The QoGS Method Application for Selection of Computing Resources in Intercloud

R. Butkiene[1], G. Vilutis[2], I. Lagzdinyte-Budnike[2], D. Sandonavicius[2], K. Paulikas[2]
[1]*Department of Information Systems, Kaunas University of Technology,*
Studentu St. 50–309, LT-51368 Kaunas, Lithuania
[2]*Department of Computer Network, Kaunas University of Technology,*
Studentu St. 50–416, LT-51368 Kaunas, Lithuania
gytis.vilutis@ktu.lt

***Abstract*—In this paper the selection of the right Cloud in Intercloud of computing services for fulfilment of the user task is analyzed. The main problem approached is how to select the right Cloud in Intercloud for the user task when there are no available or relevant resources in the private and public Cloud. In this case the Cloud, which gives the calculation results to the user in the shortest time including waiting time in queue, should be selected. For selection of the right Cloud the QoGS method has been chosen. The principle schema of QoGS method application in Intercloud has been presented. However, QoGS method works properly if the right set of weighted coefficients (SoWC) is selected. Existing methods are not suitable for selection of weighted coefficients because they require sending a lot of test tasks to Intercloud. Therefore, a new algorithm for selection of the best SoWC is presented. Results of the experiment show that it allows minimizing the workload of Intercloud by test tasks significantly.**

***Index Terms*—Cloud, load flow control, computer networks.**

I. Introduction

A Cloud computing technology becomes more and more popular. Therefore, users face the problem when they are selecting the Cloud for the task fulfilment. The time of task fulfilment in the Cloud is the main indicator of the computing Cloud quality and the main criterion in the Cloud selection. The time of the task fulfilment depends on many parameters of the Cloud: an amount of resources allocated for tasks, a data transmit rate, a size, complexity, and a number of the tasks in queue, etc. However, these parameters are not usually presented for a user. Even if the parameters are presented, the Cloud selection remains difficult if for different Clouds different sets of the parameters are presented. A unification of parameters describing the Cloud would decrease these difficulties. One way to unify the Cloud description is the use of ontology [1]. An application of ontology would allow to search and choose the right Cloud not only for the human-users but also for the software agents. Nevertheless, there are many parameters describing a Cloud. To assess all of them is time-consuming and requires the user's experience. If we want to automate a selection of Cloud, a method, which is able to assess all parameters of Cloud and choose the best one, is needed.

The unification of Cloud's parameters is not analyzed in this paper. The aim of this paper is to choose and apply the method for the right Cloud selection in Intercloud. Furthermore, this paper focuses on the Clouds providing services according SaaS (Software as a Service) model, for example graphics rendering services. The right Cloud selection is difficult when choosing from Clouds providing services according IaaS (Infrastructure as a Service) or PaaS (Platform as a Service) model. In these cases the preparation and adjustment of infrastructure, upload of image and so on can take considerable time.

II. Intercloud Structure

The method for selection of the right Cloud for the user's task fulfilment requires certain information about each computing Cloud; in other words, it requires the parameters about each Cloud. In this paper we refer to the Intercloud [2]–[6] (Fig. 1) where information about each Cloud in the Cloud network could be found, using services of the Intercloud Root. The Intercloud Root hosts the Cloud Computing Resource Catalogs for the Intercloud computing resources. The Intercloud Root and Intercloud Exchanges facilitate and mediate the initial Intercloud negotiating process among Clouds. Once the initial negotiating process is completed, each of these Cloud instances collaborates directly with each other.

The user accesses Private Cloud and receives its resources via Interface. Main function of this Interface is to describe the user's requirements for the computing resources formally. When the specification of the user requirements is received, the Cloud Broker looks up for the proper resources in the Private Cloud. If the resources satisfying the user's requirements are not found, the Cloud Broker accesses the Intercloud Root and looks up for the resources in the Public Cloud. The method for the resource search in the Intercloud is discussed in this paper.

III. Related Works

A method used by Cloud Broker (CB) for the resource selection plays an important role.

Manuscript received August 5, 2012; accepted April 19, 2013.

