 doi: 10.1109/WCNC.2016.7564810

QoS Estimation and Selection of CSP in Oligopoly Environment for Internet of Things

Subarna Chatterjee, *Student Member*, *IEEE*, Sudip Misra, *Senior Member*, *IEEE*
School of Information Technology
Indian Institute of Technology Kharagpur
Email: {subarna, smisra}@sit.iitkgp.ernet.in

***Abstract*—This work focuses on an automated selection of Cloud Service Provider (CSP) for a naive end-user in an IoT scenario. In traditional cloud computing model, the end-users are knowledgeable about the Virtual Machines (VMs) and are technically aware of their requirements in terms of the computing cores, processing abilities, and storage requirements. In case of IoT, the users are envisioned to be widespread from naive, unsophisticated people to even objects or things who are devoid of the required knowledge and expertise. Further, in IoT technology, multiple Cloud Service Providers (CSPs) may possess the potential of serving an IoT application. Therefore, it is required for the end-user to judiciously select a single CSP based on the maximum obtainable Quality of Service (QoS) from a CSP. This work proposes an algorithm *QoS based Automated Selection of CSP* (*QASeC*) for automated selection of a CSP from a set of nominated CSPs based on the maximum achievable QoS. The work identifies and models the QoS parameters for every CSP and defines a QoS utility metric for each CSP. Based on the metric, the work proposes an optimization for selection of the appropriate CSP and the cloud gateway associated with it. From the obtained results, we infer the suitability of QASeC in real-life IoT scenarios.**

***Index Terms*—Wireless Sensor Networks, Cloud Service Provider, Oligopoly, Internet of Things, Quality of Service**

I. Introduction

With the inception of the emerging Internet of Things (IoT), it is anticipated that at the end of 2015, approximately 25 billion sensor enabled devices will be connected to the Internet [1] and by 2020 the global market of IoT is expected to observe a revenue growth of around \$300 billion [2]. Therefore, its is intuitive that a very large number of sensors and applications will be involved in the functioning of IoT, thereby leading to enormous growth of IoT users [3].

With the onset of IoT, every IoT end-user is required to undergo through the interactive Service Level Agreement (SLA) process for negotiation of the obtainable cloud services. Without the loss of generality it can be assumed that a naive end-user is not capable of analyzing the quality of a cloud service. Further, within an oligopoly environment [4] of IoT, the dynamic decision making of selection among multiple Cloud Service Providers (CSPs) is also difficult without the apriori knowledge on the Quality of Service (QoS). To address the difficulty, this work focuses on QoS based selection of CSP in an oligopolistic IoT environment for effective negotiation and analysis of the SLA in sensor-cloud based IoT platforms.

A. Motivation

The smart world of IoT is expected to be heavily oligopolistic in nature with respect to the number of CSPs [5]. Therefore, a typical IoT end-user has to undergo the phase of negotiation of cloud services with multiple CSPs and subsequently selection of one or more CSPs who can provision IoT services within the satisfactory limit. This implies, for effective negotiation and decision making, an end-user has to possess an apriori knowledge and perception on quality of cloud services. As mentioned earlier, a naive end-user is devoid of such knowledge or experience. This work addresses such a problem scenario in sensor-cloud based IoT environments.

Contemporary cloud end-users hold requirements on software (SaaS), platform (PaaS), or infrastructure (IaaS). The end-users are aware of the service requirements in terms of the computational or storage resources of the cloud units, or the strength of the computing cores allocated to the virtual machine (VMs). A typical end-user is required to undergo an interactive SLA negotiation phase in which both the parties (the end-user and the CSP) agree on the QoS of the obtainable cloud services. This necessitates the general cloud end-user to be a technical person equipped with the knowledge of the computing cores, VMs, RAMs, and processors. Further, s/he is experienced in prior estimation of resources e.g. the amount of memory, bandwidth, computation, and storage required for his/her intended applications.

On the other hand, IoT serves much wider range of beneficiaries. The end-users of IoT are the common people varying from naive, unsophisticated personnel to technically equipped IT professionals. It can be judiciously speculated that a major population of IoT users are plainly customers of sensor-based services and typically focus on obtaining the services, rather than getting into the hard technical complexities of it. The end-users enjoy the abstracted IoT services by not actually comprehending or specifying the technical details associated with the service.

B. Contribution

This work proposes an algorithm – *QoS based Automated Selection of CSP* (*QASeC*) that enables selection among multiple CSPs through a process of judgment even without possessing a definite perception or awareness of the cloud services. Assuming a set of CSPs nominate themselves for serving a particular application, the goal of QASeC is to perform QoS based service analysis of every CSP thereby, obtaining a relative ordering of the nominated CSPs. Eventually, an end-user can determine and evaluate the QoS from the scores obtained by every CSP.

