A Green Mechanism Design Approach to Automate Resource Procurement in Cloud

Doshi Chintan Ketankumar*, Gaurav Verma and K. Chandrasekaran
Department of Computer Science and Engineering, National Institute of Technology, Surathkal, Karnataka, Srinivasnagar; Dakshina Kannada, Mangalore 575 025, India

Abstract

Cloud computing paradigm is emerging as the solution to all the infrastructure setup problems of IT industry. But the thriving demand of cloud infrastructure has increased the energy consumption of the data centers drastically. As the energy consumption of the data center rises, it leads us to high carbon emissions which are dangerous for the environment. In this paper, we propose a green cloud broker for resource procurement problem by considering the metrics of energy efficiency and environmental friendly operations of the cloud service provider. We use mechanism design methods to decide the allocation and payment for the submitted job dynamically. We perform experiments and show the results of comparisons of energy consumption and emission of greenhouse gases between the allocation decided by the proposed green cloud broker and a without taking the green metric into consideration.

© 2015 The Authors. Published by Elsevier B.V.

Keywords: Cloud computing; Green cloud broker; Resource procurement; Mechanism design.

1. Introduction

Cloud computing is becoming an increasingly essential part of the business operations in the twenty-first century. The high scalability, ease of use, low setup cost, easy and speedy access to massive infrastructure, less total power consumption are the advantages of cloud computing which has helped in changing the way IT businesses are started and handled. With the growth of high speed networking facilities over the last decade, the rise in cloud usage is astonishing. According to the International Data Corporation (IDC) report the global IT cloud services spending has increased from $16 billion in 2008 to $42 billion in 2012, representing an annual growth of 27%. The enthralling growth rate has attracted the attention of almost all major IT and hardware solutions providers all around the world. Companies like Salesforce.com, Amazon, Google, Microsoft, SAP, IBM, Verizon have dedicated major parts of their capital for the establishment, development and research in cloud computing. Amazon AWS (Amazon web services) cloud platform reports 90% growth and as estimated by the financial analysis firm Pacific Crest, the total revenue of AWS will double every two years and it will soon reach the target of 10% of the total revenue of Amazon1.

However, the expansion of cloud is happening at a time when global warming, climate change and energy efficient infrastructure is of paramount concern. With the increasing usage of cloud comes an increasing demand of energy.
To deliver the efficient, agile and scalable cloud services, the data centers and server farms consume incredible amount of energy. The CO2 emissions of by server farms and data center in 2007 were 116 Metric Ton which is 14% of global information and communication technologies (ICT) CO2 emissions. The climate group and the global e-sustainability initiative (GeSI) reported in that this consumption will increase to 257 Metric Ton which will be 18% of the global ICT CO2 emissions by 2020. They further predict that the global carbon footprint of the main components of cloud-based computing – data centers and the telecommunications network – would see their emissions grow, on average, 7% and 5% respectively each year between 2002 and 2020. Underlying this analysis is the number of data centre servers growing on average 9% each year during this period.

Hence strict attention to research in development of energy efficient cloud computing solutions is required. The current research work suggests that companies have paid serious attention to optimizing the data centers total usage. The power management can be done at both software and hardware level. A. Beloglazov et al. present a taxonomy of all the approaches to energy efficiency at the system level. But one has to keep in mind that when environmental efficiency and reduction in carbon footprint is concerned, not only technical optimizations are enough. As these efficiencies can only slow down the growth of the emissions. The data centers can reduce their carbon footprints and achieve necessary reductions by practicing green approach, i.e. they can shift for renewable and non conventional sources of energy such as solar, wind or water energy. But currently no such serious efforts have been noticed by any company. Greenpeace international present the statistics of the percentage of coal energy usage by the data centers and they are suggesting that most of the companies are not serious about the source of the energy. According to that the coal intensities of Apple 54.5%, Facebook 53.2%, HP 49.4% and IBM is 51.6%. Only a few companies have taken steps to steer their infrastructure investments toward cleaner energy, but the sector as a whole remains focused on rapid growth. The replacement of dirty sources of electricity with clean renewable ones is still the essential missing link in the sustainability efforts of the sector.