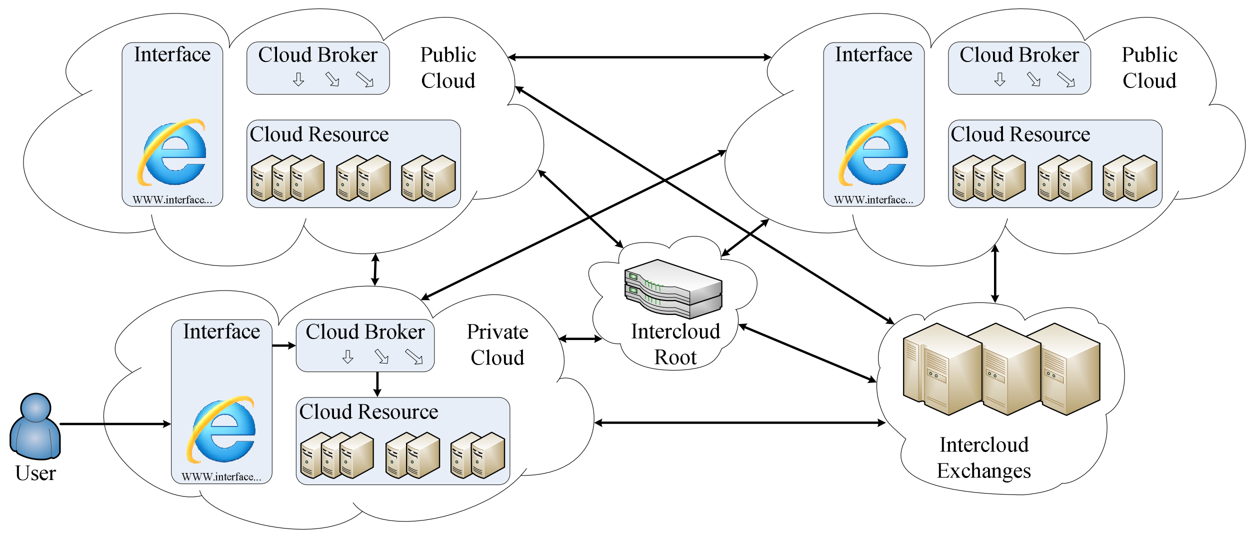

Fig. 1. The reference intercloud structure.

The role of the Cloud Broker is similar to the role of the Resource Broker, which is a common component of the computing Grid.

ERT method is mostly used resource selection method by Grid's Resource Broker [7]. However, many researchers state that the Resource Broker working on ERT method poorly divides tasks in the network [8], [9]. The reason is that ERT method evaluates only two parameters of the resource: central processor speed C and job queue length L. Other methods, such as GSQN [10], Q-QoSM [11], [12] and QoS-based [13], evaluate three parameters of the resource. Each of them uses different sets of the parameters: GSQN evaluates the resource availability, a total time to delivery, and a cost; Q-QoSM – a data transfer rate, latency, and a cost; and QoS-based –central processor speed C, a storage size, and a RAM memory size. Whereas the QoGS resource selection method [14], [15] evaluates all parameters of the resource, which are evaluated in the methods mentioned above, and more. A possibility to evaluate more parameters allows the user determining the right resource for his/her task fulfilment more precisely. An experiment shows that the QoGS method chooses the resources better than other methods do [15]. Moreover, the QoGS method forecasts the dynamic parameters of the resource, such as average tasks fulfilment time in queue. The methods mentioned above do not have such feature.

The selection of the right resource in the QoGS method is based on the evaluation of the quality of the Grid Cluster Q_{nk} by

$$Q_{nk} = \frac{\sum_{i=1}^{I}\left(\omega_{ki}^{D}\cdot\frac{V_{ni}^{D}}{V_{i}^{D\max}}\right)+\sum_{j=1}^{J}\left(\omega_{kj}^{M}\cdot\frac{V_{j}^{M\max}-V_{nj}^{M}}{V_{j}^{M\max}-V_{j}^{M\min}}\right)}{\sum_{i=1}^{I}\omega_{ki}^{D}+\sum_{j=1}^{J}\omega_{kj}^{M}}, \quad (1)$$

where n – a number of a clusters, ($n=\overline{1,N'}$); k – job index ($k=\overline{1,K}$); V_{ni}^{D} – i-th parameter value of n-th cluster, where D marks a parameter, and when this parameter value increases, qualitative estimate increases; V_{nj}^{M} – j-th parameter value of n-th cluster, where M marks a parameter, and when this parameter value increases, qualitative estimate decreases; $V_{i}^{D\max} = \max_{n=1,\dots,N} V_{ni}^{D}$ – the highest value of parameter i-th in Grid network. $V_{j}^{M\max}$ is the highest and $V_{j}^{M\min}$ is the lowest value of parameter j-th in Grid network. ω_{ki}^{D} and ω_{kj}^{M} – weights of i-th and j-th parameters, set by k-th job.