The contributions of the work are multi-fold and are listed as follows:

- QASeC evaluates a CSP by mathematically modeling the obtainable QoS. The chosen parameters include average available bi-directional link service capacity, mean transmission delay, mean hop delay, mean processing delay, and mean sensor accuracy.
- The formulation of the objective of selecting a single CSP from a set of nominated CSPs involves mathematical quantification of the QoS that can be rendered by a CSP. A QoS utility metric is designed and normalized for computational convenience.
- The work also designs an optimization problem that allocates the gateway that has the highest QoS utility metric relative to the other CSPs. The formulation minimizes error or the mean square deviation of the QoS parameters.
- The work presents a detailed case study of the gateway allocation process of a single CSP in QASeC. aA multiple CSP scenario is also demonstrated and results indicate the selection of the best CSP in terms of the highest obtainable QoS.

II. Related Work

This Section discusses the prior work done in this domain so far. The SLA negotiation in generic cloud platforms have been widely explored. Wu *et al.* [6] discusses a customer driven SLA-based resource provisioning algorithm to meet the user satisfaction. The algorithm accepts end-user's quality parameters through a web based interface and dynamically allocates resources based on the application demand. In another work, Morshedlou and Meybodi [7] focused on mitigating SLA violations by introducing two new parameters – willingness to pay for service and willingness to pay for certainty. These two parameters are interactively obtained through web interfaces. Zheng *et al.* [8] focused on interactive cloud service negotiation using a mixed negotiation approach in presence of possible conflicts. In the context of dynamic resource provisioning through SLA, few works addressed the challenges and the importance of SLA negotiation [9], the performance of the process of optimal negotiation [10], and selection of a trusted CSP [11]. However, in all the above works, the end-user is expected to fill in the application demands and his/her satisfactory parameters. This work focuses to automate the process of SLA negotiation and selection of CSPs for naive users so that s/he can enjoy abstraction and obtain QoS, simultaneously. As far as IoT is concerned, Lee and Chong [12] discussed the provisioning of intelligent objects in Web-of-Objects as an IoT service. To enable IoT communication in a better manner, Jin *et al.* [13] proposed a Physical Service Model (PSM) that illustrates the IoT services and the relationships. Fredj *et al.* [14] focused on obtaining automatic service response based on request matching and forwarding using routing information. Zhang *et al.* [15] addressed the problem of inability of distributed events to describe the business logic. Thus, most of the work proposes and enables functions, algorithms, processes, and models to be implemented within the IoT architecture without manual intervention.

It is observable that the biggest challenge of IoT is the property of its enabling communication among various heterogeneous objects that do not possess the intelligence of processing or computation. Considering this challenge, this work focuses to design an algorithm QASeC, that automates the selection among multiple CSPs in a non-interactive manner simply based on the dynamic demands of an application.

III. QASeC: Problem Formulation

The problem of QASeC can be formulated as weighted voting and selection problem comprising of set $\mathcal{N} = \{\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_n\}$ of n CSPs, nominated to serve a particular IoT application App_i of end-user e_i. For any application served by $\mathcal{N}_i$, data packets are forwarded by cloud gateways associated with $\mathcal{N}_i$, denoted by $g_i \in \mathcal{G}$. A particular g_i is characterized by its uplink and downlink service capacity.

Average Available Bi-directional Link Service Capacity:

Definition 1. The uplink capacity of a gateway g_i at time t is denoted by $ul_{g_i,t}$ and is necessarily a value within the closed interval $[ul_{g_i}^{min}, ul_{g_i}^{max}]$, expressed in bits per second.

Definition 2. The downlink capacity of a gateway g_i at time t is denoted by $dl_{g_i,t}$ and is necessarily a value within the closed interval $[dl_{g_i}^{min}, dl_{g_i}^{max}]$, expressed in bits per second.

The uplink and downlink requirements of an application App_i is denoted by ul_{App_i} and dl_{App_i}, respectively.

Definition 3. The available uplink capacity of a gateway g_i at time t is the difference of its maximum uplink capacity and the summation of the uplink capacities of the total number of running application being served by g_i at t (denoted by η).

$$\mathcal{C}_{g_i,u}^t = ul_{g_i}^{max} - \sum_{i=1}^{\eta} ul_{App_i} \tag{1}$$

Definition 4. The available downlink capacity of a gateway g_i at time t is the difference of its maximum downlink capacity and the summation of the downlink capacities of the total number of running application being served by g_i at t.

$$\mathcal{C}_{g_i,d}^t = dl_{g_i}^{max} - \sum_{i=1}^{\eta} dl_{App_i} \tag{2}$$

The available bi-directional link service capacity of $\mathcal{N}_i$ with respect to g_j is obtained as $\mathcal{L}_{\mathcal{N}_i}^{g_j} = \mathcal{C}_{g_j,u}^t + \mathcal{C}_{g_j,d}^t$. Therefore, if κ number of gateways are mapped to a CSP $\mathcal{N}_i$ [16], the average available bi-directional link service capacity of $\mathcal{N}_i$ at time t is expressed as,

$$\hat{\mathcal{L}}_{\mathcal{N}_i} = \frac{1}{\kappa}\sum_{j=1}^{\kappa} \mathcal{L}_{\mathcal{N}_i}^{g_j} = \frac{1}{\kappa}\sum_{j=1}^{\kappa}\left(\mathcal{C}_{g_j,u}^t + \mathcal{C}_{g_j,d}^t\right) \tag{3}$$

Therefore, for an oligopolistic scenario, we have a set of the average available bi-directional link service capacities of of n nominated CSPs, denoted as $\{\hat{\mathcal{L}}_{\mathcal{N}_1}, \hat{\mathcal{L}}_{\mathcal{N}_2}, \ldots, \hat{\mathcal{L}}_{\mathcal{N}_n}\}$.

