

# **A Web Service Discovery based on QoS Negotiation Approach**

**A. Meligy**

Math and Computer Science  
Department, Faculty of  
Science, Menoufia University,  
Egypt

**Passent El-Kafrawy**

Math and Computer Science  
Department, Faculty of  
Science, Menoufia University,  
Egypt

**Hanaafathi**

Faculty of Science,  
Menoufia University,  
Egypt

## **ABSTRACT**

Quality of Service (QoS) support in web services plays a great role for the success of web service selection. There exist many web services which has similar functional characteristics. Therefore, the massive growth of web services and also the forked view of services by both the parties (service provider and service consumer) make the task of service selection a complex job. To resolve this issue on divergent view of services, and to select an appropriate services, this paper propose Service discovery and selection model based on QoS Negotiation model that aims to facilitate the service selection process to consumer .QoS Negotiation model is responsible for creating negotiation with the service provider about the services that accept negotiation on its QoS attributes. After the success of negotiation process the service provider update the value of services attribute in the UDDI. The results show that the negotiation model enhances the discovery system performance and usability.

## **Keywords**

Web Service Discovery, Negotiation, Quality of Service (QoS).

## **1. INTRODUCTION**

The Web is no longer a simple way to access information; its growth makes it also a provider of services. Web Services refers to applications exposed by providers over the Web. They supply standard means of interoperating between different software applications, running on a variety of platforms and/or frameworks. Service discovery is the process of locating Web service, and retrieving Web services descriptions that have been previously published [1].

With the rapid growth of Web Services, a large number of Web Services with the same function are developed and published. Service discovery deals with the process of locating or discovering related service descriptions that describes a particular web service using the Web Service Description Language (WSDL). Whereas service selection deals with choosing a service implementation among the located services to satisfy the customer needs. The adoption of Negotiation while discovering the best service has several benefits including the discovery time optimization, the resolution of conflicts between the providers' and clients' preferences, and the improvement of the success rate.

Negotiation phase is the process conducted at the end to reach an agreement between concerned parties for a required level of QoS [2]. QoS for web services is defined as the non-functional properties of the service being provided to its users. These properties are also called metrics; common quality attributes for web services are Response time, Availability, Latency, Cost, and Throughput [2].

The type of the service and the QoS architecture play a major role in the way the Negotiation is conducted. However, the main weakness of these solutions is related to the difference between Consumer's behavior during their interactions and those of agents during the systems functioning. These systems do not simulate some common negotiators' behaviors observed during their real life interactions. Simulating these details about negotiators' behaviors will positively impact the system's performance and outputs. The process of service selection needs to be more appropriate; this paper propose Service discovery and selection model based on QoS Negotiation and reputation model that aim to facilitate the service selection process to consumer. Service discovery and selection model performs three main functions. Firstly, it finds the services that match their requirements. Secondly, if the request has a reputation requirement, the Reputation Module computes the trust factor of the consumer that assign reputation to services, then it return the reputation value of the services after computing. QoS Negotiation model is responsible for creating negotiation with the service provider about the services that accept negotiation on its QoS attributes. After the success of negotiation process the service provider update the value of services attribute in the UDDI. The list of services that is produced from the negotiation process is added to the list of selection returned to consumer and it meets the consumer requirements. Also, it has a QoS attributes with better quality than the returned from the selection before negotiation. Finally, a Discovery Unit returns the matching list of services to the consumers.

This paper is organized as follows. Section 2 outlines the related work to our approach. The proposed discovery and selection model is illustrated in Section 3. Section 4 evaluates the effectiveness of the new QoS Negotiation model and presents the experimental results. Section 5 presents the conclusion and the future work.

## **2. RELATED WORK**

Many significant research efforts have been produced in order to study the QoS-based Web Service description, Negotiation, and reputations systems. We provide an overview of some of these works as a context for the research discussed in the remainder of the paper.