The QoGS method could be adapted for selection of the right resource in the Intercloud. In this case the formula (1) should be treated in this way: n should express a number of Clouds in the Intercloud, k - index of the task sent to Cloud, V_{ni}^{D} – i-th parameter value of n-th Cloud, V_{nj}^{M} – j-th parameter value of n-th Cloud, $V_{i}^{D\max}$ – the highest value of the parameter i-th in the Intercloud, $V_{j}^{M\max}$ and $V_{j}^{M\min}$ would be the highest and the lowest value of the parameter j-th in the Intercloud respectively, ω_{ki}^{D} and ω_{kj}^{M} – weights of i-th and j-th parameters, set by k-th task.

A good selection of resources in the QoGS method is ensured by the right selection of weighted coefficients of the parameters (WCoP). The weighted coefficients (WC) point out parameters of the resource, which influence the time of the task fulfilment in the Cloud most. It is important to note that the QoGS method is sensitive for the weighted coefficients the user has been inputted. If the weighted coefficients are selected improperly, the QoGS method can even worsen the time of the task fulfilment, comparing to other methods. Therefore, it is very important to determine the right set of weighted coefficients (SoWC).

Using the QoGS method, the determination of the set of the weighted coefficients is done by the user. However, for the inexperienced user such task is quite difficult [16]. The WCoP determination algorithm for the CB would be helpful. If the user does not set the best SoWC in the description of his/her task requirements, the CB sets it, considering the results of the WCoP determination algorithm.

The weighted coefficients are widely used in the clustering and classification tasks [17]–[19], neural networks [20]–[24] and other. The neural networks for determination of the weighted coefficients use self-learning methods. Such methods could be applied for recognition of the similar task that comes to the Cloud. Though, in the case of the QoGS method, not the task but the right resource for that task is selected. The parameters of the resource, in regard to a new task that came into the Cloud, differ (especially the number of tasks that are queued in the resource row). Therefore, self-learning methods of the neural networks are not applied.

For the solving optimization tasks the weighted coefficients are calculated, using Simplex method [25]–[27]. However, Simplex method cannot be used in the case of the QoGS method because it is impossible to create an equation system for selection of the best SoWC; there is a lack of information about the time of the task fulfilment. Therefore, a new solution for determination of the WCoP used in the QoGS method is needed.

IV. An Application of QoGS Method in the Intercloud

The QoGS method in the Intercloud should be applied in case when the relevant resource in the Private Cloud is not found. A principle schema of the Intercloud resource selection is depicted in Fig. 2. The user sends a task to the Cloud Broker and specifies the requirements for resources of Cloud (sets the preferences, weighted coefficients and constraints on parameters of Cloud) via Interface. The CB receives the parameters of all Intercloud resources from the Intercloud Root and filters the resources satisfying the user's requirements. If resources, satisfying the user's requirements, are found, the CB applies the QoGS method for selection of the most relevant Intercloud resource from filtered ones. When a relevant resource of the Intercloud is selected, the task is sent to that resource. After the task fulfilment a result is sent to the user via Interface.

When the CB applies the QoGS method for selection of the relevant resource, it takes the SoWC from WC storage. The SoWCs of all Intercloud resources are calculated by and stored in the Intercloud Root. The Intercloud Root recalculates the SoWCs of the Cloud resources if the structure of the Cloud has been changed. The Cloud Broker receives the SoWCs from the Intercloud Root and stores them in its own storage. The WC storage makes the normal work of the CB possible when the connection to the Intercloud Rood is lost.

V. An Algorithm for Determination of the WCoP of the Intercloud Resource

The purpose of the algorithm is to determine the best set of the weighted coefficients of the parameters of the Intercloud resources. As was mentioned above, the WCoP are the essential factor in the selection of the Cloud.

For the further consideration, the four parameters of the Cloud, which influence the successful and the fastest fulfilment of the task most, have been chosen: a number of the processors N, the average speed of the processors $\bar{C}$,. The parameters N and $\bar{C}$ define a capacity of the Cloud. Knowing these parameters, it is possible to predict how quickly the user's task will be fulfilled. Knowing parameters L and $\bar{W}$, it is possible to predict when the fulfilment of the user's task will start if the queue of the tasks exists. The parameter $\bar{W}$ is especially important when deciding which Cloud should be selected.