Mean Transmission Delay: Given the demand of App_i as λ_{App_i} (expressed as p packets of size per packet as P bits, i.e., $\lambda = pP$) and the propagation speed as V_{pr} (in meter per second), the transmission delay of a packet at cloud end is given as,

$$T^{g_j}_{tr_{App_i}} = \frac{1}{p}\left(\frac{\lambda_{App_i}}{\mathcal{C}^t_{g_j,u}} + \frac{\xi(g_j, c_{App_i})}{V_{pr}}\right) \tag{4}$$

Therefore, the mean transmission delay of a packet at cloud end is formulated as $\hat{T}^{\mathcal{N}_i}_{tr_{App_i}} = \frac{1}{p\kappa}\sum_{j=1}^{\kappa} T^{g_j}_{tr_{App_i}}$, i.e.,

$$\hat{T}^{\mathcal{N}_i}_{tr_{App_i}} = \begin{cases} \frac{1}{p\kappa}\sum_{j=1}^{\kappa}\left(\frac{\lambda_{App_i}}{\mathcal{C}^t_{g_j,u}} + \frac{\xi(g_j,c_{App_i})}{V_{pr}}\right) & \text{,if } ul_{App_i} \leq \hat{\mathcal{L}}_{\mathcal{N}_i} \\ \infty & \text{,otherwise} \end{cases} \tag{5}$$

c_{App_i} being the coordinate of the center of App_i and $\xi(g_j, c_{App_i})$ is the Euclidean distance between a gateway and an application center. If $ul_{App_i} > \hat{\mathcal{L}}_{\mathcal{N}_i}$, the CSP is unable to serve App_i. In such case, we consider $\hat{T}^{\mathcal{N}_i}_{tr_{App_i}}$ to be infinite for mathematical convenience.

Mean Hop Delay: As IoT thrives on underlying Wireless Sensor Networks (WSNs), without the loss of generality, it can be assumed that the communication occurs for a k-hop scenario ($k \geq 1$) and the hop nodes are denoted by $h_j \in H, 1 \leq j \leq k$. Thus, referring to the work of Pragad *et al.* [17], the delay at an intermediate hop is obtained as,

$$T^{h_j}_{hop} = E[\mu]\{(1 - P^o_R)T^l + P^o_R T^g\} \tag{6}$$

where, μ is a random variable of the random walk model [18] and T^l and T^g are respectively the local and global delays due to binding. P^o_R is the probability of a data packet moving out of a the sensing domain of sensor node (serving App_i) of radius R opposite to the direction of the gateways, $g_i \in \mathcal{G}$ [19]. Therefore, $(1 - P^o_R)$ is the probability of a data packet being forwarded towards the cloud gateways. T^l and T^g can be obtained as [17], $T^l = \frac{1}{n_1}\xi(h_j, g_i)T^p_{hop}, T^g = \frac{1}{n_2}(\xi(c_{App_i}, h_j) + \xi(h_j, g_i))T^p_{hop}$, where T^p_{hop} is the per packet delay at each hop node. n_1 and n_2 are the unit balancing constants. Equation (6) assumes App_i to be served by a single sensor node. However, for multiple sensors m serving App_i, Equation (6) can be rewritten as,

$$T^{h_j}_{hop} = \sum_{i=1}^{m}\left(E[\mu]\{(1 - P^o_{R_i})T^l + P^o_{R_i}T^g\}\right) \tag{7}$$

Now, for a k-hop scenario, the total delay incurred due to hops to reach g_i and the mean hop delay of a CSP is respectively obtained as,

$$T^{tot,g_i}_{hop} = \sum_{j=1}^{k} T^{h_j}_{hop} = \sum_{j=1}^{k}\sum_{i=1}^{m}\left(E[\mu]\{(1 - P^o_{R_i})T^l + P^o_{R_i}T^g\}\right) \tag{8}$$

$$\hat{T}^{\mathcal{N}_i}_{hop} = \frac{1}{\kappa}\sum_{j=1}^{\kappa} T^{tot,g_j}_{hop} \tag{9}$$

Mean Processing Delay: The current processing load of a gateway g_i is the summation of the total number of application requests being served by (α_s) and queued (α_q) within g_i. It is expressed as, $L_{g_i,t} = \alpha_{s_t,g_i} + \alpha_{q_t,g_i}$. Assuming τ_1 as the per bit processing delay within a gateway and τ_2 as the per application extraction delay from the queue, the processing delay of App_i is obtained as,

$$\begin{aligned} T^{g_i,App_i}_p &= \tau_1\left[\sum_{j=1}^{\alpha_{s_t,g_i}} \lambda_{App_j} + \sum_{k=\alpha_{s_t,g_i}+1}^{\alpha_{q_t,g_i}} \lambda_{App_k}\right] + \tau_2(\alpha_{q_t,g_i} + 1) \\ &= \tau_1\sum_{j=1}^{L_{g_i,t}} \lambda_{App_j} + \tau_2(\alpha_{q_t,g_i} + 1) \end{aligned} \tag{10}$$