In our work we analyze the various metrics available to measure the efficiency and environmental awareness of a cloud service provider and we create a composite metric to aggregate all the metrics. Then we consider one user multiple service provider allocation problem. We create a cloud broker application which takes all the QoS specifications, costs and the green metrics for each service provider. As the user submits a request, the cloud broker assigns one of the available service providers to the user. The allocation function takes into account all the three parameters, namely the QoS specifications, the cost of the cloud service provider and the green metrics. As the cost is the private information of the cloud service provider, so they may tend to become strategic and provide the false information to the broker. Hence to ensure truthfulness we use mechanism design concepts and provide a strategy-proof mechanism where reporting the true value becomes the most optimal strategy for the cloud service provider. We simulate this mechanism in a randomly generated scenario and provide the results of the allocation values. We compare the results with the allocation without taking the suggested green metric into picture and show how such green cloud broker applications can contribute in the green initiative.

The rest of the paper is divided into 6 sections. In section 2 we discuss all the related literature with our work. We discuss all the current research work in the direction of green cloud computing and all the game theory and mechanism design applications for resource procurement in cloud computing. In section 3 we formulate the problem which we are solving. We show the method to compute an aggregate green metric and QoS metric. We discuss what would be the optimal approach if no agent were strategic. We provide logical explanation about the problems relevance to mechanism design and all the properties we desire to achieve in this problem. In section 4 we explain the selection of the mechanism and we show that the mechanism holds all the desired properties. In section 5 we present the results of our experiment and prove how our results are better. And finally section 6 we present the conclusion and future work which can be done to help the situation.

2. Related Work

We divide our literature survey into two sub-sections. In the first sections we present the work related with green cloud computing and in the second section we present all the work consisting of game theory and mechanism design applications in cloud computing resource procurement problem.
2.1 Green cloud computing

There has been serious research work in the area of making the cloud computing greener\textsuperscript{7,8}. The organizations and government departmental programs taking active interest in this area are listed below:

- The Green Grid
- Green IT Promotion Council
- U.S. Environmental Protection Agency
- European Commission Joint Research center
- Ministry of Economy, Trade and Industry, Japan
- Climate Group and the Global e-Sustainability Initiative

The SMART 2020\textsuperscript{2} report was among the first reports to shed the light on the rising problem of the energy consumption and CO\textsubscript{2} emission by information and communication technologies industries. Kurp \textit{et al.}\textsuperscript{9} presented the risks of the pollution due to computing industry and presented the optimal view of a consumer. ODCA\textsuperscript{8} provides specific solutions and enterprise IT groups involved in planning, operations, and procurement to business decision makers and defining standards for standardization institutes. In\textsuperscript{7} report the authors define and provide measurement guidelines for the green cloud metrics such as Power Usage Effectiveness, Green Energy Coefficient (GEC), Energy Reuse Factor (ERF) and Carbon Usage Effectiveness (CUE). We use these metric and calculation information later in our paper as the basis for composite metric calculation. The report U.S. EPA\textsuperscript{10} provides the data about trends in growth and energy use associated with servers and data centers in U.S. They discuss various methods of energy and cost savings methods using energy efficiency. They give the recommendations on incentives and voluntary programs. U.S. energy information administration\textsuperscript{11} provides the instructions on voluntary reporting of greenhouse gases form EIA 1605. S. Garg \textit{et al.}\textsuperscript{3} explain how all components of a cloud computing environment contribut to energy consumption and the solution for future research directions to enable green cloud computing. S. Garg \textit{et al.}\textsuperscript{12} provide a user oriented cloud architectural framework called Carbon Aware Green Cloud Architecture which addresses the environmental problem with overall usage of cloud computing resources. A. Belogazov \textit{et al.}\textsuperscript{4} will be a good reference to look at if one wants to explore more in optimizing energy usage in cloud.