Gabriel Silva, Itana de Souza Gimenes and Beatriz de Toledo [3] proposed an integrated web service negotiation process that takes into account human interaction and the use of different negotiation protocols. The negotiation process takes place in an environment that facilitates artifact reuse based on product line and feature modeling concepts. The negotiation process is composed of two main life cycles the Planning and Negotiation Agenda Settings; and the Negotiating and Establishing WS-Contract.

R.Raju, D.Dhivya, R.Saranya and S.I.Abbinaya[4] proposed a trusted Negotiation Manager (NM) framework that performs adaptive and intelligent mutual bargaining of SLAs between a service contributor and a service purchaser based on each party's high level business necessities. They also define an algorithm describes how the services are delivered with ultimate efficiency and time preservation. Negotiation Manager (NM) can't support multilateral negotiation.

Yan Kong, Minjie Zhang [5] proposed a negotiation-based method for task allocation under time constraints in an open, dynamic grid environment, where both consumer and provider agents can abstain from or enter into the environment freely. In this method, there is no central controller, and agents negotiate with each other for task allocation based only on local views. The draw back in this method is that it can't solve continuous task allocation in decentralized, dynamic, and open grid environments.

M.Sathya, P.Dhavachelvan and K. Vivekanandan[6] proposed an egalitarian based negotiation model that aims to select a required service for service consumers by achieving the egalitarian principle. The model focused on achieving a goal to search for a service with WIN – WIN situation between service consumer and service provider. The components in the negotiation model work collaboratively and the decisions are taken automatically.

Jing Zhao and Sherry X. Sun [7]proposed an architecture for QoS-based service composition where negotiation is incorporated to help service consumers exchange offers and counter offers with providers and to enable dynamic agreements on QoS attributes. The negotiation mechanism not only enhances the flexibility of the dynamic service composition but also makes the constraints for the composite service easier to be satisfied through adding collaboration among different negotiation processes between Service Negotiator and different service providers offering different component service.

Sandeep Kumar and Nikose. Mastorakis [8] presented Multi-Agent Negotiation based SWS Composition Models .These models are based on the concept that the negotiation can be performed between the SRA and the discovered SPAs, instead of SRA and the selected SPA. Also, a novel multi-agent based semantic web service composition approach presented based on Composition Model.

M. Swarnamugi [9] proposed an egalitarian based negotiation model that aims to search for a service with WIN – WIN situation between service consumer and service provider. The context information of services is represented in an ontology that is used to capture the objective experiences of service consumers with the service providers. This helps the egalitarian negotiation model to select the best service among the identified services by specifying the quality parameters.

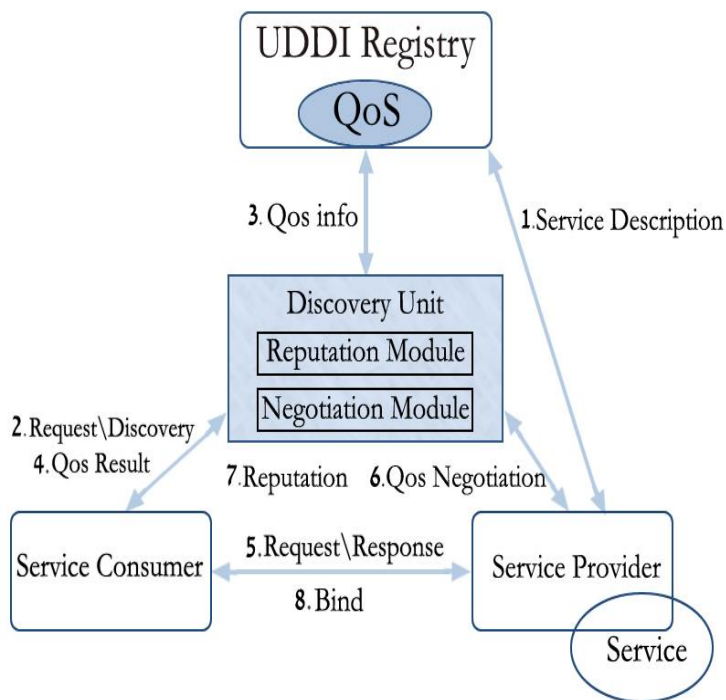
Serhani and Dssouli [10] proposed a broker-based architecture for Web Service selection and QoS management. The role of the QoS broker within the architecture was to support QoS provisioning and assurance in delivering Web Service. It introduced and implemented the concept of QoS verification and certification, which was used together with the QoS requirements in the selection process of Web Service. The proposed QoS broker was to be used as a third party Web Service publisher in UDDI registries. It was invoked when a user requests a Web Service with QoS requirements. They presented the operations of the QoS broker while processing user requests with QoS requirements. The broker arbitrates the QoS Negotiation between the consumers and the provider. The main weakness of the architecture is the cost of its adoption. In fact, the broker should be fully operational and its interface had to be known in advance to the providers and consumers.