Therefore the mean processing delay of App_i at the cloud end of CSP $\mathcal{N}_i$ is given as,

$$\hat{T}^{\mathcal{N}_i,App_i}_p = \sum_{j=1}^{\kappa} T^{g_j,App_i}_p = \tau_1\sum_{j=1}^{L_{g_i,t}} \lambda_{App_j} + \tau_2(\alpha_{q_t,g_i} + 1) \tag{11}$$

Mean Sensor Accuracy: To obtain the mean sensing accuracy of a particular sensor, the residual energy of the sensor is considered along with the deviation of the magnitude of the actual data and the final data obtained from the last hop sensor node. Assuming that $E^{s_i}_{init}$ and $E^{s_i}_{cur}(t)$ are the initial and the current energy level (in Joule) of a sensor node s_i, respectively, the residual energy can be obtained as $\gamma_{s_i}(t) = \frac{E^{s_i}_{cur}(t)}{E^{s_i}_{init}}$. It is to be noted that γ_{s_i} is a unit less quantity, $0 \leq \gamma_{s_i} \leq 1$. The quality of data transmission between a pair of nodes a and b at time t is derived as [20],

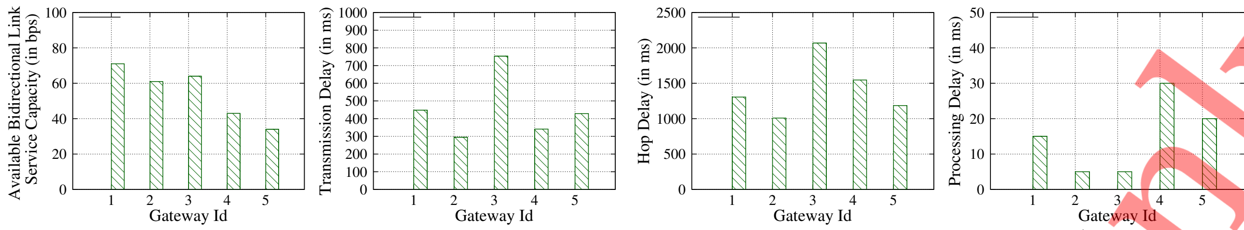
$$e_{a,b}(t) = \begin{cases} \frac{1}{N}e_{a,b}(t-1)e^{(\rho\delta)(t)}, & \rho = |D_a - D_b| < \rho_{th} \\ \frac{1}{N}e_{a,b}(t-1)e^{-(\rho\delta)(t)}, & \text{otherwise} \end{cases} \tag{12}$$

N being a normalization constant. ρ is the data deviation and δ is a profit/loss factor. Thus, for a k-hop scenario, the mean sensor accuracy of CSP $\mathcal{N}_i$ for App_i can be obtained as,

$$\hat{\mathcal{A}}^{App_i}_{\mathcal{N}_i} = e_{h_1,h_k}(t)\sum_{j=1}^{k}\gamma_{h_j}(t) \tag{13}$$

Mathematical Formulation & System Model

We now present the formulation of the problem of this work. The work focuses on a specific objective. The goal is to select a single CSP from $\mathcal{N}$. To achieve this, we initially perform a quantification of the QoS provisioned by a single CSP, against a particular gateway, in terms of the parameters – Average Available Bi-directional Link Service Capacity, Mean Transmission Delay, Mean Hop Delay, Mean Processing Delay, and Mean Sensor Accuracy. Thus, the objective function $f()$ can



(a) $\mathcal{L}_{\mathcal{N}_i}^{g_j}, 1 \le j \le 5$ (b) $T_{tr_{App_i}}^{g_j}, 1 \le j \le 5$ (c) $T_{hop}^{tot,g_j}, 1 \le j \le 5$ (d) $T_p^{g_j,App_i}, 1 \le j \le 5$

Figure 1: Study of multiple QoS parameters of a single CSP

be defined as, $f : \mathcal{N}^m \to \mathcal{N}, \mathcal{N} = \{\mathcal{N}_i\}, 1 \le i \le n$, and,

$$f(\mathcal{N}) = \mathcal{Q} = \{\mathcal{Q}_{\mathcal{N}_1}, \mathcal{Q}_{\mathcal{N}_2}, \ldots, \mathcal{Q}_{\mathcal{N}_n}\} \quad (14)$$

where, $\mathcal{Q}$ is the set of quantified QoS of every nominated CSP. Finally, the selection of a CSP is obtained as,

$$\mathcal{N}_s \in \mathcal{N}, |\ \mathcal{Q}_{\mathcal{N}_s} = \max\{\mathcal{Q}_{\mathcal{N}_1}, \mathcal{Q}_{\mathcal{N}_2}, \ldots, \mathcal{Q}_{\mathcal{N}_n}\} \quad (15)$$

