Cheating-Resilient Bandwidth Distribution in Mobile Cloud Computing

Snigdha Das, Manas Khatua, *Member, IEEE*, and Sudip Misra, *Senior Member, IEEE*

Abstract—In mobile cloud computing (MCC), optimal utilization of resources (e.g., bandwidth), while maintaining the required level of quality-of-services (QoS), is essential. A user participating in the resource allocation process can provide untruthful information for acquiring undue advantages with respect to the allocated resource amount, and the cost incurred. In this paper, we identify, formulate, and address the problem of such misbehaviour. We formulate the bandwidth distribution as a constrained convex utility maximization problem, and solve it using the proposed *cheating-resilient bandwidth distribution* (CRAB) scheme. Numerical analysis shows that, in CRAB, the misbehaving user is impelled to behave normally as the misbehaviour increases its own cost while the other users including the cloud service provider (CSP) get benefit in terms of revenue. We investigate the existence of Nash Equilibrium (NE) of the proposed scheme. Both the problem and the solution are extensively analysed theoretically. The maximum and minimum selling prices of bandwidth, and the optimal solution for individual user are computed using the method of Lagrange multiplier.

Index Terms—MCC, user misbehaviour, bandwidth distribution, auction theory, utility maximization

1 Introduction

At the present time, mobile devices are inseparable components of human society. However, the exploitation of their full functionality is restricted by the limited energy supply, low bandwidth, low computational capacity, and seamless mobility issues. To cope with such restrictions, mobile devices are integrated with existing cloud computing infrastructure forming mobile cloud computing (MCC). The data processing and storing applications are moved from the mobile devices to the servers in a cloud, which, in turn, improves the efficiency and lifetime of the mobile devices. Following the MCC architecture described in [1], [2], we present a schematic view of the architecture in Fig. 1. In the architecture, mobile users are connected with access points (APs). The requests of the mobile users are transmitted to the cloud through the interfacing gateways. The cloud service provider (CSP) processes the requests with the help of application and infrastructure controllers, and, then, sends back the reply to the mobile users through the same interfacing gateway. If the CSP has sufficient resources, it allocates requested resources to the users. Otherwise, it may rent some resources from other sources. In this architecture, the gateways are involved only for transferring the service requests and responses. They also request resources from the cloud for maintaining uninterrupted connection between the users and the cloud.

1.1 Motivation

In MCC, for providing real-time computing, processing, and storing services over wireless networks, bandwidth is one of the fundamental considerations. It may happen that despite having adequate bandwidth in the cloud for providing the requested services to the mobile users, the users are not able to receive the services due to inappropriate bandwidth management. Additionally, the CSP needs to assure certain level of quality-of-service (QoS) in terms of delay, reliability, jitter, and response time for encouraging the users to take services from it. On the other hand, mobility is the true nature of a device in MCC. When a mobile device changes its location, the corresponding gateway for maintaining connectivity with the cloud may change. We assume that the update of location information and its corresponding wireless connectivity change is maintained by the mobile network administrator. In such a changing environment, bandwidth management for maintaining QoS requirement is a fundamental challenge, as discussed by Misra et al. [2]. Consequently, the authors proposed the *bandwidth distribution* solution approach, in which the users' QoS requirements are fulfilled. Bandwidth distribution ensures fair distribution of the total amount of available bandwidth proportional to their demand. However, if any of the connecting gateways exhibits its misbehaving characteristics during the distribution process, optimal bandwidth management cannot be assured. Further, the CSP also suffers from undesirable distribution.

Definition 1 (Misbehaving Gateway). *A gateway is termed as misbehaving, when it provides "untruthful" information during the resource management process for receiving undue privileges.*

A gateway is termed as *normal* when it provides "truthful" information during the resource management process. Therefore, a cheating-resilient bandwidth distribution scheme is essential for avoiding such loss and over-pay.

1.2 Contributions

In this paper, we analyze the vulnerability of the *generic bandwidth distribution* approach, namely *AQUM*, proposed

- *The authors are with the School of Information Technology, Indian Institute of Technology Kharagpur, Kharagpur, West Bengal 721302, India. E-mail: {snigdha00582, manaskhatua}@gmail.com, smisra@sit.iitkgp.ernet.in.*

Manuscript received 1 July 2016; revised 19 Nov. 2016; accepted 28 Nov. 2016. Date of publication 13 Dec. 2016; date of current version 5 June 2019.
Recommended for acceptance by J. Cao.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TCC.2016.2638909

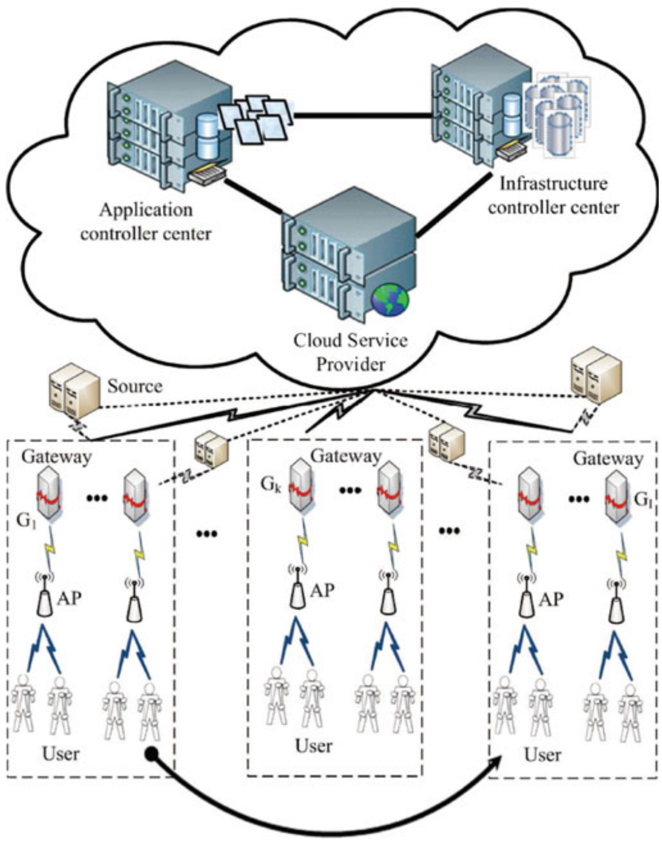


Fig. 1. Mobile cloud computing architecture.

by Misra et al. [2], in the presence of a misbehaving gateway. We address the issue of misbehaviour by designing a *cheating-resilient bandwidth distribution* algorithm, namely CRAB. The CRAB is capable of enforcing the misbehaving gateways to behave normally by increasing their cost needed to pay for purchasing the required amount of bandwidth from the CSP. We formulate it as a *constrained convex utility maximization* problem for the gateways, and solve it using the *descending price auction* mechanism. In brief, each gateway computes the total amount of required bandwidth for providing services and submits the corresponding bid to the CSP. The CSP allocates the bandwidth depending upon the final bidding value and the intermediate bidding increment value. The modified payment system in the CRAB includes a penalty factor for resisting the gateways from doing any misbehaviour. We theoretically deduce the selling price boundary of bandwidth, and prove the existence of the Nash Equilibrium (NE) in the proposed algorithm. We also establish the optimal solution of CRAB using the method of Lagrange multiplier. Finally, we prove that the net revenue received by the CSP is increased if any gateway misbehaves, which, in turn, enforces a gateway not to misbehave. We summarize our main *contributions* as follows:

- We theoretically prove the *revenue loss* of a CSP in the presence of any misbehaving gateway in the AQUM. We also prove that the misbehaving gateway gets benefit in terms of total cost pay in the AQUM.
- We propose a *cheating-resilient bandwidth distribution* (*CRAB*) algorithm, followed by designing a *constrained convex utility maximization* problem for distributing the total available bandwidth with required proportion, even in the presence of misbehaving gateway.
- We analyse the existence of *Nash Equilibrium* for the proposed algorithm, and determine the optimal solution of the proposed algorithm using the *Lagrange Multiplier* method.
- We analyse the proposed algorithm for multiple misbehaving gateways to confirm its *scalability*.

1.3 Paper Organization

The rest of the paper is organized as follows. In Section 2, we briefly present few relevant related works reported in the existing literature. In Section 3, we theoretically prove that vulnerability exists in the generic approach of bandwidth distribution. We explain the total utility maximization problem formulation followed by a distributed and cheat-proof auction theory-based solution approach in Section 4. The theoretical analysis of the proposed algorithm is presented in Section 5, followed by discussions on simulation results in Section 6. Finally, we conclude the paper in Section 7 with discussions about how this work can be extended in the future.

2 Related Work

Even though the integration of cloud computing in the mobile environment benefits users in many ways, few recent works such as [1], [3], and [8] highlights some of the major drawbacks such as limited bandwidth, pricing problem, inadequate QoS provisioning, and security issues. The *utility-over-network* model in MCC makes the network more vulnerable to attack.

Resource allocation problems extensively studied in other domains such as cognitive radio networks (CRNs), wireless networks, and mobile social networks. Wang et al. [7] proposed a novel auction-based approach for spectrum sharing in CRNs. A stochastic differential game-theoretic dynamic bandwidth allocation problem was investigated by Zhu et al. [9] for both the users and the service providers. A game-theoretic resource allocation scheme is proposed by Kaur et al.[10] for balancing the demand and supply in micro-grids. Zhang et al. [11] briefly summarized the different auction approaches for resource allocation in wireless systems. Few works also exist in cloud computing environment. Zhang et al. [4] proposed a combinatorial auction scheme for distributing heterogeneous resources, and Lin et al. [12] addressed the problem of varying QoS requirements for different data incentive applications in cloud computing systems. Bonacquisto et al. [13] proposed an alternative auction-based mechanisms for selling the "residual" computing capacity in cloud market. A resource provisioning framework through a grid of local computing resources is proposed in [14]. Gai et al. [15] proposed a cloudlet-based model in MCC for conserving energy resource. Many researchers have also used utility theory as a strategy for bandwidth allocation. For example, the utility-based resource allocation problem is reported in [16]. However, the total amount of requested resource must be within the availability of that resource. If the demand exceeds beyond the availability, two scenarios occur in existing networks. First, the requested demand

TABLE 1
Different Resource Management Schemes and Respective Solutions with Parameter Metrics

Schemes	Resource Type	Delay Metric	Misbehaviour	Mobility	Distributed Algo.	Distribution/Allocation
[3]	CPU, Data Storage	×	System damage	×	✓	Allocation
[4]	CPU, Database	×	Untruthful	✓	✓	Allocation
[5]	Virtual Machine	×	Untruthful	×	✓	Allocation
[6]	Spectrum	✓	Untruthful	×	✓	Allocation
[7]	Spectrum	×	No	×	✓	Allocation
AQUM[2]	Bandwidth	✓	No	✓	×	Distribution
This work	Bandwidth	✓	Untruthful	✓	✓	Distribution

cannot be served. Second, the requested demand serves with reduced QoS.