2.2 Game theory and mechanism design applications for resource procurement in cloud computing

The issue of multi service provider and one user scenario have been handled using game theory and mechanism design in various works. Z. Kong \textit{et al.}\textsuperscript{13} utilize mechanism design to allocate resources among selfish virtual machines in a non-cooperative cloud computing environment. They apply stochastic approximation as the procurement of the parameters needed to define the allocation function may not be noise-free. In our problem we consider the allocation among various users and how user completes the job is transparent to us. D. Niyato \textit{et al.}\textsuperscript{15} present the idea of resource and revenue sharing with a coalition formation in cloud service providers. This idea is related to the cloud broker application but however in our problem we do not let the cloud service provider share the submitted job as the job is an indivisible entity in our scenario. They do not consider any green approach and the division and allocation are based on the valuation only. They suggest a stochastic linear programming game to find the allocation and payment function. The approach by A. Prasad\textsuperscript{16} is similar to our approach in some ways. They propose three mechanisms named C-DSIC, C-BIC and C-OPT which are the application of VCG, d-AGVA and optimal mechanisms. But they neither consider any green parameters nor incentivize environmental friendly cloud service provider in their literature. Hence we believe our proposed idea is unique as the amalgamation of green cloud computing and mechanism design is not proposed by anyone to the best of our knowledge. For more applications of game theory and mechanism design concepts in resource procurement problems\textsuperscript{17,18} are good survey papers to refer.

3. Problem Formulation

3.1 Problem definition

We base our model on the mechanism design models explained by Y. Narahari \textit{et al.}\textsuperscript{6} We assume that all the cloud service providers participating in the process are rational and they aim to maximize their utility. Their utility in this
case is the revenue they receive from the completion of the job. Here we have \( n \) cloud service providers participating in the mechanism design setting. The cloud user submits the job to the cloud broker with the QoS specification and cost they are ready to pay for the job. The cloud broker is a automated software which has all the information about the cloud service providers\(^{19}\). The information consists of the QoS specification, minimum cost they would charge for the job and the values of all the green metric we take into consideration which are Power Usage Effectiveness (PUE), Green Energy Coefficient (GEC), Energy Reuse Factor (ERF) and Carbon Usage Effectiveness (CUE). The calculation of the metric is explained in\(^7\) with all the details. Hence we delegate the task of providing the broker the value of the metrics to the cloud service provider. Now we can assume safely that the values of this metrics cannot be reported falsely because if it is, then it becomes a legal issue for the cloud service providers and they can charged heavily for that. The cloud broker here is a software application which takes all the values described above as inputs. It becomes open to accept the jobs from cloud users. Once it receives the job, it computes the allocation and payment by the mechanism design principles and passes the job to the selected service provider. As the service provider finishes the job, the broker collects the payment from the user and transfers it to the service provider and simultaneously collects the results from the cloud service provider and passes it on to the cloud user. Hence the interaction is always mediated through the broker. It is completely transparent to the cloud user how the execution takes place.

3.2 Calculation of a composite green metric

The information about the green metrics that we are using for our composite metric calculation is given in Table 1. For the calculation of the composite metric we use the Simple Additive Weighting (SAW) method described in\(^20\). A holistic framework\(^7\) approach suggested allows the user to be aware about the effect of changes made to the data center specific to the various suggested metrics. For example, if one improvement at a data center that is targeted to improve a specific performance metric, it may have made the result of another metric look worse. Hence such a holistic framework helps the operator keep in mind the effects on all metrics simultaneously. But for our calculation we need to weigh each metric with respect to the other metrics and get one value of a composite metric which represents the environmental awareness of the cloud service provider as a whole. Now we assume that \( A_{i,k} \) is the value of the green metric \( k \) of the service provider \( i \). The values acquired during the input collection by the broker. We construct a matrix \( A = \{ A_{i,j}, 1 \leq i \leq n, 1 \leq j \leq k \} \). Each row of the matrix corresponds to the green metrics of the cloud service provider \( i \). Let \( B = \{ B_{i,j}, 1 \leq i \leq n, 1 \leq j \leq k \} \) be the matrix and \( B_{i,k} \) be the normalized value of the green parameter \( A_{i,k} \).

