Applied Mathematical Sciences, Vol. 9, 2015, no. 65, 3213 - 3225
HIKARI Ltd, www.m-hikari.com
http://dx.doi.org/10.12988/ams.2015.54309

Machine Driven and Effectual Web Service Discovery by Hybrid Algorithm

[1] A. Amirthasaravanan, [2]Paul Rodrigues and [3]R. Sudhesh

[1]Department of Information Technology
University College of Engineering Villupuram, Villupuram, India

[2]Department of Computer Science and Engineering
Velammal Engineering College, Chennai, India

[3]Depatment of Mathematics
University College of Engineering Villupuram, Villupuram, India

Abstract

Web Service is an open standard based web applications that interact with other web applications for the reason of exchanging data. Web service is self telling service that will perform fine tasks and can be accessed through the web. Web services discovery have the Hyperlink-Induced Topic Search (HITS) linking analysis algorithm that rates Web pages. The increasing convolution of compositions and the ration of people and services involve adaptive and context-aware contact models. This method is an imperative for the internet penetrating techniques. But this emerging technique faces Topic drift problem which is a pitfall in web service discovery. This paper we introduce HYBRID algorithm to endorse Topic drift drawback. This algorithm is combination of Multi Objective Bat Algorithm (MOBA) and Multi objective cuckoo search (MOCS) algorithms .MOBA is used for the multi data access and the MOCS algorithm is used for fast access techniques .By using this HYBRID algorithm the efficiency and accuracy can be improved. So this technique is called as HYBRID algorithm (MOBA &MOCS). The experimental evaluation shows that this algorithm reduces topic drifting problem and increases the accuracy of the web services discovery.

Keywords: Hyperlink-Induced Topic, Search Human Provider Service, Hybrid Algorithm, Multi Objective Bat Algorithm, Multi -Objective Cuckoo Search

1. Introduction

Web Service is a standards-based, language agnostic software entity that have specially formatted requests from other software entitle on remote machines via seller and transport neutral communication protocols producing application tough responses. Web service discovery is the process of discovering or locating one or more related documents that discover a particular web service using the WSDL. SOA is an architecture paradigm that focuses on building system through the use of dissimilar web service, integrating them unitedly to make up the whole system. Web services have paved the way for a new flux of compostable systems. Services play an important role in meeting organizations business objectives because process stakeholders can contrive, implement, and perform business processes using web services and languages such as the Business Process Execution Language (BPEL) [1]. Users and developers have the power to use services in diverse applications since services present a well-determined and programmable interface. In march-centric relationship, a top-down access is normally taken by fixing process actions and labors anterior to deploying and doing the process. Ahead creating the mold, the designer must fully understand each abuse in the march. Flexibility in composition models are fixed since surprising changes expect remodeling of the march. It is important to back up adaptively in collaborations and compositions. A significant role towards adaptive marches is the power to sustain the murder of ad hoc activities and flexibleness in human fundamental interaction to reply to surprising events. The most extensively established and successful type of service is the *XML Web servic*e. This type of service has two fundamental requirements: It communicates via Internet protocols; it sends and receives data formatted as XML documents.

2. Existing System

2.1 The Skilled Web

Current system approach is based on the Human-Provided Services perception enabling knowledge workers to propose their skills and expertise in service-oriented systems. Web services discovery is very much subjective by (behavioral) trust and context mechanisms. Expected information is once in a while updated to capture with dynamism changing interaction preference and trust relations. The current HITS based discovery can be computed in a connected way, thus empowering full personalization at runtime. These approaches in personalized expertise mining algorithm usually perform offline interaction analysis. This empirical evaluation has shown that HITS reveal the desired properties; trust and evaluation weights influence hub and authority scores which intern Impact on rank-

ings. Ensure that this algorithm discovers experts which are fine connected to further experts.
But on the other hand present HITS based algorithm, where later than query processing the services are discovered according to the Hyperlink-Induced Topic Search (HITS-page ranking) which do not impose present web search engine strategy since web is dynamic in nature. These will lead to **Topic Drift** and **Time Stamp** problems [1]. Topic drift is getting irrelevant and unrelated web services discovered by the search engine. Time taken (in ms) to discover a query from UDDI repository and its response time to the user is more. In order to rise above this problem we go for Hybrid algorithm.