Our Web Service Discovery Model based on QoS Negotiation reputation model is different in:

- The QoS Negotiation Model can negotiate about services that accept Negotiation. So, consumers can have more services in the matched list. The services produced from negotiation have better quality attribute than service from traditional selection and meets the customer requirements.
- By the time our QoS Negotiation service enhances the discovery system performance and usability by implementing a Negotiation process that is closer to Consumer' interactions.
- Computing the reputation of the services based on the trust factor of consumer.

### **3. THE PROPOSED MODEL**

The traditional Web Services publish and discovery model has three roles: service provider, service consumer and UDDI registry. In our model, the UDDI registry is extended with QoS information. In addition, the service provider and the service consumer have their own quality matrices. The third party and the Discovery Unit (DU) are added in our model. The proposed model is shown in Figure 1. The white boxes represent the existing roles in the current Web Services architecture and the shaded boxes/circles represent the new roles in our model. The UDDI registry stores QoS information of services. The Discovery Unit acts as third party between the service consumer and the UDDI registry to discover the Web Services that satisfies the consumer's functional, QoS and Reputation requirements. The service Reputation Module collects and processes service reputation from consumer and creates service Reputation Scores. Negotiation Module make Negotiation about the service that accept the Negotiation, after the Negotiation process succeeds then the new QoS values of the service are updated in the service description in the UDDI registry.



**Fig 1: Web Services Discovery and Selection Model based on Reputation and Negotiation**

As shown in Figure 1, the operation in the model is as follows: The Service Provider publishes the service description and the QoS information in the UDDI Registry. The Consumer requests a Service that matches its QoS matrices from the Discovery Unit. The Discovery Unit receives the request from service consumer and returns the services that match his/her requirements. Functional QoS and reputation requirements are specified in the discovery request. The Consumer receives result as a list of Web Services from the Discovery Unit. The consumer requests the services from the provider after receiving result of searching from the Discovery Unit. The QoS Negotiation Module can make Negotiation about a list of WS that accept Negotiation, in case of not finding a Web Service matching the consumer requirements. After the consumer gets the service, consumer can assign a Reputation Score for the Services. Then the Reputation Module calculates the new reputation value for the service based on the value assigned by the consumer, the trust factor of the consumer, and the previous reputation value of the service.

The calculation of QoS scores of services is performed by the equation below:

$$QoSScore_i = \frac{\sum_{i=1}^n QoS_i}{n}$$

Where  $QoSScore_i$  is the QoS score of service  $i$  and  $QoS_i$  is the value of the quality of services attribute of service  $i$ ,  $n$  is the total number of attribute of service  $i$ .