The easiest way to obtain the best SoWC is to generate all possible combinations of the weighted coefficients (SoWC) for the Cloud parameters in specified variation limits, to calculate the Q_{nk} (see formula (1)) with each SoWC for each Cloud of the Intercloud, to select the Cloud with the highest Qnk, to send the test tasks to the selected Cloud, and to observe fulfilment time of the test task. The test task loads the Cloud resources minimally. Its purpose is to inform the Cloud Broker that it got into the computing resource. The test task ends immediately when a message to the CB is sent. From the set of the received test task fulfilment times the shortest one is selected.

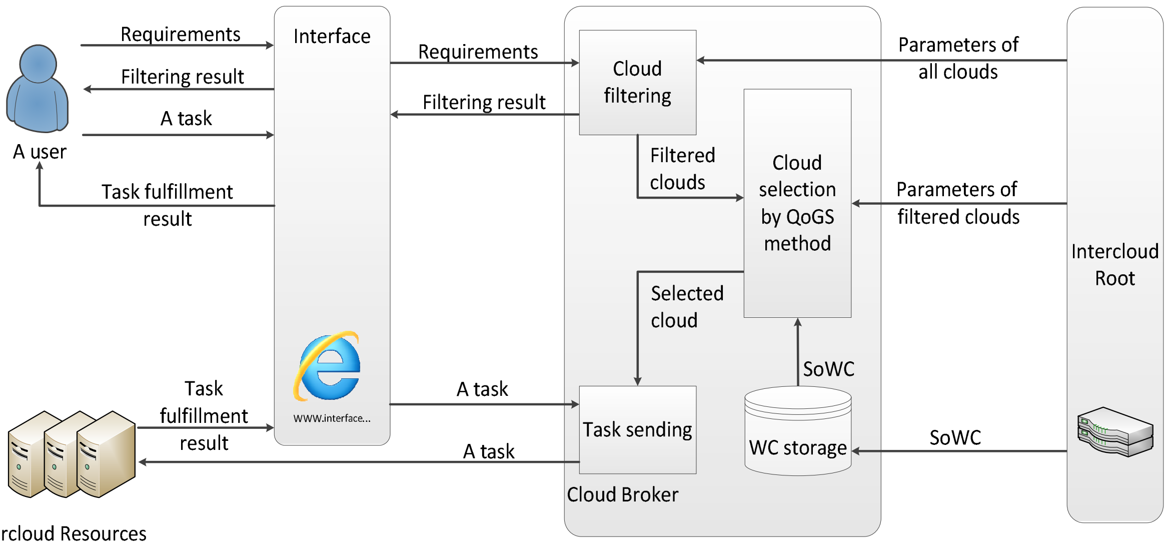

Fig. 2. A Principle schema of the cloud selection in the intercloud.

The SoWC, which pointed out the Cloud, which executed the test task fastest, is considered as the best SoWC. However, this way can be time-consuming.

A number of generated SoWCs grows when the number of parameters increases or the variation limits of the WC are expanded. For example, when the variation limits of the WC are set to [1..2], a variation step is 1, and the number of the parameters increases from 4 to 5, the number of the SoWC increases from 16 to 32. When the number of the parameters is 4 and the variation limits of the WC are expanded from [1..2] to [1..10], the number of the SoWC increases from 8 to 10000.

It is necessary to emphasize that different but reiterating SoWCs generated among all possible SoWC can be found. A reiterating SoWC is considered such set where all values of that set are multiples of the values of another SoWC. The example where the SoWC is reiterating is presented in the formula (2), where the SoWC defines four parameters of the Cloud

$$Q_{nk}^{SoWC1}\left|\begin{matrix}\omega_W = 1\\ \omega_L = 1\\ \omega_C = 2\\ \omega_N = 4\end{matrix}\right. = Q_{nk}^{SoWC2}\left|\begin{matrix}\omega_W = 2\\ \omega_L = 2\\ \omega_C = 4\\ \omega_N = 8\end{matrix}\right., \qquad (2)$$

where ω_W – WC for the average time of the tasks fulfilment $\overline{W}$; ω_L – WC for the number of the tasks in the queue (or queue length) L; ω_C – WC for the average processor speed of the Cloud $\overline{C}$; ω_N – WC for the number of the processors N.

Different but reiterating SoWCs calculate the quality Q of the Cloud equally and do not give any benefit in the selection of the relevant Cloud. Thus, the reiterating SoWC could be rejected.