The problem of social surplus for efficient bandwidth allocation in wireless networks using generalized VCG auction mechanism with network coding is discussed in [17]. Niyato et al. [18] proposed game-theoretic solution scheme for allocating bandwidth in mobile social networks. However, the fundamental *differing characteristic* of the work presented in this paper from the existing ones is that we consider bandwidth distribution in the presence of misbehaving users. Additionally, we consider the scenario in which the total amount of requested resources exceed the availability of that resources.

Concurrently, many works such as [6], [8], [19], and [20], on different aspects of security issues in cloud computing, have been done. Chen et al. [6] proposed three auction-based mechanisms for distributive spectrum allocation in CRNs based on three important characteristics—convergence, social welfare, and cheat-proofness. A cheat-proof bidding approach for cloud environment is also discussed in [3]. The authors proposed an anti-cheating bidding scheme for investigating the bidding strategy of bidders, and, in turn, protecting the system from malicious bidders. Zaman and Grosu [5] proposed an auction-based mechanism for dynamically allocating virtual machines (VM) in cloud environment. Intelligent context switching of VMs for its proper allocation is proposed by Kumar et al. [21]. A double multi-attribute auction based resource allocation mechanism in cloud environment has been studied by Wang et al. in [22].

The bandwidth allocation problem in MCC, as such, has been studied in the paper. Amamou et al. [23] proposed a service level agreement-aware dynamic bandwidth allocator (DBA), in which bandwidth is allocated among the VMs for different application requirements associated with each VM. Papagianni et al. [24] proposed a unified resource allocation framework for networked clouds. Das et al. [25] considered the mapping of cloud server and mobile users in MCC. Similarly, many other works on bandwidth allocation exist in the literature. Recently, Misra et al. [2] studied the bandwidth shifting and bandwidth redistribution problem and Das et al. [26] studied the quality-assured load sharing problem in a typical MCC. However, none of these schemes addresses the problem of providing QoS-guaranteed bandwidth distribution in the presence of misbehaving users in MCC. A comparative study of few important works is shown in the Table 1.

3 Mobile Cloud Network Model

At the outset, we list the notations used in this paper in Table 2 for removing ambiguity. We consider a simple mobile cloud system with one CSP and I single-channel gateways $\mathbf{G} = \{G_1, G_2, \ldots, G_I\}$ connected with the CSP through a wireless channel. Each gateway G_i connects K number of mobile users at time t, and each mobile user N_{ik} experiences ideal transmission delay T_{ik} for accessing the service through the gateway G_i. Hence, the set of users connected with gateway G_i at time t is $\mathbf{N}_i(t) = \bigcup_{j=1}^{K} N_{ij}(t)$, and the total set of users present in the system is $\mathbf{N} = \bigcup_{i=1}^{I} \mathbf{N}_i(t)$. Let us assume that the CSP has B_{tot} amount of total bandwidth available with it. We further assume that protocol overhead and spectral efficiency of each channel are distinct, and are represented by the vectors $\boldsymbol{\alpha}(t) = \{\alpha_1(t), \alpha_2(t), \ldots, \alpha_I(t)\}$ and $\mathbf{E}(\mathbf{t}) = \{E_1(t), E_2(t), \ldots, E_I(t)\}$, respectively. In fact, we consider the same network environment as considered by Misra et al. [2], and, therefore, the effective spectral efficiency of a wireless channel is same as explained in Section 3.1 of [2]. At time t, $\mathbf{B}(t) = \{B_1(t), B_2(t), \ldots, B_I(t)\}$ represents the allocated bandwidth vector

TABLE 2
Summary of Notations

Notation	Description
G_i	Gateway i
I	Total number of gateways
N_i	Set of nodes connected to a gateway G_i
U_i	Utility of gateway G_i in normal case
$\widetilde{U}_i$	Utility of gateway G_i in misbehaving case
C_i	Cost paid by the gateway G_i in normal case
$\widetilde{C}_i$	Cost paid by the gateway G_i in misbehaving case
R_s	Revenue received by the CSP
E_i	Spectral efficiency of a channel associated with G_i
T_{ik}	Transmission delay required for accessing a service by a mobile node N_{ik} connected to gateway G_i
d_i	Service delay associated with a gateway G_i
B_i	Allocated bandwidth to a gateway G_i
B_{tot}	Total available bandwidth in CSP
β	Reserved bandwidth for CSP
B^*_{tot}	Available bandwidth excluding reserved amount β
α_i	Protocol overhead corresponding to a gateway G_i
b_i	Bid value given by a gateway G_i
s_i	shifting factor of gateway G_i
r_i	Revenue per unit service delay received by G_i
q_i	Revenue per unit transmission rate received by G_i
p	Price per unit bandwidth allocation charged by CSP
d_{θ_i}	Threshold value of service delay corresponding to G_i
ϕ_i	Minimum requirement of bandwidth for a gateway G_i to maintain its own operations and QoS
δ	A parameter contains positive number
Δp	Price decrement factor
H	Number of iteration in normal case
$\widehat{H}$	Number of iteration in misbehaving case

for the gateways. In this work, we consider QoS-guarantee in respect of *service delay* [2]. We compute the service delay d_i for the gateway G_i as $d_i = [B_{tot} \sum_{k=1}^{|N_i|} T_{ik}]/[E_i(1-\alpha_i)B_i]$, where $(1-\alpha_i)B_i$ represents the bandwidth allocated to the gateway G_i. Using this service delay metric, we only attempt to capture the delay variation that would occur in service provisioning to a user changing its interfacing gateway with the CSP due to mobility. To clarify further, the service delay metric does not measure the real delay that occurs in a real network, and, thus, does not include the other delay parameters related to real delay such as queuing and computation.

3.1 Gateway Utility Formulation

We design a utility function for computing the overall benefit of each gateway. The utility function of the gateway depends on the service it provides to the mobile users and the bandwidth it buys for providing the services. Each gateway pays certain price for getting the required bandwidth from the CSP. On the other hand, each gateway charges certain amount of revenue from the mobile users for providing the services to them, and assuring QoS in terms of service delay. We assume that no information loss occurs during this service request and resource allocation process. Hence, the utility function for all the gateways is expressed as follows:

$$\mathbf{U}(\mathbf{b}, p) = \mathbf{R}^Q(\mathbf{b}) + \mathbf{R}^B(\mathbf{b}) - \mathbf{C}(\mathbf{b}, p), \tag{1}$$

where $\mathbf{b} = \{b_1, b_2, \dots, b_I\}$ denotes the requested bid vector by the interfacing gateways. In other words, the submitted bid vector $\mathbf{b}$ also represents the requested bandwidth vector, in which b_i defines the requested bandwidth of the interfacing gateway G_i. The parameter $p(t)$ represents the confirmed price per unit bandwidth allocation given by the CSP. The vectors $\mathbf{R}^Q(\mathbf{b})$ and $\mathbf{R}^B(\mathbf{b})$ define the gateway revenue function based on service delay, and bandwidth allocation, respectively. $\mathbf{C}(\mathbf{b}, p)$ defines the bandwidth allocation cost function. The unit of all the three factors in the utility function is price. We model the three utility components as follows:

1. *Service Delay Specific Revenue Function*. The gateway with small service delay increases the overall revenue. We get the *service delay specific revenue function* for a gateway from the model presented in [2] as follows: $\mathbf{R}^Q(\mathbf{b}) = \mathbf{s} - \mathbf{rd}$, where the vector $\mathbf{r} = \{r_1, r_2, \dots, r_I\}$ defines the revenue per unit service delay occurring at the corresponding interfacing gateways, $\mathbf{d} = \{d_1, d_2, \dots, d_I\}$ is the service delay vector, and $\mathbf{s} = \{s_1, s_2, \dots, s_I\}$ defines the shifting factor. This factor shifts the impact of revenue degradation related to service delay.

2. *Bandwidth Allocation Specific Revenue Function*. High bandwidth allocated to a gateway results in an increase in the overall revenue. Similarly, we get the *bandwidth allocation-specific revenue function* from the model given in [2], as follows: $\mathbf{R}^B(\mathbf{b}) = \mathbf{qE}(\mathbf{1} - \boldsymbol{\alpha})\mathbf{B}$, where, the vector $\mathbf{q} = \{q_1, q_2, \dots, q_I\}$, $\mathbf{1} = \{1, 1, \dots, 1\}$, and $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \dots, \alpha_I\}$ denote the revenue per unit transmission rate, and protocol overhead, respectively.

3. *Cost Function*. The cost function of the gateway is computed using the submitted bid and the price per unit allocation as

$$\mathbf{C}(\mathbf{b}(t), p(t)) = p(t)\mathbf{b}(t). \tag{2}$$

3.2 CSP Revenue Formulation

One of the primary objectives of the CSP is to maximize the revenue received from the consumers by distributing the available resources among them. Let us consider that, at time t, the final allocation of resources is done. Let the bid vector and final price given by the CSP be $\mathbf{b}(t)$ and $p(t)$, respectively. Therefore, the total revenue received by the CSP is computed as follows:

$$R_s(\mathbf{b}(t), p(t)) = p(t) \sum \mathbf{b}(t) - \mathcal{C}_{rent}, \tag{3}$$

where, $\mathcal{C}_{rent}$ represents the cost for renting the resources from the third party.

3.3 Attack Model

The existing bandwidth distribution approach, namely AQUM [2], is primarily suitable for normal gateways. In the presence of misbehaviour of gateways, this approach is unable to distribute bandwidth with optimal cost. In this work, we consider the misbehaviour of a gateway by providing untruthful bidding values. In the truthful scenario, the bid value represents the proportional bandwidth demand by the users connected with the gateway. There exists many other possible ways of misbehaviour such as allocating more bandwidth to a subset of users, while the others get lower than their demands. However, we did not consider anything except the first one, and we plan to consider few more possibilities in the future.

3.4 Vulnerability of AQUM

We mathematically deduce a set of sufficient conditions under which the generic approach fails to achieve the specified objectives if any gateway misbehaves. That conditions are as follows:

Condition 1. (a) $\frac{\overline{p}}{\Delta p} > 2I\widehat{H}$ (b) $\frac{b}{\delta} < H(I-1)$ (c) $\delta < \frac{I}{2(I-1)}$

where $\overline{p}$, Δp, I, b, δ, H, and $\widehat{H}$ represent upper cost pricing limit, price decrement factor, total number of gateways, bid value, bid increment value, and the maximum number of iteration under normal and misbehaving scenario, respectively. The fulfilment of the criteria leads to get undue benefit by the misbehaving gateways in terms of revenue and cost. The first condition shows the lower bound of the number of iterations the auction process should run before the final allocation of bandwidth in presence of misbehaving gateway. The third condition shows the lower bound of the bid increment in each iteration by the gateway. Finally, the second condition shows the lower limit of final bid value for a gateway increasing the bid value by δ amount.