### 3.1 QoS Negotiation Module

The QoS Negotiation Module is responsible for making Negotiation on the services that accept Negotiation on its QoS attributes. There are two cases:

**First:** the Discovery Unit does not find a Web Service that match consumer requirements.

**Second:** the Discovery Unit finds a Web Service that match consumer requirements.

In the two cases the negotiation module starts the negotiation process, the negotiation process starts between two matrices first is the consumer metrics and the services provider that accept negotiation on its attribute is the second one. The negotiation module matches two matrices (consumer, provider). Each attribute in the services that accept negotiation has negotiation range to its value, the negotiation module negotiates committed the range of negotiation of each attribute until it reach to value that meets consumer requirements. The negotiation module then reaches an agreement with the service provider. Finally, the provider updates the QoS information in the UDDI registry with new values after the Negotiation process. The result of negotiation module in the first case is services that match consumer requirement instead of no services. But, the result in the second case is more services that meet the consumer requirement with good QoS attributes to choose from. In some cases, the services that result from negotiation have QoS attributes with higher quality than the one produced from traditional discovery.

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<_tModelDetailxmlns="urn:uddi-org:api_v3"
xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
<_tModel deleted="false"tModelKey="uddi:
tdev.com:carrentoneservice1" />
<name>CarRentOneTModel</name>
<_categoryBag>
<keyedReferencekeyName="trustFactor"keyValue="0.2"
tModelKey="uddi: tdev.com:carrentoneservice1" />
<keyedReferencekeyName="price"keyValue="0.01"tMod
elKey="uddi: tdev.com:carrentoneservice1" />
<keyedReferencekeyName="priceToRange"keyValue="1.
0"tModelKey="uddi: tdev.com:carrentoneservice1" />
<keyedReferencekeyName="ThroughputToRange"keyVa
lue="1000"tModelKey="uddi:
tdev.com:carrentoneservice1" />
<keyedReferencekeyName="negotiable"keyValue="true"
tModelKey="uddi: tdev.com:carrentoneservice1" />
<keyedReferencekeyName="providerTrustFactor"keyVal
ue="0.8"tModelKey="uddi:
tdev.com:carrentoneservice1" />
<keyedReferencekeyName="Throughput"keyValue="800
"tModelKey="uddi: tdev.com:carrentoneservice1" />
<keyedReferencekeyName="type"keyValue="MONO_IN
CREASING"tModelKey="uddi:tdev.com:carrentoneservi
ce1" />
<keyedReferencekeyName="responseTimeToRange"key
Value="1.2"tModelKey="uddi:
tdev.com:carrentoneservice1" />
<keyedReferencekeyName="responseTime"keyValue="0.
08"tModelKey="uddi: tdev.com:carrentoneservice1" />
<keyedReferencekeyName="availabilityToRange"keyVal
ue="100"tModelKey="uddi:
tdev.com:carrentoneservice1" />
```

