems vol. 29 (2013) 1700–1708.

[16] Rakpong Kaewpuang, Dusit Niyato, Ping Wang, and Ekram Hossain, “A Framework for Cooperative Resource Management in Mobile Cloud Computing”, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, Vol. 31, No. 12, December 2013.

[17] Mohammad Aazam, Eui-Nam Huh, “Inter-Cloud Architecture and Media Cloud Storage Design Considerations” proceedings of 7th IEEE CLOUD, Anchorage, Alaska, USA, 27 June – 02 July, 2014.

[18] Salvatore J. Stolfo, Malek Ben Salem, Angelos D. Keromytis, "Fog Computing: Mitigating Insider Data Theft Attacks in the Cloud", Security and Privacy Workshops (SPW), 2012 IEEE Symposium on. IEEE, 2012.