The algorithm (Fig. 3) described in this section does not embody shortcomings mentioned above. In the beginning of the algorithm an initial SoWC must be set. If the algorithm is executed for the first time, the value 1 is set to all WC in the initial SoWC. The variation limits of the WCoP are set to [1..2]. Then all possible SoWC are generated. Each generated SoWC is tested by sending the test tasks to the Intercloud for fulfilment. When the data about the fulfilled test tasks is received, the best SoWC of the iteration is selected. The best SoWC is considered a set, which pointed the task to the resource where fulfilment time of the task was the shortest. The best SoWC selected is compared to the best SoWC selected in the previous iteration. If the best SoWC selected in the current iteration is equal to the best SoWC selected in the previous iteration multiplied by 2, then the best SoWC selected in the current iteration is divided by 2 and saved by the Intercloud Root. The algorithm completes. Otherwise, all values of the best SoWC selected in the current iteration are multiplied by 2 and for each parameter the variation limits of the weighted coefficients are changed in this way: the lower variation limit of the WC is obtained by decreasing the value of the WC by 1, and the upper variation limit of WC is obtained by increasing the value of the WC by 1. For example, if WC of certain parameter is 4, then the variation limits of this parameter will be [3..5]. When the new variation limits for the WCoP are set, the next iteration for selection of the best SoWC starts.

If the algorithm was executed once, then the best SoWC determined in the previous calculations is set as initial SoWC. The values of the initial SoWC are multiplied by 2 and the variation limits of the WCoP are set in the way described in the previous paragraph. Then the iteration for selection of the best SoWC starts. When the best SoWC are determined, it is recorded into the WC storage and used in the QoGS method. However, if the structure of the Cloud changes, it is necessary to recalculate the SoWC. The best SoWC received in the next iterations of the algorithm are more and more precise – it allows selecting more relevant Cloud.

However, the search of the most precise SoWC can take a lot of time.

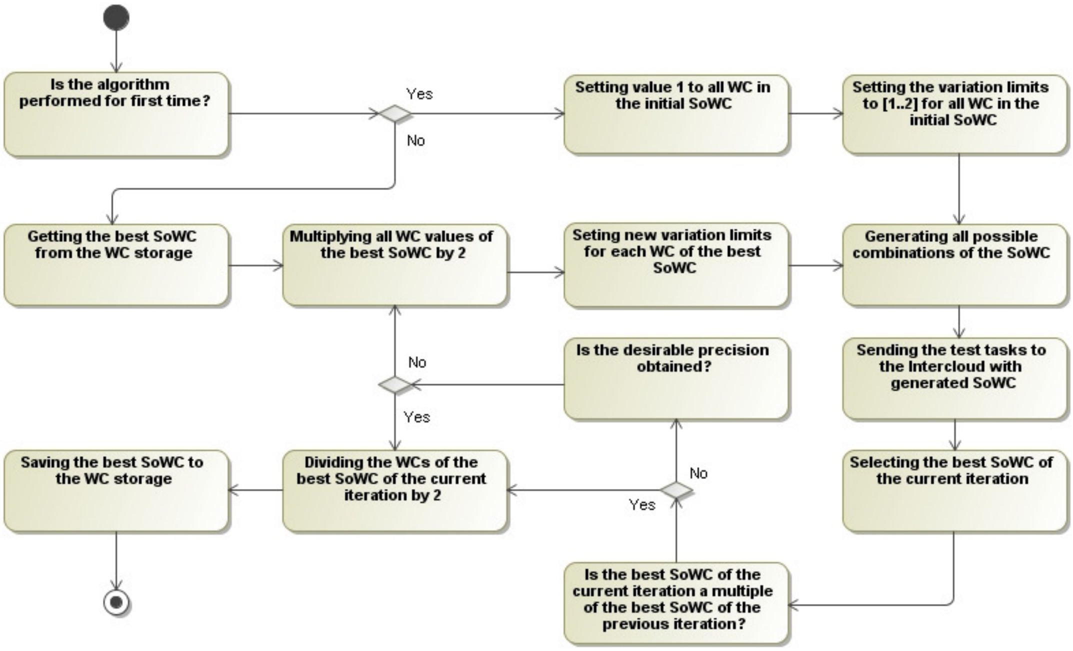
Is the algorithm performed for first time?
Yes
No
Setting value 1 to all WC in the initial SoWC
Setting the variation limits to [1..2] for all WC in the initial SoWC
Getting the best SoWC from the WC storage
Multiplying all WC values of the best SoWC by 2
Seting new variation limits for each WC of the best SoWC
Generating all possible combinations of the SoWC
No
Is the desirable precision obtained?
Sending the test tasks to the Intercloud with generated SoWC
Yes
Saving the best SoWC to the WC storage
Dividing the WCs of the best SoWC of the current iteration by 2
No
Yes
Selecting the best SoWC of the current iteration
Is the best SoWC of the current iteration a multiple of the best SoWC of the previous iteration?