Unlike the centralized scenario, the distributed system does not sense misbehaviour due to the absence of information about all gateways. We theoretically prove the limitations of AQUM algorithm by Theorems 1, 2, and 3.

Theorem 1. *The total revenue received by the CSP reduces if a gateway misbehaves in the auction process.*

Proof. Let us consider the gateway G_i misbehaves while submitting bids by not increasing its bid value in subsequent iterations, and thus, the auction process concludes after $\widehat{H}$ number of iterations instead of H. In this case, the

CSP receives $R_s(\widehat{H}) = p(\widehat{H})\sum \mathbf{b}(\widehat{H}) - \mathcal{C}_{rent}$ units of revenue from the gateways. However, if the gateway G_i behaves normally, the revenue would be $R_s(H) = p(H)\sum \mathbf{b}(H) - \mathcal{C}_{rent}$. The other considerations for both scenarios remain the same, which include the equal initial bid value b for all gateways, bidding increment by δ value by a normal user, and the decrement of unit price Δp for a CSP in each iteration.

In case of a normal scenario, a gateway increases its bid value H number of times. We get the maximum value of H using the Lemma 4 in [2] as, $H = (B^*_{tot} - Ib)/(I\delta)$ where $B^*_{tot} = B_{tot} - \beta$, and β is the amount of reserved bandwidth used for internal works of the CSP. Similarly, we get the value of $\widehat{H}$ as

$$\widehat{H} = \frac{B^*_{tot} - Ib}{(I-1)\delta} = \frac{I}{I-1}H. \tag{4}$$

According to the revenue function of the CSP shown in Equation (3), $R_s(\widehat{H})$ and $R_s(H)$ are computed as: $R_s(\widehat{H}) = (\overline{p} - \widehat{H}\Delta p)\sum_{i=1}^{I} b_i(\widehat{H}) - \mathcal{C}_{rent}$, and $R_s(H) = (\overline{p} - H\Delta p)\sum_{i=1}^{I} b_i(H) - \mathcal{C}_{rent}$, and, then, we get,

$$R_s(H) - R_s(\widehat{H}) = (\widehat{H}\Delta p B^*_{tot})/I > 0. \tag{5}$$

This is because all the variables in Equation (5) are positive and $I > 1$. This ensures that the CSP always receives less revenue when a gateway misbehaves. □

Theorem 2. *A gateway pays less cost while it misbehaves, compared to the cost when it behaves normally.*

Proof. Let us consider the scenario as mentioned in Theorem (1). In this case, the misbehaving gateway pays $C_i(\widehat{H}) = p(\widehat{H})b_i(\widehat{H})$ units of cost to the CSP. However, if G_i behaves normally, the cost would be $C_i(H) = p(H)b_i(H)$. The other considerations for both scenarios remain the same.

According to the cost function modelling shown in Equation (2), $C_i(\widehat{H})$ and $C_i(H)$ are computed as: $C_i(\widehat{H}) = (\overline{p} - \widehat{H}\Delta p)b$ and $C_i(H) = (\overline{p} - H\Delta p)(b + H\delta)$. Then, we have,

$$\begin{aligned} &C_i(H) - C_i(\widehat{H}) \\ &= (b + H\delta)(\overline{p} - H\Delta p) - b\left(\overline{p} - \frac{I}{I-1}H\Delta p\right) \\ &= \frac{b\Delta p H}{I-1} + H\delta(\overline{p} - H\Delta p) > 0. \end{aligned} \tag{6}$$

This is because all the variables in Equation (6) are positive, $I > 1$, and $(\overline{p} - H\Delta p) \geq \underline{p} > 0$. □

Theorem 3. *Let the Conditions 1 (a) and (b) hold. The cost paid by a normal gateway in the presence of misbehaving gateway is greater than the cost paid in the absence of a misbehaving gateway.*

Proof. Let us consider that in the presence of a misbehaving gateway, the normal gateway G_j concludes the auction process with cost $C_j(\widehat{H})$ at the $\widehat{H}$th iteration, whereas, in its absence, the cost is $C_j(H)$. According to Equation (2), we compute $C_j(\widehat{H})$ and $C_j(H)$ as: $C_j(\widehat{H}) = (\overline{p} - \widehat{H}\Delta p)(b + \widehat{H}\delta)$ and $C_j(H) = (\overline{p} - H\Delta p)(b + H\delta)$. Then, we have,

$$\begin{aligned} C_j(\widehat{H}) - C_j(H) &= \frac{\widehat{H}}{I}\left[\overline{p}\delta - \Delta p b - \frac{2I-1}{I}\widehat{H}\Delta p\delta\right] \\ \Rightarrow [C_j(\widehat{H}) - C_j(H)] &> \frac{\widehat{H}}{I}\Delta p\delta\left[\frac{\overline{p}}{\Delta p} - \frac{b}{\delta} - 2\widehat{H}\right] > 0. \end{aligned} \tag{7}$$

This is because all the variables in Equation (7) are positive, $\widehat{H} > H$, and $I > 1$. This concludes the proof. □

4 Cheating-Resilient Bandwidth Distribution

In this section, we describe the modifications done in cost and revenue computation formulations followed the algorithm to find the cheat-proof optimal bandwidth allocation. We also prove the cheat-proof claims in subsequent sections using few Theorems and Lemmas.

4.1 Revised Utility Formulation of the Gateway

The generic approach considers the cost function as the multiplication of submitted bid and price per unit bandwidth allocation, as shown in Equation (2). However, for enforcing the gateway to act normally, we incorporate a penalty factor if there is any unbidden bandwidth. Unbidden bandwidth is the residual bandwidth that has not been bidden from the gateways. The penalty value increases with the increase in the amount of unbidden bandwidth, and is computed by simply dividing the maximum possible penalty by the amount of unbidden bandwidth at each iteration. Therefore, the *modified cost function* is as follows:

$$\begin{aligned} \widetilde{C}_i(b_i, p) = p(t)B_i &+ B^*_{tot} \\ &\times \left(\frac{\overline{p} + \underline{p}}{2}\right)\sum_{h=0}^{t-1}\left[\frac{1}{B^*_{tot} - \sum_{\forall j \neq i} b_j(h)}\right]. \end{aligned} \tag{8}$$

Hence, the modified *utility function* of G_i is

$$\begin{aligned} \widetilde{U}_i = s_i &- \frac{r_i B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}B_i} + q_i E_i B_i\{1-\alpha_i\} - p(t)B_i \\ &- B^*_{tot}\left(\frac{\overline{p} + \underline{p}}{2}\right)\sum_{h=0}^{t-1}\left[\frac{1}{B^*_{tot} - \sum_{\forall j \neq i} b_j(h)}\right]. \end{aligned} \tag{9}$$

We consider service delay as a utility maximization constraint for performing bandwidth redistribution. For providing QoS-guarantee in terms of service delay, the value of the service delay of gateway G_i should be less than or equal to the threshold value, i.e., $d_i \leq d_{\theta_i}$. Finally, we propose a utility maximization scheme using the revenue and cost functions of the gateway, as follows:

$$\begin{aligned} &maximize \quad \sum \widetilde{\mathbf{U}} \\ &subject\ to \quad d_i \leq d_{\theta_i},\ \forall i \in (1, I) \\ &\qquad\qquad\quad 0 < B_i \leq B_{tot},\ \text{and} \ \sum_i B_i = B^*_{tot}. \end{aligned} \tag{10}$$

4.2 Revised Revenue Function of the CSP

Besides optimal bandwidth distribution, a CSP also tries to maximize the revenue it receives from the gateways. Let the final allocation be $\mathbf{B}$, and the final price given by the CSP be

$p(t)$. The total revenue received by the CSP is computed as follows:

$$\widetilde{R}_s(\mathbf{B}, p(t)) = p(t)\sum\nolimits_{i=1}^{I} B_i + \sum\nolimits_{i=1}^{I}\left[B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\right.$$
$$\left.\times \sum\nolimits_{h=0}^{t-1}\left(\frac{1}{B^*_{tot} - \sum_{\forall j \neq i} b_j(h)}\right)\right] - \mathcal{C}_{rent}. \quad (11)$$

The first part of Equation (11) describes the revenue received the gateways by distributing the total bandwidth among them. The CSP charges an additional penalty for preventing the gateways from the misbehaving characteristic which is shown in the second part of Equation (11). Further, the CSP tries to distribute the bandwidth from its own quota. If the available bandwidth is not sufficient to fulfil the service requirements of the gateways, the CSP rents additional amount of required bandwidth from the third party sources. $\mathcal{C}_{rent}$ represents the cost for renting the bandwidth from the third party. At this juncture, it is pertinent to mention that, in this paper, we disregard how the CSP manages the bandwidth from the third party sources. We have plan to investigate this issue in the future.

4.3 CRAB Algorithm Design

In economic theory, the sellers and buyers use auction-based approaches to accomplish actual price discovery, winner determination, and payment management. In this work, we follow the descending price auction theory-based approach for designing the cheating-resilient and QoS-guaranteed bandwidth distribution algorithm, namely CRAB, in MCC. In this approach, the gateways are considered as buyers and the CSP acts as a seller-cum-auctioneer. In the auction process, we observe a trade-off between the unit price and the bandwidth request, as, in general, the amount of bandwidth request reduces with the increase of price per unit bandwidth allocation. For incorporating the trade-off condition, we modified the termination condition of the descending price auction. In our modified descending price auction process, the price decreases over time until the total bid reaches the total available bandwidth. In the interim, if the price p becomes less then $\underline{p}$, the ceiling price is reset again for continuing the auction process. In a real network, a gateway is not able to know the bid of the other gateways. Therefore, in addition to making the algorithm cheating-resilient, we design it for distributed system. The pseudo-code of the CRAB is presented in Algorithm 1, and its steps are explained below.

1. Initialization. Each gateway G_i knows its Shannon spectral efficiency E_i, protocol overhead α_i, and revenue per unit service delay r_i. We assume that r_i is determined based on the QoS-guarantee between the gateway and the connecting mobile users. The CSP computes the upper and lower pricing limits, and initializes the auction process by broadcasting the maximum price per unit allocation as $\overline{p}$. We explain the computation process of pricing limits in Theorem 8.