The SAW method can be divided into two individual stages of evaluation as:

3.2.1 Scaling

For all the metrics we have, we need to redirect all the parameters and scale them accordingly. Now the parameters can be either positive or negative. So we need two different equations to scale them. For the scaling of the negative parameters we use the following equation:

\[
B_{i,j} = \begin{cases} 
\frac{\text{max}(j) - A_{i,j}}{\text{max}(j) - \text{min}(j)} & \text{if } \text{max}(j) \neq \text{min}(j) \\
1 & \text{else}
\end{cases}
\]  

(1)

For the scaling of the positive parameters we use the following equation:

\[
B_{i,j} = \begin{cases} 
A_{i,j} - \text{min} j & \text{if } \text{max}(j) \neq \text{min}(j) \\
1 & \text{else}
\end{cases}
\]  

(2)

The scaled parameters will be in [0,1] and there will show positive behavior. By positive behavior we mean that more the value of the parameter, more aware the cloud service provider is about that factor. The information about the scaling of the parameters we use is given in Table 2.
Table 1. Behaviour, calculation information and weight for the metrics.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Behaviour</th>
<th>Minimum value</th>
<th>Maximum value</th>
<th>Scaling required</th>
<th>Equation</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUE</td>
<td>Negative</td>
<td>1</td>
<td>max (available inputs)</td>
<td>Yes</td>
<td>eqn 1</td>
<td>0.1</td>
</tr>
<tr>
<td>GEC</td>
<td>Positive</td>
<td>0</td>
<td>1</td>
<td>No</td>
<td>N.A.</td>
<td>0.5</td>
</tr>
<tr>
<td>ERF</td>
<td>Positive</td>
<td>0</td>
<td>1</td>
<td>No</td>
<td>N.A.</td>
<td>0.15</td>
</tr>
<tr>
<td>CUE</td>
<td>Negative</td>
<td>0</td>
<td>max (available inputs)</td>
<td>Yes</td>
<td>eqn 1</td>
<td>0.35</td>
</tr>
</tbody>
</table>

Table 2. Description of the metrics.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Description</th>
<th>Formulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUE</td>
<td>Power Usage Effectiveness</td>
<td>$PUE = \frac{\text{Total Data Center Source energy}}{\text{IT equipment energy}}$</td>
</tr>
<tr>
<td>GEC</td>
<td>Green Energy Coefficient</td>
<td>$GEC = \frac{\text{Green Energy Used by the Data Center}}{\text{Total Data Center Source Energy}}$</td>
</tr>
<tr>
<td>ERF</td>
<td>Energy Reuse Effectiveness</td>
<td>$ERF = \frac{\text{Reuse energy outside of the Data Center}}{\text{Total Data Center Source energy}}$</td>
</tr>
<tr>
<td>CUE</td>
<td>Carbon Usage Effectiveness</td>
<td>$CUE = \frac{\text{Total CO}_2\text{emissions caused by the Total data center energy}}{\text{IT equipment energy}}$</td>
</tr>
</tbody>
</table>

3.2.2 Weighting

In this stage the composite metrics are calculated for each cloud service provider. They are computed using the following formula.

$$s_i = \sum_{j=1}^{k} B_{i,j} \ast w_j \text{ where } w_j \in [0, 1] \text{ and } \sum_{j=1}^{k} w_j = 1 \quad (3)$$

The values of the weights are taken from Table 2. Hence the all green metric are aggregated to a number. Now we need to scale the number by a scaling function. We use the following function to scale values.

$$G_i = 1 + s_i \quad (4)$$

Here the values of the weights are taken by observation of the importance of the factor. A different scheme for weights can be taken to take the result in a different direction. Similarly the scaling function can be changed to change the strictness of the mechanism. The effect of the change in scaling function on the strictness on mechanism is explained in the next section.

3.3 Relevance of mechanism design to this problem

The problem of the allocation of the job as described above is essentially a decision or optimization problem with incomplete information. Now our scenario has three specific characteristics25, 26.

1. A set of cloud service providers who interact in a strategic way. They want to maximize their revenue and all of them are assumed to be rational. Now as job is indivisible entity, only one player will be allocated the job and only that player will get a chance to earn revenue. Hence the objectives of the players are clearly conflicting.

2. Each service provider holds private information which no one has any way to know and only the player knows this deterministically. This information is the cost which the service provider will incur by completing the job assigned to it. Thus, this is the situation of incomplete and decentralized information. There is some information such as QoS value, green metric which is a common knowledge among the player.