Definition 5. The QoS utility metric for the QoS of a CSP $\mathcal{N}_i$ with respect to gateway g_j is expressed in terms of the Available Bi-directional Link Service Capacity ($\mathcal{L}$), Transmission Delay (T_{tr}), Hop Delay (T_{hop}), Processing Delay (T_p), and Mean Sensor Accuracy ($\mathcal{A}$) and is defined as,

$$\mathcal{U}_{\mathcal{N}_i}^{g_j} = \beta_{\mathcal{N}_i}^{g_j} + \frac{1}{\theta_{\mathcal{N}_i}^{g_j}} + \Psi_{\mathcal{N}_i} \quad (16)$$

where, $\beta_{\mathcal{N}_i}^{g_j}$, $\theta_{\mathcal{N}_i}^{g_j}$, and $\Psi_{\mathcal{N}_i}$ are the unit less components of $\mathcal{U}_{\mathcal{N}_i}^{g_j}$, expressed as,

$$\beta_{\mathcal{N}_i}^{g_j} = \frac{\mathcal{L}_{\mathcal{N}_i}^{g_j}}{\pi_1} \quad (17)$$

$$\theta_{\mathcal{N}_i}^{g_j} = \frac{T_{tr_{App_i}}^{g_j} + T_{hop}^{tot,g_j} + T_p^{g_j,App_i}}{\pi_2}, \Psi_{\mathcal{N}_i} = \frac{\mathcal{A}_{\mathcal{N}_i}^{App_i}}{\pi_3} \quad (18)$$

where π_1, π_2, and π_3 are system constants introduced to cancel the effects of units of $\mathcal{L}_{\mathcal{N}_i}^{g_j}$, $T_{tr_{App_i}}^{g_j}$, or T_{hop}^{tot,g_j}, or $T_p^{g_j,App_i}$, and $\mathcal{A}_{\mathcal{N}_i}^{App_i}$, respectively.

Thus, using Equations (16) through (18), we obtain the vector $\mathcal{U} = \{\mathcal{U}_{\mathcal{N}_1}, \mathcal{U}_{\mathcal{N}_2}, \ldots, \mathcal{U}_{\mathcal{N}_n}\}$ for the set of nominated CSPs. Now, it can be observed that each of the components of $\mathcal{U}_{\mathcal{N}_i}$, except for $\Psi_{\mathcal{N}_i}$ vary with the change in the chosen gateway of $\mathcal{N}_i$ i.e., $\beta_{\mathcal{N}_i}^{g_j}$ and $\theta_{\mathcal{N}_i}^{g_j}$ are gateway dependent factors of $\mathcal{U}_{\mathcal{N}_i}$. Therefore, the QoS utility metric of a CSP values vary with the change in the gateway. Thus, to get selected, a CSP should strategically allocate the gateway that maximizes the provisioned QoS with respect to a particular end-user and his/her center of application. Herein, the optimization is formulated that allocates a gateway to an end-user so that the QoS utility metric is maximized. The objective of the optimization is stated as,

$$\text{Maximize } \{\mathcal{U}_{\mathcal{N}_i}^{g_j}\}, 1 \le j \le \kappa \quad (19)$$

The constraints of the optimization is to minimize the error (or the mean square deviation) within a threshold. As $\Psi_{\mathcal{N}_i}$ is gateway independent, the formal optimization is stated as,

$$\text{Maximize } \left\{\beta_{\mathcal{N}_i}^{g_j} + \frac{1}{\theta_{\mathcal{N}_i}^{g_j}}\right\}, 1 \le j \le \kappa \quad (20)$$

subjected to the constraints,

$$\left(\mathcal{L}_{\mathcal{N}_i}^{g_j} - \hat{\mathcal{L}}_{\mathcal{N}_i}\right)^2 \le \Delta_{\mathcal{L}}, \left(T_{tr_{App_i}}^{g_j} - \hat{T}_{tr_{App_i}}^{\mathcal{N}_i}\right)^2 \le \Delta_{T_{tr}}$$

$$\left(T_{hop}^{tot,g_j} - \hat{T}_{hop}^{\mathcal{N}_i}\right)^2 \le \Delta_{T_{hop}}, \left(T_p^{g_j,App_i} - \hat{T}_p^{\mathcal{N}_i,App_i}\right)^2 \le \Delta_{T_p}$$

where $\Delta_{\mathcal{L}}$, $\Delta_{T_{tr}}$, $\Delta_{T_{hop}}$, and Δ_{T_p} are the error thresholds for the link service capacity, transmission delay, hop delay, and the processing delay, respectively. We apply optimization technique using Lagrangian multiplier and obtain $g_{\mathcal{N}_i}^*$ as the selected gateway of $\mathcal{N}_i$ to be allocated for App_i, such that,

$$\mathcal{Q}_{\mathcal{N}_i} = \mathcal{U}_{\mathcal{N}_i}^{g_{\mathcal{N}_i}^*} = \max\{\mathcal{U}_{\mathcal{N}_i}^{g_j}\}, 1 \le j \le \kappa \quad (21)$$