Fig. 3. An Algorithm for determination of WCoP.

Therefore, a possibility to terminate the algorithm is provided. If the precision obtained (task fulfilment time) meets the user's requirements, he/she can terminate the algorithm.

VI. Experiment

The aim of the experiment was twofold:

1) To determine how many times the number of the test tasks sent to Intercloud is decreased when the proposed algorithm is used comparing to the number of the test tasks sent to the Intercloud when all possible combinations of the SoWC are generated;
2) To determine when the execution of the proposed algorithm should be stopped.

The simulation model was created to do an experiment. The model simulates work of Cloud Broker in the Intercloud. The Cloud Broker applies the QoGS method and the proposed algorithm for the WCoP determination. Static parameters of the Clouds are presented in Table I. These parameters were used in simulation when, according to the method described in this paper, the load balancing of the flow of simulated tasks was performed. The first column "No." identifies the Cloud. A number of processors *N* and average processors speed $\bar{C}$ influenced dynamic parameters (a number of the tasks in the queue *L*, and the average time of the task fulfilment $\bar{W}$) of the Clouds directly.

The structure of the Intercloud is similar to the structure of Grid network including services offered by them. But the structure of the Intercloud is underdevelopment now, so there is no possibility to acquire statistical data about operation of the Intercloud. Therefore the simulation model was validated comparing its operation with the work of BalticGrid network [28]. The simulation model comparing with BalticGrid network works with the average square error 0.7249.

During simulation the WCs $(\omega_W, \omega_L, \omega_C, \omega_N)$ were chosen for 4 parameters: $\bar{W}$, *L*, $\bar{C}$, *N* (see the description of parameters in the previous section). The parameters of the tasks generated by the stream were:

1) The required CPU speed – 1 GHz;
2) The required number of the processors – 10-19;
3) The duration of the task performance – 10-100 min on the 1GHz processor.

TABLE I. THE STATIC PARAMETERS OF THE CLOUDS.

No.	$\bar{C}$, GHz	N
1.	1	25
2.	1	35
3.	1.5	45
4.	1.5	55
5.	2	40
6.	2	45
7.	2	50
8.	2	60
9.	2.5	50
10.	2.5	55
11.	3	40
12.	3	60

The results of the experiment are presented in Table II. The data of the experiment is compared with the data, which would be obtained if all possible SoWC are generated under the given WC variation limits. An increase of the number of the SoWC is exponential if all possible combinations of the SoWC are generated. The results show that the proposed algorithm allows minimizing the number of the test task sent to the Intercloud from 6.25 % to 99.98 %.

An expansion of the WC variation limits and a correction of the WC of the best set did not give a reasonable effect. When the WC variation limits has been expanded up to [1..10], the best set of the WC of four parameters was found. Next expansions gave the same results, which is the same average time of the task fulfilment.

In order to ascertain the practical benefit of the selected best SoWC, the stream of 1,000,000 tasks has been generated in the simulation model. The average time of fulfilment of these tasks is presented in Table II.

The experiment results show that there is no reasonable reason to continue the algorithm when after two last iterations the average time of the task fulfilment is received the same, or difference is not significant (the user's needs are satisfied).

TABLE II.THE RESULTS OF THE EXPERIMENT.