```
<keyedReferencekeyName="availability"keyValue="99.99"tModelKey="uddi:tdev.com:carrentoneservice1" />
</categoryBag>
</tModel>
</tModelDetail>
```

**Fig 2: the tModel with the QoS Information**

Figure 2 shows a tModel containing QoS information for a service named "carrentoneservice1". Each QoS attribute has two values the first its real value assigned by provider of service, the next value is the range negotiation value. As shown in the service named "carrentoneservice1", the price attribute value is 0.01 and it accepts Negotiation to range 1.0. The QoS Negotiation model starts Negotiation committed to the Negotiation range value of each attribute in service. After the QoS Negotiation reaches an agreement the service provider updates the QoS information in the UDDI registry with new values after the Negotiation process. Then the list of service after negotiation is returned to the consumer.

The service Reputation Module is responsible for collecting data from the service consumer, processing data, updating the Reputation Scores for related service provider. The QoS Reputation Score is calculated based on the reputation assigned by the service consumer; service consumer trust factor and the current reputation value of the service. The consumer that uses a service assigns it a reputation value. So, Reputation Module built a good trust between the provider and consumer of services through the correct reputation calculation.

### 3.2 Discovery Approach

The discovery approach includes the functional and nonfunctional properties. The functional properties include the attributes that the service must have to perform the service and the nonfunctional properties present the QoS including the service reputation attribute. Figure 3 shows a simplified version of service selection Based on Reputation and negotiation algorithm that based on the algorithm in [11] When the discovery Unit receives a discovery request, it executes match (line 2) which return a list of services LS1 that meet the functional requirements. If QoS requirements are specified, QoS match (line 4) function is executed next on the set of services LS1 and it returns a subset of services that meet the QoS requirements LS2. selectServices(line 6) always returns a list of M services to the customer where M denotes the maximum number of services to be returned as specified in the discovery request. If QoS requirements are not specified; selectServices returns M randomly selected services from LS1. If only one service satisfies the selection criteria, it returns this service to the customer. In the case where no reputation requirement is specified, qosRank(line 15) calculates QoS scores of the services in LS2 and returns a list of services LS3. If QoS requirements are specified the Discovery Unit returns a list of matched services with the minimum requirements. Services are sorted in descending order based on their QoS scores. The QoS score is calculated in the range of 0 to 1 for each service. selectServices(line 16) returns the top M services to the customer. If M is not specified, one service is randomly selected and returned. In the case where a reputation requirement is specified, reputationRank(line 9) calculates reputation scores of the services in LS2 and returns a filtered list of services LS4 containing only those services that have a reputation score equal to or above the specified. In case of serviceNego(line 12) a list of services that accept

negotiation are returned in Accepted LS5 that add to the list of LS1 and LS4.