Hence for every $\mathcal{N}_i \in \mathcal{N}$ we obtain, $\{g_{\mathcal{N}_1}^*, g_{\mathcal{N}_2}^*, \ldots, g_{\mathcal{N}_n}^*\}$. Subsequently, we obtain $\{\mathcal{Q}_{\mathcal{N}_1}, \mathcal{Q}_{\mathcal{N}_2}, \ldots, \mathcal{Q}_{\mathcal{N}_n}\}$. Using Equation (15), we obtain $\mathcal{N}^*$, such that,

$$\mathcal{Q}_{\mathcal{N}^*} = \max\{\mathcal{Q}_{\mathcal{N}_1}, \mathcal{Q}_{\mathcal{N}_2}, \ldots, \mathcal{Q}_{\mathcal{N}_n}\} \quad (22)$$

Thus CSP $\mathcal{N}^*$ is selected by end-user for application App_i.

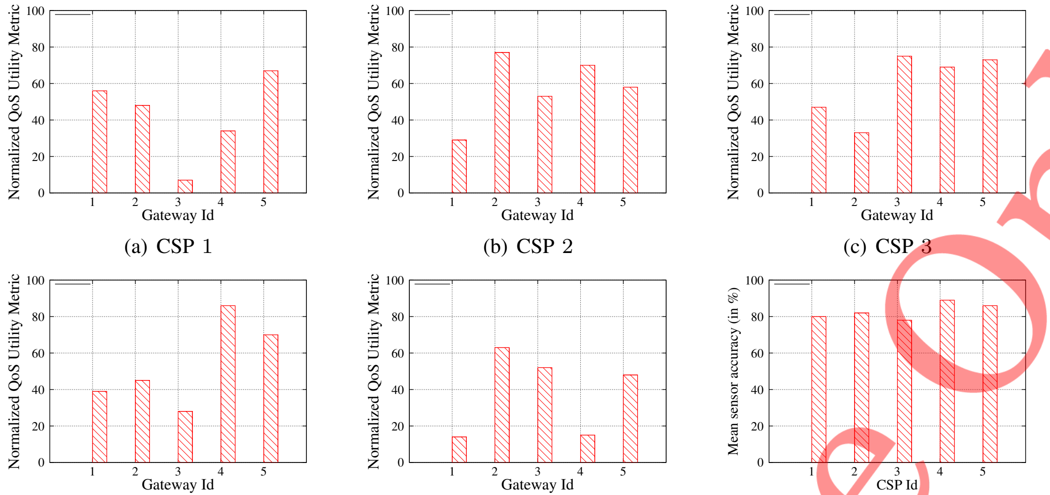
IV. Performance Evaluation

In this Section, we initially illustrate the steps of QoS quantification and gateway allocation of a single CSP. The experimental setup is provided in Table I.

Table I: Experimental setup for a single IoT application

Parameters	Values
Minimum link service capacity	40 bps
Maximum link service capacity	100 bps
Demand distribution of an incoming application	Poisson ($\lambda = 7$ bps)
Number of hops (k)	3
$\Delta_{\mathcal{L}}$	50 bps
$\Delta_{T_{tr}}$	400 ms
$\Delta_{T_{hop}}$	1200 ms
Δ_{T_p}	10 ms

A. Case Study of QoS Quantification for a Single CSP



(a) CSP 1 (b) CSP 2 (c) CSP 3

(d) CSP 4 (e) CSP 5 (f) Mean sensor accuracy for each CSP

Figure 2: Study of multiple QoS parameters of a multiple CSPs

To illustrate through a case study, we perform an experiment of a single CSP, 5 gateway scenario, $g_1, g_2, \ldots, g_5$. For an incoming application, the QoS parameters of the CSP are evaluated. We obtain distinct values for Available Bi-directional Link Service Capacity ($\mathcal{L}_{\mathcal{N}_i}^{g_j}$) in Figure 1(a), Transmission Delay ($T_{tr_{App_i}}^{g_j}$) in Figure 1(b), Hop Delay (T_{hop}^{tot,g_j}) in Figure 1(c), and Processing Delay ($T_p^{g_j,App_i}$) in Figure 1(d). For the single application, we consider the Mean Sensor Accuracy ($\mathcal{A}$) to be 0.8 as it is gateway independent. The optimized measures are illustrated in Table II. Having performed the constrained optimization on the retrieved values, it is observed that the normalized sum of error for the gateways in sequence are 0.02, 0.39, 0.9, 0.05, and 0.11, respectively. It can be observed that, g_3 exhibits a very high transmission and hop delay resulting to a very high error value. On the contrary, we observe that g_1 peaks in terms of the available link service capacity and performs moderately in terms of the delays. Consequently, after optimization, g_1 achieves the lowest magnitude of the normalized error. Hence, a CSP would allocate g_1 for an incoming application.