2. Bid. After receiving the announced price $p(t)$, each gateway G_i submits a bid $b_i(t)$ $(0 < b_i(t) \leq B_{tot})$ at time t, while satisfying the service delay constraint. Gateway computes the bid $b_i = \sum_{k=1}^{|N_i|} b_{ik} + \phi_i$, where ϕ_i represents the minimum requirement of bandwidth to maintain its own operations in the network, and the rest amount of demand value is required for the connected users. In the initial round, the CSP adjusts its total available bandwidth B_{tot}, only if the total initial bid exceeds the current availability. The CSP performs this arrangement by renting bandwidth from third party sources. Note that the renting option is applicable only at the initial stage. The next iteration onwards this situation is the exit condition from iteration, as mentioned in Algorithm 1.

Algorithm 1. CRAB Algorithm

Inputs: $\underline{p}$, $\overline{p}$, β, I, B_{tot}, Δp, δ
Outputs: B

```
1: CSP broadcasts price p(t) to all gateways. /* see Section 4.3:
   Initialization */
2: Gateway calculates b(t) and U(t). /* see Section 4.3: Bid */
3: for i := 1 to I do
4:   if (U_i(t) > U_i(t-1)) then
5:     Gateway G_i submits bid b_i(t)
6:   else
7:     Gateway G_i submits bid b_i(t-1)
8:   end if
9: end for
10: CSP adjusts B_tot, depending upon the initial bid. /* see Sec-
    tion 4.3: Bid */
11: if (sum_{i=1}^{I} b_i(t) <= B*_tot) then
12:   CSP revises the price p(t+1) := p(t) - Δp
13:   if (p(t+1) < p_) then
14:     CSP reset the price p(t+1) := p̄
15:   end if
16:   Goto Step 1 for next iteration
17: else
18:   CSP computes and allocates B to gateways. /* see
      Section 4.3: Allocation */
19:   CSP confirms the final cost C̃_i to gateways. /* see
      Section 4.3: Payment */
20: end if
```

In every iteration, each gateway checks its present utility, computed by using the expected incremented bid $b_i(t+1) = b_i(t) + \delta$, with the previous utility. Each gateway confirms its bid increment if the current utility value is higher than the previous utility value. Otherwise, the gateway sticks with the older bid value. In each intermediate iteration, the utility value is computed as follows:

$$U_i(t) = s_i - \frac{r_i B_{tot} \sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\} b_i(t)} + [q_i E_i\{1-\alpha_i\} - p(t)] b_i(t),$$

where $\{p(t) b_i(t)\}$ represents the cost of allocation in intermediate iteration.

3. Allocation. In each iteration, CSP aggregates all the bid values. If the aggregated value is greater than the available bandwidth B^*_{tot}, then the CSP concludes the auction process, and subsequently allocates B_i to the gateway G_i using Equation (12). Otherwise, the CSP decrements the price per unit allocation by Δp. If the current price per unit allocation reduces below $\underline{p}$, the CSP resets the price to the maximum value $\overline{p}$. For maintaining the total available bandwidth equal to the total allocated bandwidth, the B_i amount of bandwidth is allocated corresponding to the bid value $b_i(t)$. B_i is computed as follows:

$$\begin{aligned} B_i &= b_i(t) - \left[\sum\nolimits_{i=1}^{I} b_i(t) - B^*_{tot}\right] \\ &\quad \times \frac{[b_i(t) - b_i(t-1)] + \cdots + [b_i(1) - b_i(0)]}{\left[\sum_{i=1}^{I} b_i(t) - \sum_{i=1}^{I} b_i(t-1)\right] + \cdots + \left[\sum_{i=1}^{I} b_i(1) - \sum_{i=1}^{I} b_i(0)\right]} \\ &= b_i(t) - \left[\sum_{i=1}^{I} b_i(t) - B^*_{tot}\right]\left[\frac{b_i(t) - b_i(0)}{\sum_{i=1}^{I} b_i(t) - \sum_{i=1}^{I} b_i(0)}\right]. \end{aligned} \tag{12}$$

The final allocated bandwidth for the gateway G_i is the difference between its final bid value and a proportionate penalty. The proportionate penalty is computed by multiplying the ratio of bid increment with the total amount of overbid by the gateway G_i with respect to total increment by all gateways. In the proposed scheme, the total amount of overbid equals $(\sum_{i=1}^{I} b_i(t) - B^*_{tot})$, and the ratio of bid increment by the gateway G_i equals $[\{b_i(t) - b_i(0)\}/\{\sum_{i=1}^{I} b_i(t) - \sum_{i=1}^{I} b_i(0)\}]$.

4. *Payment*. The gateway G_i pays the final cost $\widetilde{C}_i$ to the CSP after receiving B_i units of bandwidth. The CSP computes $\widetilde{C}_i$ using the post-pay mechanism (discussed in Section 4.1) as follows:

$$\widetilde{C}_i = p(t)B_i + B^*_{tot}\left(\frac{\overline{p} + \underline{p}}{2}\right)\sum_{h=0}^{t-1}\left(\frac{1}{B^*_{tot} - \sum_{\forall j \neq i} b_j(h)}\right). \tag{13}$$

4.4 Cheat-Proof Theorem

Using the following Theorems and Lemmas, we prove that the proposed CRAB algorithm enforces the gateways to behave normally.

Theorem 4. *Let the Conditions 1 (a), (b), and (c) hold for the mobile-cloud system. The revenue received by the CSP is more in presence of misbehaving gateway compared to that in absence of misbehaving gateway.*

Proof. Let us consider the scenario as mentioned in Theorem 1. Due to the modified revenue function of the CSP as shown in Equation (11), at present, the modified revenues of the CSP $\widetilde{R}_s(\widehat{H})$ and $\widetilde{R}_s(H)$ are computed as follows:

$$\begin{aligned} \widetilde{R}_s(\widehat{H}) &= (\overline{p} - \widehat{H}\Delta p)\sum\nolimits_{i=1}^{I} b_i(\widehat{H}) - \mathcal{C}_{rent} \\ &\quad + IB^*_{tot}\left(\frac{\overline{p} + \underline{p}}{2}\right)\sum\nolimits_{h=0}^{\widehat{H}-1}\left(\frac{1}{B^*_{tot} - \sum_{\forall j \neq i} b_j(h)}\right) \\ \text{and } \widetilde{R}_s(H) &= (\overline{p} - H\Delta p)\sum\nolimits_{i=1}^{I} b_i(H) - \mathcal{C}_{rent} \\ &\quad + IB^*_{tot}\left(\frac{\overline{p} + \underline{p}}{2}\right)\sum\nolimits_{h=0}^{H-1}\left(\frac{1}{B^*_{tot} - \sum_{\forall j \neq i} b_j(h)}\right). \end{aligned}$$

Subtracting the value of $\widetilde{R}_s(\widehat{H})$ from $\widetilde{R}_s(H)$, and using Equation (4), we get,

$$\begin{aligned} &\widetilde{R}_s(\widehat{H}) - \widetilde{R}_s(H) \\ &= (\overline{p} - \widehat{H}\Delta p)B^*_{tot} - (\overline{p} - H\Delta p)B^*_{tot} + B^*_{tot}\left(\frac{\overline{p} + \underline{p}}{2}\right) \\ &\quad \times \left[\sum\nolimits_{h=0}^{\widehat{H}-1}\left(\frac{1}{B^*_{tot} - (I-1)(b + h\delta)}\right)\right. \\ &\quad + \sum\nolimits_{h=0}^{\widehat{H}-1}\left(\frac{I-1}{B^*_{tot} - (I-1)b - (I-2)h\delta}\right) \\ &\quad \left. - \sum\nolimits_{h=0}^{H-1}\left(\frac{I}{B^*_{tot} - (I-1)(b + h\delta)}\right)\right] \\ &> B^*_{tot}\left[\frac{\widehat{H}}{I}\left\{\Delta p \Upsilon_1 + \frac{\underline{p}}{2(b + H\delta)}\right\}\right. \\ &\quad \left. + \left(\frac{\overline{p} + \underline{p}}{2}\right)(I-1)\Upsilon_2\right], \end{aligned}$$

where, $\Upsilon_1 = I\widehat{H}/(b + H\delta) - 1$, and $\Upsilon_2 = \sum_{h=0}^{\widehat{H}-1}\left(\frac{1}{b + \{IH - (I-2)h\}\delta}\right) - \sum_{h=0}^{H-1}\left(\frac{1}{b + \{IH - (I-1)h\}\delta}\right)$. Now, the theorem would be proved, if it can be proved that both Υ_1 and Υ_2 are positive. First, we prove that $\Upsilon_1 > 0$, using the method of contradiction. Let us consider $\Upsilon_1 < 0$. Hence, we get,

$$\{I\widehat{H}/(b + H\delta)\} < 1 \Rightarrow IH < b. \tag{14}$$

The inequality in Equation (14) contradicts Conditions 1 (b) and (c). Hence, our consideration is wrong, i.e., $\Upsilon_1 > 0$.

Second, we prove that $\Upsilon_2 > 0$, for all $H > 1$ and $I > 1$, using induction theory. We see that the inequality is true for $H = 2$ and 3. Therefore, using induction hypothesis, it may be inferred that the inequality is true for $H = \eta$ where η is a non-negative integer. Hence, we get,

$$\begin{aligned} &\left[\sum\nolimits_{h=0}^{\{I\eta/(I-1)\}-1}\left(\frac{1}{b + \{I\eta - (I-2)h\}\delta}\right)\right. \\ &\quad \left. - \sum\nolimits_{h=0}^{\eta-1}\left(\frac{1}{b + \{I\eta - (I-1)h\}\delta}\right)\right] > 0. \end{aligned} \tag{15}$$

The inequality is true for any value of $H > 1$, if it is true for $H = \eta + 1$ too, where $\eta \in \mathbb{Z}^+$. We rewrite the equation of Υ_2 for $H = \eta + 1$, as follows:

$$\begin{aligned} \Upsilon_2 &= \sum\nolimits_{h=0}^{\{I(\eta+1)/(I-1)\}-1}\left(\frac{1}{b + \{I(\eta+1) - (I-2)h\}\delta}\right) \\ &\quad - \sum\nolimits_{h=0}^{\eta}\left(\frac{1}{b + \{I(\eta+1) - (I-1)h\}\delta}\right). \end{aligned} \tag{16}$$

If equal number of terms, say ρ, is considered from the starting values of h for the expression of Υ_2, where $H = \eta$ and $H = \eta + 1$, respectively, then we see that,

$$\sum_{h=0}^{\rho}\Upsilon_2(H = \eta + 1) > \sum_{h=0}^{\rho}\Upsilon_2(H = \eta). \tag{17}$$