3. Each service provider has a choice set of values from which it can select one value which will maximize its generated revenue. This availability of choices results in strategic actions by the player which may result in false reporting of the cost.

Now we need the system-wide solution to satisfy some desirable properties. The best way to achieve is induce a game amongst the service providers and let them decide their strategy. As everyone will know the allocation and payment
Table 3. Symbol table.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Number of service providers</td>
</tr>
<tr>
<td>N</td>
<td>Set of the service providers</td>
</tr>
<tr>
<td>X</td>
<td>Set of outcome</td>
</tr>
<tr>
<td>$q_i$</td>
<td>QoS composite of service provider $i$</td>
</tr>
<tr>
<td>$\theta_i$</td>
<td>Preference/type of the service provider $i$</td>
</tr>
<tr>
<td>$c_i$</td>
<td>Cost of the service provider $i$</td>
</tr>
<tr>
<td>$c_i'$</td>
<td>Reported cost of the service provider $i$</td>
</tr>
<tr>
<td>$G_i$</td>
<td>Green composite metric of service provider $i$</td>
</tr>
<tr>
<td>$\Theta_i$</td>
<td>Set of types of service provider $i$</td>
</tr>
<tr>
<td>$t_i$</td>
<td>Payment for service provider $i$</td>
</tr>
<tr>
<td>$k_i$</td>
<td>Allocation for service provider $i$</td>
</tr>
<tr>
<td>$v_i$</td>
<td>Valuation for service provider $i$</td>
</tr>
<tr>
<td>$b$</td>
<td>Bid Vector</td>
</tr>
</tbody>
</table>

function, they will optimize their strategies (reported cost) to achieve the maximum possible revenue. So if the broker decides the allocation and the payment functions in such way that the optimized strategies of each player satisfy all the properties the broker wants the system to have. Such a reverse engineering of the game theory is done by mechanism design. Mechanism design provides the setting where a social planner mechanism designer (broker) faces the problem of the aggregation of the reported preferences-types of multiple players (service providers) into a collective decision when the real preferences-types are unknown. The social choice function describes the collective decision. It takes all the reported types as input and gives the collective decision-outcome as output. Each player (service provider) will have predefined utility function which maps the actual type of the player and the outcome to a real number which suggests the happiness of the player\textsuperscript{6, 27, 28}.

3.4 Desired properties

3.4.1 Incentive compatibility

If no player receives any more utility by reporting false preference than by reporting the true preference, such mechanism is called an incentive compatible mechanism. In such mechanism the best response for the player is to report the true value of their preference. There exists a Nash equilibrium or weakly dominant strategy equilibrium in the games generated by such mechanisms. If there is Nash equilibrium, then a player cannot do any better by reporting false preference than the true preference; given that all other player report their true preference. Such mechanisms are called Bayesian-Nash incentive compatible mechanism (BIC). If there is weakly dominant strategy equilibrium, then a player cannot do any better by reporting false preference than the true preference in any circumstances. Such mechanisms are called Dominant Strategy Incentive compatible mechanisms (DSIC). This is a strategy-proof situation where the player has no other option then reporting the truth. Such mechanisms are also called truth-revealing mechanisms. This is an extremely desirable property in our scenario.

3.4.2 Pareto-efficiency or ex-post efficiency

If the outcome produced by the social choice function is such that, there is no other outcome possible, which will make some players gain more utility and the remaining gets the same utility as they were getting with the produced outcome. Hence the outcome is the best for all players with their current preferences.

3.4.3 Allocative efficiency

A mechanism is called allocatively efficient if the allocation of the resources is done to the player who values it the most. Such property is extremely desirable in our case.
3.4.4 Budget balance

When the sum of all monetary transfer is equal to or less than zero, then such mechanism is called budget balanced. That suggests that there is no deficit in the mechanism and the mechanism will run without any monetary support from outside. If the sum is equal to zero then it corresponds to strong budget balance which implies no surplus and no deficit. Such mechanisms are self sustainable. If the sum is less than zero then there is a surplus of money in the mechanism which can be consumed by the broker or market maker.