```
// find services that match QoS requirements and services that
// accept negotiation
1 QoSMatch (services, QoSRequirements, ReptRequirements,
acceptnegotiation) {
// find services that meet the functional requirements
2 matches= Service (function Requirements);
3 if QoS requirements specified {
// match services with QoS information
4 QoS matches = QoSService(matches, qosRequirements); }
5 else {
// select max number of services to be returned
6 return selectServices (matches, maxNumServices,
"random"); }
7 matches.add(s);
8 if reputation requirements specified
// rank matches with QoS and reputation information
9 matches=reputationRank(QoSMatches,QoSRequirements,
repuRequirements);
// select services according the max number of services to be
// returned
10 return selectServices (matches, maxNumServices,
"byQOS");
// if service accept negotiation
11 if (Service.AccNego)
{
12 Accepted=serviceNego(QoSRequirements);
Matches.add;
// list of service accept negotiation
13 return selectServices (matches, maxNumServices,
"acceptnego");
}
14 else {
// matches with QoS information
15 matches = QoSRank(qMatches, qosRequirements);
// select max number of services to be returned
16 return selectServices (matches, maxNumServices, "by
Overall"); }
```

**Fig 3. Web Services Selection Based on Negotiation and Reputation Algorithm**

## 4. EXPERIMENTAL RESULT

This section presents experimental results to evaluate the effectiveness of our selection algorithm. A number of programs are used to simulate various roles in the model.

- A Web Service acts as customer by using a simulation program that generates service requests

with different QoS and reputation requirements to be sent to the Discovery Unit.

- A Discovery Unit program that has three functions:
  1. It receives simulated requests, retrieves service QoS information, and Reputation Scores, if necessary, and finally runs the algorithm to select services for the consumer.
  2. It has a Reputation Model program to calculate Reputation Score to a service.
  3. It has a QoS Negotiation Model program to negotiate on the QoS attributes of Web Service according to negotiation range value.

In the following experiments, we assume that all the services provide the same functionality and that every consumer request has the same functional requirements, which are satisfied by all the services. The QoS parameters are price, response time, availability, throughput and reputation.

**Experiment 1**

This experiment demonstrates the probability of selecting a service, which best meets a customer’s requirements, without using negotiation in the discovery request. Table 1 summarizes the reputations, QoS status of 20 services (S1-S20), where High QoS means high value, lowQoS means low value and the status of services negotiation, services accepting negotiation (AN), services not accepting negotiation (NOT AN).

**Table 1.Summary of Services Negotiation Status**

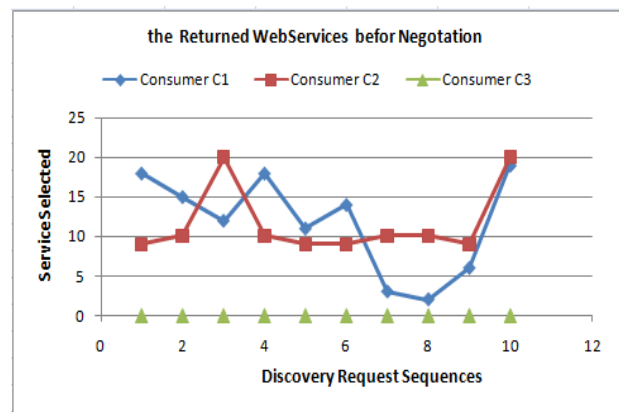
Services				Negotiation status	QoS [Response time(Sec), Availability (%),Throughput (transaction/sec), Price(\$ per Transaction),Reputation]
S14	S15	S16	S4	NOT AN	Low QoS
S13	S6	S7	S8		
S9	S10	S11	S12	AN	High QoS
S1	S2	S3	S5		
S17	S18	S19	S20		

We ran the experiment for each consumer for all services. For each consumer and services, the same service discovery request was run 10 times and the service selected was recorded. A service is selected for customers C1, C2 and C3 from services S1 to S20 without negotiation, according to table 2 that shows the QoS requirements of three consumers C1, C2, C3.

**Table 2.Consumers’ QoS requirements**

Consumer	Price	Response time	Availability	Throughput	Reputation
C1	No	No	No	No	No
C2	0.03	0.04	90%	600	5/10
C3	0.02	0.02	99%	1000	9/10

As shown in figure 3 we find that: first, The list of services returned to C1 is all services from S1to S20 randomly, because they need only functional requirements. Second, the list of services returned to C2 is S9, S10, and S20. Last, the list of services returned to C3 is an empty list because there is no service that matches the consumer C3 requirements.

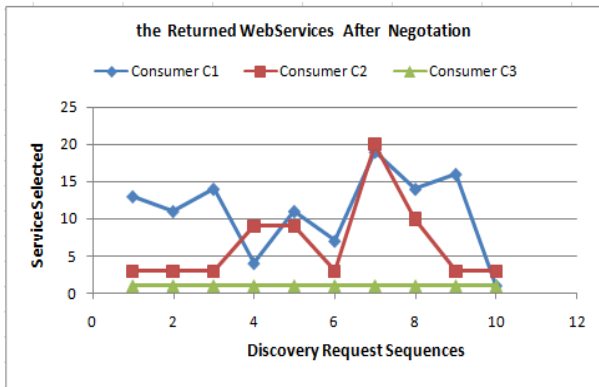


**Figure3. Returned Services before Negotiation**

**Experiment 2**

This experiment shows the list of returned services that meets the QoS and reputation to the consumer’sC1, C2 and C3 requirements for the same request in table 2. For each consumer and services, the same service discovery request ran 10 times and the service selected was recorded. Services selected for customers C1, C2 and C3 from services S1 to S20 with negotiation were recorded.