Applying Equations (15) and (17) on (16) we get,

$$\begin{aligned}\Rightarrow \Upsilon_2 > &\sum_{h=I\eta/(I-1)}^{\{I(\eta+1)/(I-1)\}-1}\left(\frac{1}{b+\{I(\eta+1)-(I-2)h\}\delta}\right)\\ &-\left(\frac{1}{b+\{I(\eta+1)-(I-1)\eta\}\delta}\right)\\ &+\sum_{h=0}^{\{I\eta/(I-1)\}-1}\left(\frac{1}{b+\{I\eta-(I-2)h\}\delta}\right)\\ &-\sum_{h=0}^{\eta-1}\left(\frac{1}{b+\{I\eta-(I-1)h\}\delta}\right)\\ &> \frac{I/(I-1)}{b+\{I\eta/(I-1)+I\}\delta}-\frac{1}{b+\{I+\eta\}\delta} > 0.\end{aligned} \tag{18}$$

This concludes the proof. □

Theorem 5. *Let the Conditions 1 (a), (b), and (c) hold for a mobile-cloud system. The cost paid by a misbehaving gateway is greater than the cost it pays when it behaves normally while executing the CRAB.*

Proof. Let us consider the gateway G_i misbehaves. In this case, the selfish gateway pays $\widetilde{C}_i(\widehat{H})$ cost to the CSP instead of $\widetilde{C}_i(H)$ which is paid while it behaves normally. Normally, a gateway increases its bid value by δ in each iteration which is not followed while it misbehaves. In CRAB, the CSP begins with price per unit allocation $\overline{p}$, and then its value is decremented by Δp in each subsequent iteration. The final allocation $B_i(H)$ and $B_i(\widehat{H})$ at time slot H and $\widehat{H}$, respectively, are computed using Equation (12), as follows:

$$B_i(H) = b + H\delta \tag{19}$$

$$\text{and, } B_i(\widehat{H}) = b. \tag{20}$$

According to the modified cost function, shown in Equation (8), $\widetilde{C}_i(\widehat{H})$ and $\widetilde{C}_i(H)$ are computed as

$$\begin{aligned}\widetilde{C}_i(\widehat{H}) &= \{\overline{p}-\widehat{H}\Delta p\}B_i(\widehat{H}) + \left[B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\right.\\ &\quad\left.\times\sum_{h=0}^{\widehat{H}-1}\left(\frac{1}{B^*_{tot}-\sum_{\forall j\neq i} b_j(h)}\right)\right]\text{and,}\widetilde{C}_i(H)\\ &= \{\overline{p}-H\Delta p\}B_i(H) + \left[B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\right.\\ &\quad\left.\times\sum_{h=0}^{H-1}\left(\frac{1}{B^*_{tot}-\sum_{\forall j\neq i} b_j(h)}\right)\right].\end{aligned}$$

Subtracting the value of $\widetilde{C}_i(H)$ from $\widetilde{C}_i(\widehat{H})$, and using Equations (19) and (20), we get,

$$\begin{aligned}&\widetilde{C}_i(\widehat{H}) - \widetilde{C}_i(H)\\ &= (\overline{p}-\widehat{H}\Delta p)b - (\overline{p}-H\Delta p)(b+H\delta)\\ &\quad + B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\sum_{h=0}^{\widehat{H}-1}\left(\frac{1}{B^*_{tot}-(I-1)(b+h\delta)}\right)\\ &\quad - B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\sum_{h=0}^{H-1}\left(\frac{1}{B^*_{tot}-(I-1)(b+h\delta)}\right)\\ &> \left[B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\frac{\widehat{H}}{I(b+H\delta)} - \frac{\widehat{H}}{I}\Delta pb - (\overline{p}-H\Delta p)H\delta\right]\\ &> \frac{HI\underline{p}}{2(I-1)} > 0.\end{aligned}$$

This concludes that the selfish gateway has to pay more cost using the CRAB algorithm. □

Theorem 6. *Let the Conditions 1 (a), (b), and (c) hold for the mobile-cloud system. The cost paid by the normal gateway in presence of a misbehaving gateway increases compared to the cost paid in the absence of a misbehaving gateway, while executing the CRAB algorithm.*

Proof. In the presence of a misbehaving gateway, a normal gateway G_j receives the bandwidth distribution at the $\widehat{H}$th iteration with cost $\widetilde{C}_j(\widehat{H})$, whereas, the cost is $\widetilde{C}_j(H)$ in the absence of the misbehaving gateway. The final allocation $B_j(H)$ and $B_j(\widehat{H})$ are computed using Equation (12), as follows:

$$B_j(H) = b + H\delta \tag{21}$$

$$\text{and, } B_j(\widehat{H}) = b + \widehat{H}\delta. \tag{22}$$

According to the modified cost function, shown in Equation (8), $\widetilde{C}_j(\widehat{H})$ and $\widetilde{C}_i(H)$ are computed as

$$\begin{aligned}\widetilde{C}_j(\widehat{H}) &= \{\overline{p}-\widehat{H}\Delta p\}B_j(\widehat{H}) + \left[B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\right.\\ &\quad\left.\times\sum_{h=0}^{\widehat{H}-1}\left(\frac{1}{B^*_{tot}-\sum_{\forall i\neq j} b_i(h)}\right)\right]\\ \text{and, }\widetilde{C}_j(H) &= \{\overline{p}-H\Delta p\}B_j(H) + \left[B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\right.\\ &\quad\left.\times\sum_{h=0}^{H-1}\left(\frac{1}{B^*_{tot}-\sum_{\forall i\neq j} b_i(h)}\right)\right].\end{aligned}$$

Subtracting the value of $\widetilde{C}_j(H)$ from $\widetilde{C}_j(\widehat{H})$, and using Equations (16), (21), and (22), we get,

$$\begin{aligned}&\widetilde{C}_j(\widehat{H}) - \widetilde{C}_j(H)\\ &= (\overline{p}-\widehat{H}\Delta p)(b+\widehat{H}\delta) - (\overline{p}-H\Delta p)(b+H\delta)\\ &\quad + B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\sum_{h=0}^{\widehat{H}-1}\left(\frac{1}{B^*_{tot}-(I-1)b-(I-2)h\delta}\right)\\ &\quad - B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\sum_{h=0}^{H-1}\left(\frac{1}{B^*_{tot}-(I-1)(b+h\delta)}\right)\\ &= \Upsilon_3 + B^*_{tot}\left(\frac{\overline{p}+\underline{p}}{2}\right)\Upsilon_2,\end{aligned} \tag{23}$$

where, $\Upsilon_3 = (\overline{p}-\widehat{H}\Delta p)(b+\widehat{H}\delta) - (\overline{p}-H\Delta p)(b+H\delta)$. Using Theorem (3) and Equation (18), we get $\Upsilon_3 > 0$ and $\Upsilon_2 > 0$. Hence, in Equation (23), we get, $[\widetilde{C}_j(\widehat{H}) - \widetilde{C}_j(H)] > 0$. This concludes the proof. □

5 Theoretical Analysis

At the outset, we analyse the complexity of the proposed algorithm. We derive the pricing boundary of bidding, and prove that the designed utility function is convex in nature under certain conditions. We investigate the presence of Nash Equilibrium and derive the optimal solution using the Lagrange's method.

Theorem 7. *The time and space complexities of the proposed algorithm CRAB are same with those of the benchmark protocol, AQUM [2], which equals $O(\frac{\overline{p}}{\Delta p})$ and $O(1)$, respectively.*

Proof. According to the auction policy, the bidding value increases by a positive constant δ in every iteration. However, the misbehaving gateway does not follow this usual increment. Let us consider that the initial bid for every gateway is b, and that there are I gateways in the system. Let us consider that the algorithm terminates at the $\widehat{H}$th iteration, Hence, the total bid value at the final iteration is $Ib + \widehat{H}(I-1)\delta$. On the other hand, the algorithm terminates only when the total bid value exceeds the available bandwidth. From this condition we get, $\widehat{H} = \frac{B^*_{tot} - Ib}{(I-1)\delta}$. Further, we know that $\delta < \frac{I}{2(I-1)}$, as mentioned in Section 3.4. Therefore, $\widehat{H} \geq 2(\frac{B^*_{tot}}{I} - b)$. Again, we know that $\frac{\overline{p}}{\Delta p} > 2I\widehat{H}$, as mentioned in Section 3.4. Hence, $\widehat{H} < \frac{\overline{p}}{\Delta p}\frac{1}{2I}$. Additionally, each iteration takes I times to compute the bidding value (see the *for* loop in Algorithm 1). Hence, the time complexity of the proposed algorithm is $O(\widehat{H}I)$, i.e., $O(\frac{\overline{p}}{\Delta p})$, where Δp is a constant value and the computation of $\overline{p}$ is explained in Theorem 8. Under normal scenario, every gateway increases its corresponding bid value after each iteration. Let us consider that the algorithm terminates at Hth iteration, and, then, we get $H = \frac{B^*_{tot} - Ib}{(I\delta)}$. Hence, the time complexity in the normal scenario is the same as that in the abnormal scenario in CRAB, which is also the same as that exhibited by the benchmark protocol, AQUM [2].