Table II: Evaluation of the mean square deviation of the gateway dependent QoS parameters

Gateway IDs	$\left(\mathcal{L}_{\mathcal{N}_i}^{g_j} - \hat{\mathcal{L}}_{\mathcal{N}_i}\right)^2$	$\left(T_{tr}^{g_j} - \hat{T}_{tr}^{\mathcal{N}_i}\right)^2$	$\left(T_{hop}^{tot,g_j} - \hat{T}_{hop}^{\mathcal{N}_i}\right)^2$	$\left(T_p^{g_j} - \hat{T}_p^{\mathcal{N}_i}\right)^2$
1	268.96	24.19883983	13852.01479	0
2	40.96	25028.32237	171572.6326	100
3	88.36	89993.70956	416792.8988	100
4	134.56	12605.27026	15328.78311	225
5	424.36	604.8485299	56404.63823	25

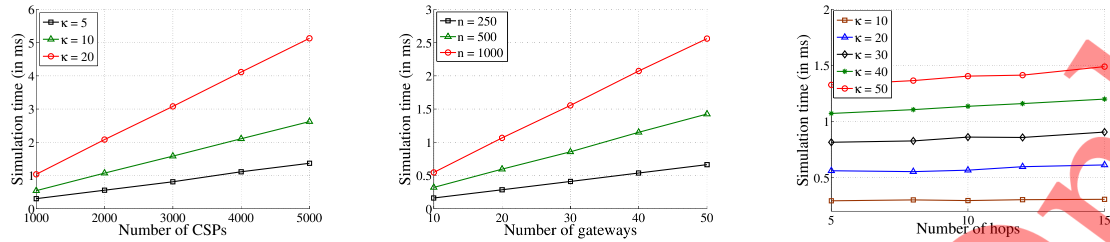
B. Performance Evaluation for Multiple CSPs

In order to evaluate a multi-CSP scenario of QASeC, we perform an experiment involving 5 CSPs with 5 gateways mapped to each CSP. We obtain the QoS utility metric, $\mathcal{U}_{\mathcal{N}_i}^{g_i}$, of every gateway of each CSP. Followed by this, we normalize the metric within a scale of $\mathcal{U}_{min}$ to $\mathcal{U}_{max}$, as shown in Figure 2 using $\hat{\mathcal{U}}_{\mathcal{N}_i}^{g_i} = \mathcal{U}_{\mathcal{N}_i}^{g_i}/\mathcal{U}_{max}, \mathcal{U}_{\mathcal{N}_i}^{g_i} \geq \mathcal{U}_{min}$, where $\mathcal{U}_{min} = 100 = 1, \mathcal{U}_{max} = 100$, in our case. However, the mean sensor accuracy is not normalized as it is gateway independent. Based on the normalized QoS utility metric and the mean sensor accuracy, we thereby perform the constrained optimization on the available data set of 5 CSPs and obtain the values as $\mathcal{Q}_{\mathcal{N}}^{*} = \{67.00, 77.00, 75.00, 86.00, 63.00\}$. Finally, using Equation (22), CSP $\mathcal{N}_4$ is selected for serving the incoming application.

C. Complexity Analysis

For the sake of analysis of the runtime efficiency of QASeC, we choose the simulation execution time to be the metric for analysis of the computational complexity. The initial simulation was performed by varying the number of CSPs from 1000 to 5000, as shown in Figure 3(a). We execute the simulation thrice by varying the number of gateways allocated under every CSP (κ) to be 5, 10, and 20. As indicated by Figure 3(a), we observe that with 1000 CSPs the simulation time varies from 0.3 ms to 1 ms. With the increase in the number of gateways the simulation execution time shows a marginally rising trend for $\kappa = 5$ and $\kappa = 10$. However, for $\kappa = 20$, the simulation time increases considerably for higher number of gateways (i.e., 4000 to 5000).

The next simulation was performed by varying κ from 10 to 50. The results were analyzed for a CSP count of 250, 500, and 1000. It can be clearly observed in Figure 3(b) that even with the simultaneous increase in the number of CSPs and gateways, the simulation execution time mildly increases and maintains an average of approximately 0.95 ms. Figure 3(c) depicts the variation of the simulation execution time with the increase in the number of hops k from 5 to 15. The simulation is repeated 5 times for different number of gateways (κ) allocated to 1000 CSPs. We observe that the



(a) Analysis of computational complexity vs. the number of CSPs

(b) Analysis of computational complexity vs. the number of gateways

(c) CSP 3

Figure 3: Study of multiple QoS parameters of a multiple CSPs

curve exhibits a parallel growth trend with an increase of 10 gateways at every iteration. However, for a particular value of k, the value of κ undergoes negligible change and is observed to be almost identical. From the above results obtained from Figures 3(a), 3(b), and 3(c), the simulation time is affordable and realizable for real-life scenarios. Thus, we infer the real-time applicability of the proposed work.