As shown in figure 4 we find that: first, The list of services returned to C1 is all services from S1to S20randomly. Second, the list of services returned to C2 isS9, S10, S20 and S3.So the number of services returned to consumer after negotiation is more than the services before negotiation. Also, S8meets the consumer requirements and has a high quality attributes than S9, S10, S20 although this service is not in the same group (High QoS). But, the value of the QoS attribute is not the same. Last, the list of services returned to C3 is S1 after negotiation process succeeds; S1 is the service that meets consumer requirements.



**Figure4. Returned Services after Negotiation**

Table 3 show the results in five iterations, and the ranking order of services in case of traditional discovery and discovery with negotiation according to the request of consumers C2 in table2. As shown the returned list will be different the order is changed because the services returned after negotiation S3 is best match the consumer requirements than the services in traditional discovery.

**Table 3: Returned Services in order after /before Negotiation**

Request	Services Returned before Negotiation	Services Returned after Negotiation
1	S9,S10,S20	S3,S9,S10
2	S9,S10,S20	S3,S9,S10
3	S9,S10,S20	S3,S9,S10
4	S9,S10,S20	S3,S9,S10
5	S9,S10,S20	S3,S9,S10

## 5. CONCLUSION

In this paper, we have introduced Web Service Discovery Model in which selecting Web Services was based on a new negotiation approach. The approach is responsible for creating negotiation with the service provider that accepts negotiation on its QoS attributes. It matches the QoS attributes of the service accept negotiation with the one of consumer, then the negotiation process is start after success of negotiation the services returned to consumer and the new value of QoS is updated by services provider in the UDDI registry .Future enhancement to the current work aims to apply cloud in negotiation system to develop Multi-Agent based negotiation system to increase the efficiency of negotiation process.

## 6. REFERENCES

- [1] Bromuri S., Urovi V., Morge M., Stathis K., and ToniF., 2009. A multi-agent system for service discovery, selection and Negotiation. AAMAS.
- [2] M. A. Serhani, R..Dssouli.QOS Integration IN VALUE ADDED WEB SERVICES. The Second International Conference on Innovations in Information Technology (IIT'05) 2005.
- [3] Gabriel Silva, Itana de Souza Gimenes and Beatriz de Toledo, "Inter-organizational Negotiation of Web services" ,International Journal of u- and e- Service, Science and Technology Vol.6, No.5 (2013), pp.97-114 <http://dx.doi.org/10.14257/ijunesst.2013.6.5.10>.
- [4] R.Raju, D.Dhivya, R.Saranya and S.I.Abhinaya, "SLA Negotiation for Web Service Consumption Based on Analytic Hierarchy Process", International Journal of Science and Modern Engineering (IJISME) ISSN: 2319-6386, Volume-1, Issue-3, February 2013. Yan Kong, Minjie Zhang, XudongLuo, DayongYe.A Negotiation Method for Task Allocation with Time Constraints in Open Grid Environments. ANAC (2013) .
- [5] M.Sathya,P.Dhavachelvan and K. Vivekanandan, "Egalitarian based Negotiation model for QoS based Web Service Selection", M.Sathya et al. / International Journal of Engineering and Technology (IJET).
- [6] Jing Zhao and Sherry X. Sun, "ServNegotiator: A Negotiation Based System for Service Composition", proceeding of the international multi conference of engineers and computer scientists 2011 vol I,IMCES March 16-8,2011,HongKong.
- [7] SandeepKumar andNikose. Mastorakis, "Multi-Agent Negotiation based Semantic Web Service Composition Models", Recent Advances in Software Engineering, Paralle and Distributed System.
- [8] M.Swarnamugi, "Context-Aware Web Service Selection – A Negotiation Model ", proceeding in International.
- [9] Journal of Advanced Research in Computer Science and Software Engineering ,Volume 4, Issue 3, March 2014 ,ISSN: 2277 128X .
- [10] M. Adel &RachidaDssouli "QoS broker based architecture for efficient Web Services" In proceedings IEEE International Conference on Web Services (ICWS'05), 2005.
- [11] Ziqiang Xu, Patrick Martin, Wendy Powley, Farhana Zulkernine, "Reputation-Enhanced QoS-based Web Services Discovery", ICWS, 2007, 2007 IEEE International Conference on Web Services, 2007 IEEE International Conference on Web Services 2007, pp. 249-256, doi:10.1109/ICWS.2007.152