To execute the proposed algorithm, the system does not need any additional space except the protocol parameters that are needed to be stored temorarily in run-time. The requred space is also not dependent on the number of gateways. Therefore, the space complexity of the proposed algorithm is $O(1)$, which is the same as that of the benchmark protocol, AQUM [2]. □

Theorem 8. *The lower and upper pricing boundaries of p for a user $i \in (1, I)$ are expressed as follows, respectively:*

$$\underline{p}_i = q_i E_i(1-\alpha_i) + \frac{r_i B_{tot} \sum_{k=1}^{|N_i|} T_{ik}}{E_i(1-\alpha_i)(B^*_{tot})^2} \tag{24}$$

$$\overline{p}_i = q_i E_i(1-\alpha_i) + \frac{r_i B_{tot} \sum_{k=1}^{|N_i|} T_{ik}}{E_i(1-\alpha_i)\phi^2} \tag{25}$$

Proof. After successful completion of the CRAB algorithm, the CSP allocates at least ϕ_i amount of bandwidth to the gateway G_i as ϕ_i is the minimum requirement of that gateway for its own purpose, where $\phi_i > 0$. Let us consider that $\phi_i = \phi, \forall i$. On the other hand, if there is no other request except from G_i, then the CSP allocates the total available bandwidth, i.e., B^*_{tot} to that gateway. Therefore, it is inferred that the value of B_i ranges in between ϕ and B^*_{tot}, i.e., $\phi \leq B_i \leq B^*_{tot}$. Taking the first derivative of the utility function shown in Equation (9), with respect to the distribution B_i, we get,

$$\frac{\partial \widetilde{U}_i}{\partial B_i} = q_i E_i(1-\alpha_i) + \frac{r_i B_{tot} \sum_{k=1}^{|N_i|} T_{ik}}{E_i(1-\alpha_i) B_i^2} - p_i.$$

As the optimum value exists at $\frac{\partial \widetilde{U}_i}{\partial B_i} = 0$, we get,

$$p_i = q_i E_i(1-\alpha_i) + \frac{r_i B_{tot} \sum_{k=1}^{|N_i|} T_{ik}}{E_i(1-\alpha_i) B_i^2}. \tag{26}$$

In Lemma (1), we prove that p is a decreasing function with respect to B_i. Therefore, substituting the minimum and maximum values of B_i in Equation (26), we get the values of $\overline{p}_i$ and $\underline{p}_i$ for a user $i \in (1, I)$ as mentioned in Equations (24) and (25), respectively. □

Lemma 1. *The confirm price p per unit bandwidth allocation is a decreasing function with respect to B_i.*

Proof. Taking the first derivative of the price function, as derived in Theorem 8 (shown in Equation (26)), with respect to B_i, we get,

$$\frac{\partial p}{\partial B_i} = -\frac{2 r_i B_{tot} \sum_{k=1}^{|N_i|} T_{ik}}{E_i(1-\alpha_i)(B_i)^3} < 0. \tag{27}$$

Therefore, we conclude that p is a decreasing function with respect to B_i. This concludes the proof. □

Lemma 2. *The utility function in Equation (9) is continuous over the interval $0 < B_i \leq B^*_{tot}$.*

Proof. Let $f(x)$ be a real-valued function defined on a subset X of real numbers $\mathbb{R}$, such that $f : X \to \mathbb{R}$. Then, $f(x)$ is said to be continuous at a point $\lambda_0 \in X$, if for any $\epsilon > 0$ there exists $\mu > 0$ such that for all $x \in X$ with $|x - \lambda_0| < \mu$, the inequality $|f(x) - f(\lambda_0)| < \epsilon$ is valid. Likewise, we have a utility function $\widetilde{U}_i$, a real-valued function defined on a subset Y of real numbers $\mathbb{R}$. Specifically, $Y \in (\phi, B^*_{tot})$, where $\phi > 0$. We assume that there exists a $\mu > 0$ such that for all $B_i \in Y$, and $\lambda_0 \in Y$, the inequality $|B_i - \lambda_0| < \mu$ is valid. Considering the equal penalty factor, we get,

$$\begin{aligned} &\left|\widetilde{U}_i(B_i, p) - \widetilde{U}_i(\lambda_0, p)\right| \\ &= \left|q_i E_i(1-\alpha_i)(B_{tot} - \lambda_0) + \Omega - p(B_i - \lambda_0)\right| \\ &< \epsilon \end{aligned}$$

$$\begin{aligned} &\text{where, } \epsilon = q_i E_i(1-\alpha_i)(B_{tot} - \lambda_0) + \Omega \\ &\text{and } \Omega = \frac{r_i B_{tot} \sum_{k=1}^{|N_i|} T_{ik}(B_i - \lambda_0)}{B_i \lambda_0 E_i(1-\alpha_i)}. \end{aligned} \tag{28}$$

As $B_i \in (\phi, B^*_{tot}), \forall i \in (1, I)$, all the terms in Equation (28) are positive. Hence, $\epsilon > 0$. Therefore, we conclude that $\widetilde{U}_i$ is a continuous function over the distribution (ϕ, B^*_{tot}), where $\phi > 0$. □

Theorem 9. *There exists Nash Equilibrium for every gateway's individual distribution B_i.*

Proof. The utility function of gateway G_i is computed by Equation (9). Taking the second derivative of the utility function with respect to B_i, we get,

$$\frac{\partial^2 U_i}{\partial B_i^2} = -\frac{2 r_i B_{tot} \sum_{k=1}^{|N_i|} T_{ik}}{E_i(1-\alpha_i)(B_i)^3} < 0. \tag{29}$$

This implies that the utility function of each gateway is a concave function with respect to its own distribution B_i. Further, we have already proved in Lemma (2) that the utility function of each gateway is continuous in the range (ϕ, B_{tot}). On the other hand, Rosen [27] proved the existence of Nash equilibrium point for every concave n-person game. As the utility function in the proposed model is also a concave function while multiple players playing the auction, we implies that there exists Nash Equilibrium in the proposed model too. □

Lemma 3. *The optimization problem in Equation (10) is convex with respect to the distribution B_i.*

Proof. We have already proved in Theorem 9 (Equation (29)) that the utility function in Equation (9) is concave with respect to the distribution B_i. For investigating the convexity of the constrained functions used in Equation (10), we rearrange the delay condition as

$$d_i^e = \frac{B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i(1-\alpha_i)B_i} - d_{\theta_i} \leq 0. \tag{30}$$

Taking the second derivative of the Equation (30) with respect to distribution B_i, we get,

$$\frac{\partial^2 d_i^e}{\partial B_i^2} = \frac{2B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i(1-\alpha_i)(B_i)^3} > 0. \tag{31}$$

This implies that the delay equation is convex with respect to the distribution B_i. Similarly, we can prove that the other constrained functions in Equation (10) are also convex. Therefore, we conclude that the optimization problem is convex. □

Theorem 10. *The optimization problem in Equation (10) provides optimal solution with respect to* $\mathbf{B}^*$.

Proof. The convergence property and the existence of Nash Equilibrium show that the proposed algorithm concludes in finite number of iterations, and, thus, converges to a solution $\mathbf{B}^* = (B_1^*, B_2^*, \ldots, B_I^*)$. Since the optimization problem in Equation (10) is convex with respect to B_i while the other parameters are constant, we find the optimal value of B_i by solving the Karush-Kuhn-Tucker (KKT) conditions [28]. The Lagrangian representation of the optimization problem in Equation (10) is defined as,

$$\begin{aligned}
&\mathcal{L}(B_i, \lambda_i, \omega_i, \varphi_i, \xi) \\
&= -\sum_{i=1}^{I}\Bigg[s_i - \frac{r_i B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}B_i} + q_i E_i B_i\{1-\alpha_i\} \\
&\quad - p(t)B_i + \Delta p \sum_{h=0}^{t-1}\Big\{B_{tot}^* - \sum_{\forall j \neq i} b_j(h)\Big\}\Bigg] \\
&\quad + \sum_{i=1}^{I}\lambda_i\left[\frac{B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}B_i} - d_{\theta_i}\right] + \sum_{i=1}^{I}\omega_i[B_i - B_{tot}] \\
&\quad - \sum_{i=1}^{I}\varphi_i B_i + \xi\left[\sum_{i=1}^{I} B_i B_{tot}^*\right].
\end{aligned}$$

Then, the corresponding KKT conditions are as follows:

$$\begin{aligned}
&(i)\ p(t) - \frac{r_i B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}B_i^2} - q_i E_i\{1-\alpha_i\} \\
&\qquad - \frac{\lambda_i B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}B_i^2} + \omega_i - \varphi_i + \xi = 0 \\
&(ii)\ \lambda_i\left[\frac{B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}B_i} - d_{\theta_i}\right] = 0, \forall i = 1, \ldots, I \\
&(iii)\ \omega_i[B_i - B_{tot}^*] = 0, \forall i = 1, \ldots, I \\
&(iv)\ -\varphi_i B_i = 0, \forall i = 1, \ldots, I \\
&(v)\ \xi\left[\sum_{i=1}^{I} B_i B_{tot}^*\right] = 0 \\
&(vi)\ \frac{B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}B_i} \leq d_{\theta_i} \\
&(vii)\ 0 < B_i \leq B_{tot}^* \\
&(viii)\ \sum_{i=1}^{I} B_i = B_{tot}^*.
\end{aligned}$$

By replacing the above conditions (ii)-(vi) in the condition (i), we get,

$$\begin{aligned}
&p(t) - \frac{r_i B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}B_i^2} - q_i E_i\{1-\alpha_i\} = 0 \\
&\Rightarrow B_i = \sqrt{\frac{r_i B_{tot}\sum_{k=1}^{|N_i|} T_{ik}}{E_i\{1-\alpha_i\}[p(t) - q_i E_i\{1-\alpha_i\}]}}.
\end{aligned} \tag{32}$$

For avoiding ambiguity, we rename the parameter B_i in Equation (32) by $\widehat{B}_i$. Further, we know that the minimum requirement of bandwidth allocation for gateway G_i equals ϕ_i as mentioned in Theorem 8. Hence, we get the optimal solution for B_i using the inequality conditions (vii)-(viii) as follows:

$$B_i^* = max[\phi, min\{\widehat{B}_{tot}^*, \widehat{B}_i\}],$$

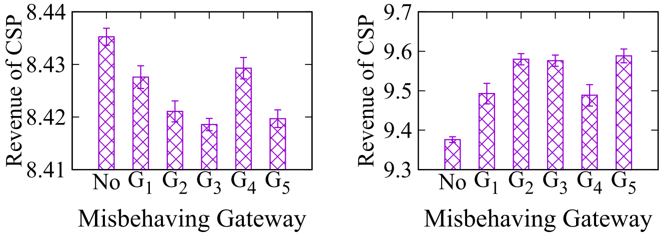
where,

$$\widehat{B}_{tot}^* = \sum_{i=1}^{I} max[\phi, min\{B_{tot}^*, \widehat{B}_i\}].$$

Finally, we conclude that $(B_1^*, B_2^*, \ldots, B_I^*)$ is the solution that maximizes the total utility of all the gateways. □

6 Numerical Results

In this section, we present the numerical simulation results of the proposed CRAB algorithm and compare its performance with the benchmark protocol AQUM [2]. The notations AQUM-No, AQUM-Yes, CRAB-No, and CRAB-Yes, are adopted in the legends of the figures to define the absence (-No) and presence (-Yes) of misbehaving gateways, in the AQUM and CRAB schemes, respective. For evaluating the effect of misbehaviour, throughout the experiments, we initially considered that only one gateway misbehaves at a time while the others behave normally. We further extend our evaluation by considering the scenarios with multiple misbehaving gateways. For plotting the figures, each experiment was executed 30 times, and the ensemble average values are plotted with 95 percent confidence interval.



(a) AQUM Algorithm (b) CRAB Algorithm

Fig. 2. Revenue received by the CSP.