3.4.5 Individual Rationality

A mechanism is individually rational if the participating player does not receive more utility by not participating in the mechanism design. If the mechanism is not individually rational then there is a possibility that the agents for which it is not rational; they can withdraw from the mechanism.

4. Mechanism Design Solution

4.1 Selection of the mechanism

As in the literature for mechanism design there are various mechanisms available such Vickry-Clarke-Groves mechanisms, d-AGV mechanisms, Mysersons optimal mechanisms. All mechanisms provide various properties which are discussed above. Now for this scenario we require a mechanism to be DSIC and AE. We do not want to give up on these two properties. Besides these, the property of strong budget balance is desirable. But it is impossible to design any mechanism such that it satisfies SBB in addition to AE and DSIC. And as we do not want to sacrifice on AE and DSIC we cannot include SBB in our mechanism. Among VCG mechanism Groves mechanism is the most general one; however for the current scenario the Clarke mechanism suffices. Clarke mechanism satisfies both the properties, DSIC and AE, which we require. Hence we use Clarke mechanism as the first approach to solve the problem.

4.2 Clarke-pivotal mechanism

In our problem setting we have $n$ cloud service providers which are represented by $N = \{1, 2, \ldots, n\}$. Now in this procurement procedure each service provider submits a bid consisting of the green composite metric, the QoS composite metric as calculated in the section above and the total cost as an ordered pair $(G_i, q_i, c_i)$. Now the cloud service provider can lie about it in order to get more revenue. Hence cost can be from the interval $[c, e]$. Now the cloud service provider has to report the true value of QoS composite index and Green composite index because false reporting of these values can lend the service provider into legal troubles. Hence the cost of the cloud service provider becomes its type-preference value. Hence, $\theta_i = c_i$. Now the following are the allocation and valuation rules for the mechanism.

- Allocation rule:

$$B_{i,j} = \begin{cases} 1 & \text{if } \frac{G_i * q_i}{c_i} = \max \left( \frac{G_1 * q_1}{c_1}, \ldots, \frac{G_n * q_n}{c_n} \right) \\ 0 & \text{else} \end{cases} \quad (5)$$

- Valuation rule:

$$v_i(k(\theta), \theta_i) = -k(\theta) * \theta_i \quad (6)$$

Now it is clear from the valuation and allocation function that if the scale of the green composite metric is changed, it changes the valuation function which decides the allocation. Hence the scale of the green composite metric affects the strictness of the mechanism. We defined the Allocative efficiency of a mechanism above. The mathematical representation of the definition is as below:

$$k(\theta) \in \arg \max_{k \in K} \sum_{i=1}^{n} v_i(k, \theta_i) \quad \forall \theta \in \Theta$$

(7)
Algorithm 1. Algorithm to Compute the Winner and the Payment

Now the payment rule for the Clarke-Pivotal mechanism is as follows\(^{29}\):

\[
t_i(\theta) = \sum_{j \neq i} v_j(k^{*}(\theta), \theta_j) - \sum_{j \neq i} v_j(k^{*}_{-i}(\theta_{-i}, \theta_j))
\]

(8)

4.3 Analysis of the model

According to the properties of the Clarke mechanism as mentioned earlier, the proposed solution achieves the desirable properties. In the Clarke-Pivotal mechanism the payment to the winner is the difference between the sum of valuation of the players other than the winner in the presence and the absence of the player. As it is clearly visible by the eqn 9 that all other players other than the winner of the allocation receive zero revenue. Hence the mechanism is individually rational. Our allocation function is trivially AE. And it is the property of the Clarke mechanism that it achieves DSIC. Thus the allocation happens to the cloud service provider whose valuation function. Hence, we can observe that the proposed solution perfectly fits the scenario theoretically.

5. Experimental Analysis

For the evaluation of the mechanism we suggested above we do not use standard CloudSim or any such toolkit for various reasons. We use a python based implementation of the Clarke-Pivotal mechanism with the allocation, valuation and payment function described in the section above. The highlights of our simulation approach are as follows:

1. For the resource prices we used the lognormal distribution in the interval of \([6000,8000]\) as described by A. Prasad et al.\(^{16}\) We limit the interval to observe the allocations with nearer price values.
2. As we wanted to evaluate the mechanism and do not perform the calculation at cloud service provider level we took QoS parameters as normal distribution with mean and variance value 5 to facilitate rating on the scale of 10.
3. The values of the green composite index before applying scaling function are taken as normal distribution with mean and variance 0.5 and then the scaling function is applied on them.
4. As simulation software for cloud applications such as CloudSim or Eucalyptus do not provide any support for auction protocols or price generations. Hence we had to use the python based simulation to evaluate the equations we presented in the section above.