V. Conclusion

This work focuses on the problem of selecting a single CSP from set of multiple CSPs in a multi-user IoT scenario. Initially, the CSPs are parameterized by few QoS parameters and based on the parameters, a QoS utility metric is formulated for every gateway of each CSP. Followed by this, we discuss the constrained optimization executed by every CSP for allocating a gateway. Eventually, the algorithm achieves a quantification of the QoS that can be provisioned by every CSP thereby obtaining the CSP with the highest QoS. Results discuss the steps of selecting a CSP from a set of nominated CSPs and further study and analyze the real-time computing ability of the proposed work in real-life IoT scenarios.

Acknowledgments

This work was partially supported by a fellowship sponsored by the Tata Consultancy Services (TCS), India.

References

[1] D. Petcu, "A taxonomy for sla-based monitoring of cloud security," in *IEEE 38th Annual Computer Software and Applications Conference (COMPSAC)*, July 2014, pp. 640–641.

[2] Internet of things market forecast:. Cisco. [Online]. Available: http://postscapes.com/internet-of-things-market-size

[3] H.-L. Truong and S. Dustdar, "Principles for engineering iot cloud systems," *IEEE Cloud Computing*, vol. 2, no. 2, pp. 68–76, Mar 2015.

[4] Y. Feng, B. Li, and B. Li, "Price competition in an oligopoly market with multiple iaas cloud providers," *IEEE Trans. on Comp.*, vol. 63, pp. 59–73, 2014.

[5] K.-Y. Foong. (2015, April) Market trends: CSP sales strategies for the internet of things and M2M communication services. Gartner, Inc. [Online]. Available: https://www.gartner.com/doc/3028219/market-trends-csp-sales-strategies

[6] L. Wu, S. Garg, S. Versteeg, and R. Buyya, "Sla-based resource provisioning for hosted software-as-a-service applications in cloud computing environments," *IEEE Transactions on Services Computing*, vol. 7, no. 3, pp. 465–485, July 2014.

[7] H. Morshedlou and M. Meybodi, "Decreasing impact of sla violations:a proactive resource allocation approachfor cloud computing environments," *IEEE Trans. on Cloud Comp.*, vol. 2, pp. 156–167, Apr 2014.

[8] X. Zheng, P. Martin, K. Brohman, and L. D. Xu, "Cloud service negotiation in internet of things environment: A mixed approach," *IEEE Trans. on Industrial Informatics*, vol. 10, pp. 1506–1515, May 2014.

[9] S. Sakr and A. Liu, "Sla-based and consumer-centric dynamic provisioning for cloud databases," in *IEEE 5th International Conference on Cloud Computing (CLOUD)*, June 2012, pp. 360–367.

[10] S. Maity and A. Chaudhuri, "Optimal negotiation of sla in federated cloud using multiobjective genetic algorithms," in *IEEE 3rd International Conference on Cloud Networking*, Oct 2014, pp. 269–271.

[11] O. Jules, A. Hafid, and M. Serhani, "Bayesian network, and probabilistic ontology driven trust model for sla management of cloud services," in *IEEE 3rd International Conference on Cloud Networking (CloudNet)*, Oct 2014, pp. 77–83.

[12] S. Lee and I. Chong, "User-centric intelligence provisioning in web-of-objects based iot service," in *International Conference on ICT Convergence (ICTC)*, Oct 2013, pp. 44–49.

[13] X. Jin, S. Chun, J. Jung, and K.-H. Lee, "Iot service selection based on physical service model and absolute dominance relationship," in *IEEE 7th International Conference on Service-Oriented Computing and Applications (SOCA)*, Nov 2014, pp. 65–72.

[14] S. Ben Fredj, M. Boussard, D. Kofman, and L. Noirie, "A scalable iot service search based on clustering and aggregation," in *IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing Green Computing and Communications (GreenCom)*, Aug 2013, pp. 403–410.

[15] Y. Zhang, L. Duan, and J. L. Chen, "Event-driven SOA for iot services," in *IEEE Intl. Conf. on Services Computing*, June 2014, pp. 629–636.

[16] S. Misra, S. Bera, A. Mondal, R. Tirkey, H.-C. Chao, and S. Chattopadhyay, "Optimal gateway selection in sensor-cloud framework for health monitoring," *IET Wireless Sensor Systems*, vol. 3, no. 4, December 2013.

[17] D. Pragad, A. Jaron, P. Pangalos, and A. H. Aghvami, "Dynamic mobility anchor selection mechanism with qos constraints," *IEEE Communications Letters*, vol. 15, pp. 1094–1096, 2011.

[18] I. F. Akyildiz and W. Wang, "A dynamic location management scheme for next-generation multitier pcs systems," *IEEE Transactions on Wireless Communications*, vol. 1, no. 1, pp. 178–189, January 2002.

[19] J. S. Ho and I. F. Akyildiz, "Mobile user location update and paging under delay constraints," *J. Wireless Netw*, vol. 1, pp. 244–255, 1995.

[20] M. Hossain, P. Atrey, and A. Saddik, "Context-aware QoI computation in multi-sensor systems," in 5th *IEEE Intl Conf on Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008.*, Sept 2008, pp. 736–741.