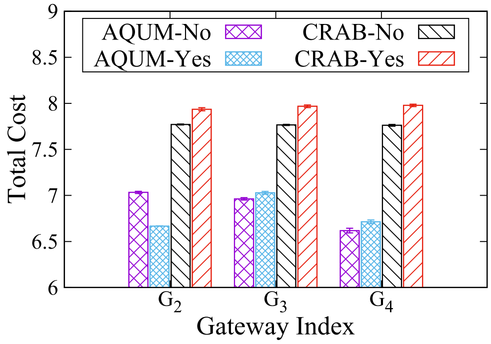


Fig. 3. Cost paid by the gateway.

6.1 Parameter Settings

Let us consider a MCC environment, as shown in Fig. 1, with one CSP and five gateways ($G_1, \ldots, G_5$). Each gateway G_i has two connected mobile users (N_{i1}, N_{i2}). We consider the following numerical values for evaluation. The total bandwidth $B_{tot} = 100$ Mbps, reserved bandwidth for CSP $\beta = 2$ Mbps, upper pricing limit $\overline{p} = 48$, lower pricing limit $\underline{p} = 10$, revenue per unit transmission rate $q_i = 30$ and revenue per unit service delay $r_i = 30$ for all gateway G_i, $i \in I$. In all the experiments, we consider SNR to be equal to 10 dB and the ideal transmission time was considered to be equal to 0.4 sec for each gateway. The target BER and protocol overhead α of all gateways are different and are represented by the sets BER = $\{10^{-5}, 10^{-4}, 5 \times 10^{-5}, 10^{-5}, 5 \times 10^{-5}\}$ and α = $\{0.02, 0.008, 0.008, 0.008, 0.02\}$, respectively. In this work, we consider the overhead according to the Transport Control Protocol (TCP) and the User Datagram Protocol (UDP). We compute the channel spectral efficiency E_i using a method described in [2], [29]. Further, we consider the Additive White Gaussian Noise (AWGN) channel model for which the spectral efficiency computation process is described in [29]. The value of the shifting parameter s depends on the delay threshold $\mathbf{d}_\theta = \{2.5, 1.5, 1.75, 2.25, 2.0\}$ and the revenue per unit service delay. It is pertinent to mention that the values of parameters such as B_{tot}, β, q_i, and r_i are assumed randomly as a test case to represent the performance metrics graphically. However, we have also tested with other combinations of values for observing the behaviour of the proposed scheme. Additionally, we mention that the solution is not limited to small number of gateways. For understanding scalability, we evaluate the CRAB algorithm for varying the number of gateways, and the results are explained in Section 6.7. We observed that an increase in number of gateways does not contribute any additional result except the increment of computational complexity.

6.2 Revenue Received by the CSP

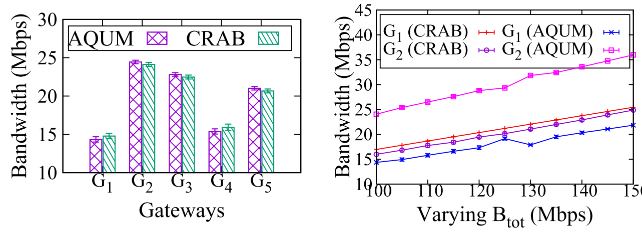
In this experiment, we measured the revenue received by the CSP from all gateways under normal and misbehaving behaviour of all the gateways one at a time. We performed the experiment for both the algorithms AQUM and CRAB, and plotted the received results in Figs. 2a and 2b, respectively, using the logarithmic scale. Fig. 2a shows that the total revenue accumulated by the CSP in presence of a misbehaving gateway is less than that in absence of it. This observation indicates that the distribution process is not favourable for the CSP if there is any misbehaving gateway. The CRAB algorithm overcomes this limitation, as shown in the Fig. 2b. Further, the revenue of the CSP varies with different misbehaving gateway. This is because of different spectral efficiency and protocol overhead corresponding to the gateways. This experiment establishes the correctness of Theorems 1 and 4.

6.3 Total Cost Paid by Each Gateway

In this experiment, we computed the total cost paid by each gateway for both scenarios—*presence* and *absence* of a misbehaving gateway. We performed the same experiment for both algorithms: AQUM and CRAB. Let us consider that the gateway G_2 misbehaves among the five gateways, while the others maintain normal behaviour. In Fig. 3, we only plot the computed cost (in log scale) corresponding to three gateways as the representative of misbehaving (G_2) and normal (G_3, G_4) gateways. From this figure, we observed few important features of the algorithms. First, the total cost paid by the misbehaving gateway reduces in the AQUM algorithm, which establishes the correctness of Theorem 2. Second, the total cost of the misbehaving gateway increases compared to that in case of its normal behaviour in the CRAB algorithm. Such negative effect enforces each gateway not to behave abnormally, and, thus, establishes the correctness of Theorem 5. Finally, the cost paid by any normal gateway increases if a gateway misbehaves. This is true for both the algorithms, and satisfies Theorems 3 and 6.

6.4 Bandwidth Distribution

We compared the final bandwidth distribution under both the normal and the abnormal scenario using both the AQUM and the CRAB algorithms. Comparison of allocation in the normal scenario for all the five gateways are plotted in Fig. 4a. We observed that the bandwidth distribution is significantly similar for both the algorithms. However, in Fig. 4b, we plotted the effective bandwidth allocation using both the algorithms under misbehaving condition. In this experiment, we also varied the amount of total available



(a) Using both the algorithms under normal scenario (b) Using both the algorithms under misbehaving scenario

Fig. 4. Comparison of final bandwidth distribution.

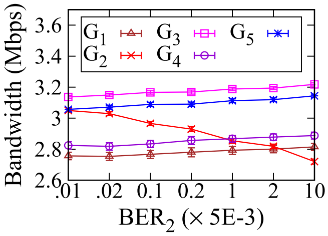


Fig. 5. Nash equilibrium in CRAB Algorithm.

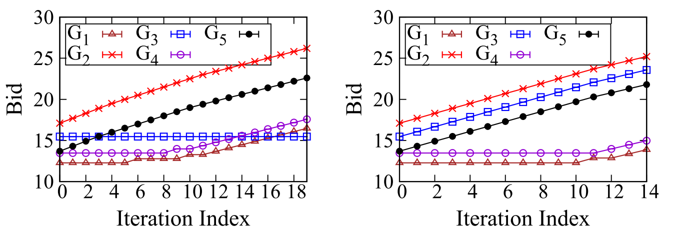
bandwidth to observe its impact on allocation. Let us consider that the gateway G_2 misbehaves. We only plot the allocation corresponding to G_2 and G_1 which represents the behaviour of abnormal and normal gateways, respectively. We observed that, irrespective of the total available bandwidth, the misbehaving gateway gets higher allocation compared to that obtained by the normal gateway, using the benchmark algorithm AQUM. However, this is not possible in the proposed algorithm CRAB. The misbehaving gateway would get less bandwidth compared to that in normal gateways as shown in the figure. Therefore, the gateways are forced to behave normally. This experiment indicates that the CRAB algorithm makes the bandwidth distribution scheme more realistic by making it distributed and cheat-proof.

6.5 Nash Equilibrium

We measured the final bandwidth distribution for all the gateways with varying channel quality. We varied the values of target BER of the gateway G_2, while the other parameters have set to default values. We observe from Fig. 5 that the allocated bandwidth of the gateway G_2 decreases with the increase in the value of target BER, and the bandwidth distribution trajectories of the gateways intersect at different points. This intersecting behaviour indicates that there exists Nash Equilibrium in the proposed algorithm, CRAB.

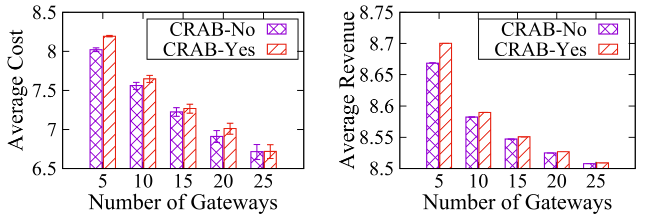
6.6 Convergence

In this experiment, we test the convergent criteria of CRAB under both the scenarios—*presence* and *absence* of misbehaving gateway. We performed this test with the default parameter settings in which gateway G_3 misbehaves for the first experiment. Figs. 6a and 6b show the bid value of each gateway in each iteration until the CRAB algorithm converges. Therefore, the bid value of G_3 remains the same for the first experiment, as shown in Fig. 6a. These figures show that the CRAB algorithm converges at iteration index 19 and 14, respectively, in the presence and absence of a misbehaving gateway. The above observations also comply with our theoretical calculation of $\widehat{H}$ and H, in which $\widehat{H} > H$.



(a) In presence of misbehaviour (b) In absence of misbehaviour

Fig. 6. Convergence of CRAB algorithm in the presence and absence of misbehaving gateway.



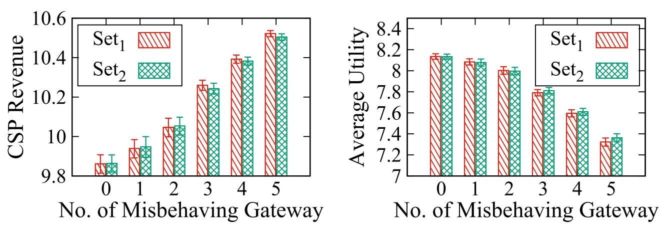
(a) Average cost paid by normal gateway (b) Average revenue of a gateway except misbehaved one

Fig. 7. Average revenue and cost under varying number of gateways.

6.7 Scalability

In this section, we extend our experiment to analyse the scalability of the proposed algorithm in MCC. We vary the number of gateways in multiples of five. The corresponding parameter settings described in Section 6.1 are also repeated accordingly, except the delay threshold $\mathbf{d}_\theta$. New delay threshold for the set of five gateways are $\{11.5, 10.5, 10.75, 11.25, 11.0\}$ *sec*. Using the above settings, we computed the average cost paid by the gateways for both scenarios—*presence* and *absence* of misbehaving gateway. Let us consider that the gateway G_1 misbehaves. The received results are plotted in Fig. 7a using logarithmic scale. From the figure, we can conclude that the cost paid by a normal gateway increases if any gateway misbehaves. However, the difference between the average cost in presence and absence of misbehaving gateway decreases with the increase in the number of gateways due to the fixed amount of total bandwidth. We measured the average revenue received by the normal gateways in the same experiment. Fig. 7b shows that the average revenue is less when all the gateways behave normally for any number of gateways. Further, the difference between the average revenue in presence and absence of misbehaving gateway decreases with the increase in the number of gateways due to less number of iteration for bandwidth distribution when the number of gateways is high. Hence, we conclude that the CRAB algorithm prevents the gateways to behave abnormally in bandwidth distribution process.