WC variation limits	A number of all possible SoWC	A number of SoWC generated by the proposed algorithm	The best SoWC (including reiterating ones)	The average time of task fulfilment, min
1-2	16	15	ωW = 1; ωL = 2; ωC = 1; ωN = 1.	27.25991
1-5	625	96	ωW = 1; ωL = 3; ωC = 1; ωN = 3.	26.72516
1-10	10000	177	ωW = 1; ωL = 6; ωC = 2; ωN = 7.	26.70401
1-20	160000	258	ωW = 1; ωL = 11; ωC = 4; ωN = 13.	26.69552

1-40	2560000	339	ωW = 1; ωL = 11; ωC = 4; ωN = 13 or
			ωW = 2; ωL = 22; ωC = 8; ωN = 26
				26.69552

1-80	40960000	420	ωW = 1; ωL = 11; ωC = 4; ωN = 13 or
			ωW = 2; ωL = 22; ωC = 8; ωN = 26 or
			ωW = 4; ωL = 44; ωC = 16; ωN = 52
				26.69552

The best average time of the task fulfilment received applying the QoGS method and proposed algorithm was 26.695 min. In order to ascertain that the obtained result is better than the results obtained applying another method, the Estimate Response Time algorithm was used in the same conditions. The average time of the task fulfilment received applying Estimate Response Time algorithm was 29.357 min.

VII. CONCLUSIONS

In this paper the QoGS method has been chosen for the selection of the right Cloud in the Intercloud. The principle schema of the QoGS method application in the Intercloud has been presented. Since the QoGS method is sensitive to the weighted coefficients of the Cloud's parameters, the algorithm for selection of the right set of WCoP has been proposed.

The proposed algorithm has been tested experimentally. The results of the experiment show that:

1) It allows minimizing the number of the test tasks sent to the Intercloud significantly. Thus, the proposed algorithm is more efficient than other algorithms;

2) The situation during execution of the algorithm when after two last iterations the average time of the task fulfilment is received the same, or difference is not significant from the viewpoint of the user, is an indication that the algorithm should be stopped;

3) The considerable expansion of the WC variation limits does not give a reasonable effect. The results of the experiment show that the WC variation limits [1..10] are enough for selection of the best SoWC.

4) The average time of the task fulfilment received applying the QoGS method is 9.067% shorter than the average time applying the Estimate Response Time algorithm.

REFERENCES

[1] D. Bernstein, D. Vij, "Using Semantic Web Ontology for Intercloud Directories and Exchanges", in *Proc. of 11th International Conference on Internet Computing*, Las Vegas, 2010, pp. 18–24.

[2] D. Bernstein, D. Vij, "Intercloud Exchanges and Roots Topology and Trust Blueprint", in *Proc. of 11th International Conference on Internet Computing*, Santa Clara, 2011, pp. 135–141.

[3] A. Celesti, F. Tusa, M. Villari, A. Puliafito, "Security and Cloud Computing: InterCloud Identity Management Infrastructure", in *Proc. of 19th IEEE International Workshop Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE),* , Larissa, 2010, pp. 263–265.

[4] R. Buyya, R. Ranjan, R. N. Calheiros, "InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services", *Algorithms and Architectures for Parallel Processing Lecture Notes in Computer Science*, vol. 6081, pp. 13–31, 2010.

[5] D. Bernstein, "Keynote 2: The Intercloud: Cloud Interoperability at Internet Scale", *in Proc. of 6th IFIP International Conference Network and Parallel Computing, (NPC 09)*, Gold Coast, 2009, pp. 1–13.

[6] N. Grozev, R. Buyya, "Inter-Cloud Architectures and Application Brokering: Taxonomy and Survey", *Software: Practice and Experience,* pp. 1–22, 2012.

[7] D. McBride, M. Krznaric, J. Darlington, O. van der Aa, M. Aggarwal, D. Colling, "Running a Production Grid Site at the London e-Science Centre", in *Proc. of e-Science and Grid Computing (e-Science 06),* Amsterdam, 2006, pp. 153–170.

[8] M. Cafaro, I. Epicoco, M. Mirto, D. Lezzi, G. Aloisio, "The Grid Resource Broker Workflow Engine", *Concurrency and Computation: Practice and Experience*, vol. 15, pp. 1725–1739, 2008. [Online]. Available: http://dx.doi.org/10.1002/cpe.1318

[9] I. Gourlay, K. Djemame, J. Padgett, "Reliability and Risk in Grid Resource Brokering", in *Proc. of Second International Conference on Digital Ecosystems and Technologies*, Phitsanulok, 2008, pp. 437–443.

[10] R. Al-Ali, A. Shaikh-Ali, O. Rana, D. Walker, "Supporting QoS-based discovery in service-oriented grids", in *Proc. of IEEE Heterogeneous Computing Workshop*, Nice, 2003, pp. 1–9.

[11] J. A. Rashid, F. R. Omer, W.W. David, "G-QoSM: Grid Service Discovery Using QoS Properties", *Computing and Informatics*, vol. 21, pp. 363–382, 2002.