Table 4 presents the data related to our simulation. The graphs in Fig. 1 show the comparison between the cost values and the green composite metric values in both the cases. Here the important result to notice in Fig. 1 is that the when the green metric is considered in the mechanism, the value of the green metric of the winner is always greater...
Table 4. Results.

<table>
<thead>
<tr>
<th>No of service provider</th>
<th>Without considering the G</th>
<th>With G included</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Received payment</td>
<td>Received payment</td>
</tr>
<tr>
<td></td>
<td>G for the winner</td>
<td>G for the winner</td>
</tr>
<tr>
<td>10</td>
<td>6847.64148956</td>
<td>6847.64148956</td>
</tr>
<tr>
<td>20</td>
<td>7070.31178585</td>
<td>6029.57367142</td>
</tr>
<tr>
<td>30</td>
<td>6932.86812016</td>
<td>6206.14430889</td>
</tr>
<tr>
<td>40</td>
<td>6510.62811092</td>
<td>6931.70474788</td>
</tr>
<tr>
<td>50</td>
<td>6432.46891114</td>
<td>7101.62660025</td>
</tr>
<tr>
<td>60</td>
<td>6700.56822365</td>
<td>6608.89015437</td>
</tr>
<tr>
<td>70</td>
<td>6341.89304004</td>
<td>6551.90621447</td>
</tr>
<tr>
<td>80</td>
<td>6845.06440855</td>
<td>6129.86051091</td>
</tr>
<tr>
<td>90</td>
<td>6788.3181607</td>
<td>6938.23422267</td>
</tr>
<tr>
<td>100</td>
<td>6276.2097097</td>
<td>6674.13065512</td>
</tr>
<tr>
<td>200</td>
<td>6227.42511039</td>
<td>7280.97873435</td>
</tr>
<tr>
<td>300</td>
<td>6176.86925999</td>
<td>6176.86925999</td>
</tr>
<tr>
<td>400</td>
<td>6141.75621733</td>
<td>6307.36845192</td>
</tr>
<tr>
<td>500</td>
<td>6190.64958101</td>
<td>6351.6756224</td>
</tr>
<tr>
<td>1000</td>
<td>6079.18866319</td>
<td>6159.3965151</td>
</tr>
</tbody>
</table>

Fig. 1. Simulation results.

than the value of green metric of the winner in the other case. Hence the service provider with more energy efficient cloud is given priority over the less efficient cloud. Figure 1 shows that there is no trend in the payment comparison to the winner which is a good result because there are some cases where the allocation happens to a cheaper and greener cloud service provider. But one needs to keep in mind the allocation function while comparing the result. If such case occurs than it will be a cloud service provider with less QoS value. Thus, our simulation prove that our mechanism is successful in choosing the greener cloud service provider.

6. Conclusion and Future Work

We conclude that the mechanism proposed in this paper helps the cause of environmental awareness in information and communication technologies. The cloud broker application selects cloud service providers with higher green composite metric. This can result into user paying more money. But as the allocation function takes green composite metric and cost as inputs, the cost difference which the cloud user pays extra won’t be an extremely higher value. The mechanism proposed by us is able to automate the selection of the cloud service provider and it reduces a lot of time overhead for the cloud user.

We believe that there is still scope for improvement in the assignment of weights to the different metric to measure the efficiency and environmental awareness. The scaling functions can be optimized to achieve results which are environment centric.
References


www.greenpeace.org/international/en/publications/reports/make-it-green-Cloudcomputing/


[8] Open Data Center Alliance Usage Model: Carbon Footprint and Energy Efficiency Rev 2.0


www.eia.doc.gov/oiaf1605/pdf/Appendix%20F–r071023.pdf