We further extend our work for varying number of misbehaving gateways. Let us consider that there are 10 gateways, and we do experiment for two settings: $\text{Set}_1 = \{\alpha_2 = 0.008,\ \ \text{BER}_2 = 10^{-4}\}$, $\text{Set}_2 = \{\alpha_3 = 0.98,\ \ \text{BER}_2 = 5 \times 10^{-5}\}$. The other parameters are set to their default values. We vary the number of misbehaving gateways up to five and compute the average utility of the gateways and the revenue of the CSP. For executing the experiment, we consider that the gateways $G_1, G_6, G_2, G_7,$ and G_3 misbehave. Figs. 8a and 8b show that the revenue of a CSP increases, and the average utility of the gateways decreases,



(a) Revenue of CSP (b) Average utility of the gateways

Fig. 8. Results under varying number of misbehaving gateways for two set of parameter settings.

as the number of misbehaving gateways increase for both the sets of data. Thus, the misbehaving characteristic increases the revenue of the CSP and decreases the utility of the gateway. Hence, we conclude that, irrespective of protocol overhead and channel status, the proposed algorithm prevents the gateways from behaving abnormally, as abnormal behaviour increases their own loss.

7 Conclusion

In this paper, we have identified and addressed the problem of revenue loss of a CSP in the presence of misbehaving gateway while it distributes bandwidth among the requesting gateways located in a MCC environment. We have formulated the problem as a constrained convex utility maximization problem, and solved it using the method of Lagrange multiplier for obtaining the optimal solution. We have proposed CRAB—a cheating-resilient bandwidth distribution algorithm for proportionately distributing the total bandwidth among the mobile devices. The advantages of CRAB are two fold. First, CRAB is a distributed algorithm. For the execution of the auction process, a CSP does not need total information from all its users. Second, the algorithm is capable of properly distributing resources in the presence of a misbehaving gateway. We validated the objectives through numerical analysis and results. We also investigated the existence of Nash Equilibrium and the convergence criteria of CRAB. We extended our test for multiple misbehaving users.

In this work, we have only considered single-hop connection between the gateway and the user. In the future, we also envision to work on an architecture consisting of multi-hop connectivity between a gateway and the mobile users. We also plan to study mixed-traffic based cheat-proof resource allocation in the future.

References

[1] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," *Wireless Commun. Mobile Comput.*, vol. 13, no. 18, pp. 1587–1611, Dec. 2013.

[2] S. Misra, S. Das, M. Khatua, and M. S. Obaidat, "QoS-guaranteed bandwidth shifting and redistribution in mobile cloud environment," *IEEE Trans. Cloud Comput.*, vol. 2, no. 2, pp. 181–193, Apr. 2014.

[3] H. Hu, Z. Li, and H. Hu, "An anti-cheating bidding approach for resource allocation in cloud computing environments," *J. Comput. Inf. Syst.*, vol. 8, no. 4, pp. 1641–1654, Feb. 2012.

[4] Y. Zhang, D. Niyato, and P. Wang, "An auction mechanism for resource allocation in mobile cloud computing systems," in *Proc. 8th Int. Conf. Wireless Algorithms Syst. Appl.*, 2013, pp. 76–87.

[5] S. Zaman and D. Grosu, "A combinatorial auction-based mechanism for dynamic VM provisioning and allocation in clouds," *IEEE Trans. Cloud Comput.*, vol. 1, no. 2, pp. 129–141, Jul.–Dec. 2013.

[6] Y. Chen, Y. Wu, B. Wang, and K. J. R. Liu, "Spectrum auction games for multimedia streaming over cognitive radio networks," *IEEE Trans. Commun.*, vol. 58, no. 8, pp. 2381–2390, Aug. 2010.

[7] X. Wang, Z. Li, P. Xu, Y. Xu, X. Gao, and H. Chen, "Spectrum sharing in cognitive radio networks—An auction-based approach," *IEEE Trans. Syst. Man Cybern. Part B: Cybern.*, vol. 40, no. 3, pp. 587–596, Jun. 2010.

[8] A. N. Khan, M. M. Kiah, S. U. Khan, and S. A. Madani, "Towards secure mobile cloud computing: A survey," *Future Generation Comput. Syst.*, vol. 29, no. 5, pp. 1278–1299, Jul. 2013.

[9] K. Zhu, D. Niyato, and P. Wang, "Dynamic bandwidth allocation under uncertainty in cognitive radio networks," in *Proc. IEEE Global Telecommun. Conf.*, 2011, pp. 1–5.

[10] K. Kaur, A. Dua, A. Jindal, N. Kumar, M. Singh, and A. Vinel, "A novel resource reservation scheme for mobile PHEVs in V2G environment using game theoretical approach," *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5653–5666, Dec. 2015.

[11] Y. Zhang, C. Lee, D. Niyato, and P. Wang, "Auction approaches for resource allocation in wireless systems: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1020–1041, Jul.–Sep. 2013.

[12] J.-W. Lin, C.-H. Chen, and J. M. Chang, "QoS-aware data replication for data intensive applications in cloud computing systems," *IEEE Trans. Cloud Comput.*, vol. 1, no. 1, pp. 101–115, Nov. 2013.

[13] P. Bonacquisto, G. D. Modica, G. Petralia, and O. Tomarchio, "A procurement auction market to trade residual cloud computing capacity," *IEEE Trans. Cloud Comput.*, vol. 3, no. 3, pp. 345–357, Jul.–Sep. 2015.

[14] H. Viswanathan, E. K. Lee, I. Rodero, and D. Pompili, "Uncertainty-aware autonomic resource provisioning for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 8, pp. 2363–2372, Aug. 2015.

[15] K. Gai, M. Qiu, H. Zhao, L. Tao, and Z. Zong, "Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing," *J. Netw. Comput. Appl.*, vol. 59, pp. 46–54, Jan. 2016.

[16] L. Chen, B. Wang, X. Chen, X. Zhang, and D. Yang, "Utility-based resource allocation for mixed traffic in wireless networks," in *Proc. IEEE INFOCOM Int. Workshop Future Media Netw. IP-Based TV*, 2011, pp. 91–96.

[17] P. Chaikijwatana and T. Tachibana, "VCG auction-based bandwidth allocation with network coding in wireless networks," in *Proc. 10th WSEAS Int. Conf. Appl. Comput. Appl. Comput. Sci.*, 2011, pp. 104–109.

[18] D. Niyato, Z. Han, W. Saad, and A. Hjorungnes, "A controlled coalitional game for wireless connection sharing and bandwidth allocation in mobile social networks," in *Proc. IEEE Global Telecommun. Conf.*, 2010, pp. 1–5.

[19] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *J. Netw. Comput. Appl.*, vol. 34, no. 1, pp. 1–11, Jan. 2011.

[20] P. Kulkarni and R. Khanai, "Addressing mobile cloud computing security issues: A survey," in *Proc. IEEE Int. Conf. Commun. Signal Process.*, Apr. 2015, pp. 1463–1467.

[21] N. Kumar, S. Zeadally, N. Chilamkurti, and A. Vinel, "Performance analysis of Bayesian coalition game-based energy-aware virtual machine migration in vehicular mobile cloud," *IEEE Netw.*, vol. 29, no. 2, pp. 62–69, Mar./Apr. 2015.

[22] X. Wang, X. Wang, C. li Wang, K. Li, and M. Huang, "Resource allocation in cloud environment: A model based on double multi-attribute auction mechanism," in *Proc. IEEE Int. Conf. Cloud Comput. Technol. Sci.*, 2014, pp. 599–604.

[23] A. Amamou, M. Bourguiba, K. Haddadou, and G. Pujolle, "A dynamic bandwidth allocator for virtual machines in a cloud environment," in *Proc. IEEE Consum. Commun. Netw. Conf.*, Jan. 2012, pp. 99–104.

[24] C. Papagianni, A. Leivadeas, S. Papavassiliou, V. Maglaris, C. C. Pastor, and A. Monje, "On the optimal allocation of virtual resources in cloud computing networks," *IEEE Trans. Comput.*, vol. 62, no. 6, pp. 1060–1071, Jun. 2013.

[25] S. Das, S. Misra, M. Khatua, and J. J. P. C. Rodrigues, "Mapping of sensor nodes with servers in a mobile health-cloud environment," in *Proc. IEEE 15th Int. Conf. e-Health Netw. Appl. Serv.*, Oct. 2013, pp. 481–485.

[26] S. Das, M. Khatua, S. Misra, and M. S. Obaidat, "Quality-assured secured load sharing in mobile cloud networking environment," *IEEE Trans. Cloud Comput.*, vol. PP, no. 99, pp. 1–1, Jul. 2015, Doi: 10.1109/TCC.2015.2457416.

[27] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," *Econometrica*, vol. 33, pp. 520–534, Jul. 1965.

[28] S. Boyd and L. Vandenberghe, *Convex Optimization*, vol. 7. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[29] A. J. Goldsmith and S.-G. Chua, "Variable-rate variable-power MQAM for fading channels," *IEEE Trans. Commun.*, vol. 45, no. 10, pp. 1218–1230, Oct. 1997.

Snigdha Das received the MS degree (by research) from the School of Information Technology, Indian Institute of Technology Kharagpur, India, in 2015, and the BTech degree in information technology from West Bengal University of Technology, India, in 2010. She worked in two sponsored research projects with IIT Kharagpur and ISI Kolkata, India, respectively. She also worked as an associate software engineer in CSC for 2 years. Her current research interests include mobile computing and cloud computing.

Manas Khatua received the BTech degree in computer science and engineering from the University of Kalyani, India, in 2003, and the MTech degree in information technology from Bengal Engineering and Science University, India, in 2007. He received the PhD degree from the School of Information Technology, Indian Institute of Technology Kharagpur, India, in 2016. Presently, he is working as a postdoctoral research fellow with Singapore University of Technology and Design, Singapore. Prior to this he was associated with Tata Consultancy Services (India) for two and half years. He was a lecturer with Bankura Unnayani Institute of Engineering, India, for more than two years. His research interests include performance evaluation, wireless LANs, sensor networks, mobile cloud computing, and network security. Presently, he focuses upon machine learning and signal processing based approaches for network security. He is a member of the IEEE.

Sudip Misra received the PhD degree in computer science from Carleton University, Ottawa, Canada. He is an associate professor in the Department of Computer Science & Engineering, Indian Institute of Technology Kharagpur. Prior to this he was associated with Cornell University, Ithaca, New York, Yale University, New Haven, Connecticut, Nortel Networks, Canada, and the Government of Ontario, Canada. He has won eight research paper awards in different conferences. He was awarded the IEEE ComSoc Asia Pacific Outstanding Young Researcher Award at IEEE GLOBECOM 2012. He was also awarded the Canadian Government's prestigious NSERC Post Doctoral Fellowship and the Humboldt Research Fellowship in Germany. He is a senior member of the IEEE.