[12] J. Yu, R. Buyya, C.K. Tham, "QoS-based Scheduling of Workflow Applications on Service Grids", in *Proc. of 1st IEEE International Conference on e-Science and Grid Computing,* Melbourne, 2005.

[13] K. J. Liang, Q. Liang, Y. Yang, "The Strategy and Method of QoSParameter Processing for Grid Service Matchmaking", in *Proc. of Innovative Computing, Information and Control*, Beijing, 2006, pp. 381–384.

[14] R. Plestys, G. Vilutis, D. Sandonavicius, R. Vaskeviciute, R. Kavaliunas, "The measurement of grid QoS parameters", in *Proc of 29th International Conference on Information Technology Interfaces*, Cavtat/Dubrovnik, 2007, pp. 703–707.

[15] V. Pilkauskas, R. Plestys, G. Vilutis, D. Sandonavicius, "Improvement of WMS functionality, aiming to minimize processing time of jobs in Grid computing", *Elektronika ir Elektrotechnika (Electronics and Electrical Engineering)*, no. 7, pp. 111–116, 2011.

[16] G. Vilutis, D. Sandonavicius, "The complex evaluation of parameters influence on QoS", in *Proc. of 30th International Conference on Information Technology Interfaces*, Cavtat/Dubrovnik, 2008, pp. 905–910.

[17] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. Piatko, R. Silverman, A. Y. Wu, "An Efficient k-means Clustering Algorithm: Analysis and Implementation", *Pattern Analysis and Machine Intelligence*, vol. 7, no. 24, pp. 881–892, 2002. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2002.1017616

[18] M. K. Osman, M. Y. Mashor, H. Jaafar, "Detection of Tuberculosis Bacilli in Tissue Slide Images using HMLP Network Trained by Extreme Learning Machine", *Elektronika ir Elektrotechnika (Electronics and Electrical Engineering)*, no. 4, pp. 69–74, 2012.

[19] M. Sokolovaa, G. Lapalmeb. "A systematic analysis of performance measures for classification tasks", *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.ipm.2009.03.002

[20] D. H. Nguyen, B. Widrow, "Neural networks for self-learning control systems", *IEEE Control Systems Magazine*, vol. 3, no. 10, pp. 18–23, 1990. [Online]. Available: http://dx.doi.org/10.1109/37.55119

[21] J. S. R. Jang, "Self-learning fuzzy controllers based on temporal back-propagation" *Neural Networks, IEEE*, vol. 5, no. 3, pp. 714–723, 1992. [Online]. Available: http://dx.doi.org/10.1109/72.159060

[22] V. Paukštaitis, A. Dosinas, "Pulsed Neural Networks for Image Processing", *Elektronika ir Elektrotechnika (Electronics and Electrical Engineering)*, vol. 95, no. 7, pp. 15–20, 2009.

[23] I. Sahin, I. Koyuncu, "Design and Implementation of Neural Networks Neurons with RadBas, LogSig, and TanSig Activation Functions on FPGA", *Elektronika ir Elektrotechnika (Electronics and Electrical Engineering)*, vol. 120, no. 4, pp. 51–54, 2012.

[24] R. J. Maya, H. R. Maierb, G. C. Dandyb, "Data splitting for artificial neural networks using SOM-based stratified sampling", *Neural Networks*, vol. 23, no. 2, pp. 283–294, 2010. [Online]. Available: http://dx.doi.org/10.1016/j.neunet.2009.11.009

[25] J. A. Nelder, R. Mead, "A Simplex Method for Function Minimization", *The Computer Journal*, vol. 4, no. 7, pp. 308–313.

[26] A. Dambrauskas, D. Udris, "Calculation of Theoretical Statistical Characteristics of Simplex Search with the Aim Drift", *Elektronika ir Elektrotechnika (Electronics and Electrical Engineering)*, vol. 69, no. 5. pp. 17–20, 2006.

[27] R. Vershynin. "Beyond Hirsch Conjecture: Walks On Random Polytopes And Smoothed Complexity Of The Simplex Method", SIAM *Journal on Computing*, vol. 39, no. 2, pp. 646–678, 2009. [Online]. Available: http://dx.doi.org/10.1137/070683386

[28] A. M. Law, W. D. Kelton, *Simulation Modeling and Analysis*, McGraw-Hill, New York, 2000, p.760.