2.2 MOBA and MOCS

A metaheuristic can be defined as an iterative generation process that guides a subordinate heuristic by combining intelligently different concepts to explore and exploit the search space. To catch closely optimal solutions, information must be structured using learning strategies. Optimization is a process of determining the best solution to make something as functional and effective as possible by minimizing or maximizing the parameters involved in the problems.

2.2.1 BA

Bat algorithm (BA) was developed by Xin-She Yang in 2010 and it is a metaheuristic optimization algorithm based on echolocation of microbats with changeable pulse rates of emission and loudness. The microbats use echolocation extensively as sensor to detect hindrance. Theses bats emit sound pulse and sense for echo as it spring back from surroundings with a time delay [2, 3]. Bats are able to distinguish targets by variation of the Doppler Effect provoked in surroundings. The echolocation behavior can be associated with objective function that needs to optimize.
In order to simplify the algorithm, three ideal rules are proposed:

- All bats use echolocation to sense distance. The bats know the difference between prey and hindrance.
- A bat fly indiscriminately with velocity at location with a fixed frequency, varying wavelength and loudness to search for prey. The frequency or wavelength of produced pulse can be adjusted automatically as well as the rate of pulse emission which is depending on proximity of the objective.
- The loudness can be differing from large (positive) to minimum value.

In simple the Algorithm can be depicted as **e**ach virtual bat flies indiscriminately with a velocity at position with a varying frequency and loudness. By way of it examines and finds its prey, it fluctuates regularity, loudness and pulse emission rate. Exploration is strengthened by a local arbitrary walk. Selections of the best keep on until stop criteria are met.

2.2.2 CS

Cuckoo Search (CS) [5, 7] is an optimization algorithm was developed by Yang and Deb in year 2009. Cuckoo Search, is inspired by the life of a bird family, Cuckoo birds have an antagonistic imitation in which females seize and lay their fertilized eggs in other birds' nests. If the host bird appreciate that the egg does not belong to it, it either chucks away or discards its nest and builds a new one elsewhere .According to Yang and Deb (2009), the CS algorithm is based on three assumptions:

- Each cuckoo puts one egg, which represents a set of solution co-ordinates, at a time and dumps it in a random nest.
- A portion of the nests comprising the best eggs, or solutions, will take over to the subsequent generation.
- The number of nests is unchanged and there is a probability that a host can discover an alien egg. If this ensues, the host can either discard the egg or the nest and this result in building a new nest in a new location.

2.2.3 Multi Objective

In simple multi-objective is the method of optimizing systematically and simultaneously a collection of objective function are called multi objective optimization. The objectives are generally inconsistent; avert simultaneous optimization of each objective. Most of the real engineering problems actually do have multiple-objectives like cost minimizing, maximizing performance, maximizing consistency, etc. These are difficult but realistic problems.

There are two key goals in multi-objective optimization (MO):

• To attain solutions as close to the true Pareto front as possible.

• To generate results as diversely as possible in the non-dominated front.

3. Proposed System

Our proposed system is a combination of Multi Objective Bat Algorithm (**MOBA**) and Multi Objective Cuckoo Search Algorithm (**MOCS**). We combine these two algorithms in-order to increase the efficiency and accuracy. So this technique is termed as **HYBRID** algorithm. MOBA is used for multiple data access and MOCS algorithm is used for fast access techniques. In our Future work we also include the security in this Hybrid algorithm.

4. Multi Objective Cuckoo Search Algorithm

The main aim of this paper is to show the efficiency of the cuckoo search algorithm [5] in solving UCP and to propose a fresh MOCS to reject the role

of the conclusion maker (system operator) in setting the reeling reserve in UCP. In MOUCP problem [12], is similar to other evolutionary process, cuckoo search algorithm begins with an initial frozen number of cuckoo nests [7]. At the end of every propagation of CSA, the size of cuckoo nests gains when compared to the sign number of nests. Hence the hazy set theory [6] is used to select the best via media solution (which is equal to the sign number of nests) that would be alert for the coming generations [11]. Inclusion of hazy set theory will be thrown away or vacate the solution (nest) which is a far away from the best root.

Algorithm 1: *Outline of discovery approach*

Cuckoo Search (CS) Pseudo-code
start
identify objective function.
generate preliminary population of host nests.
while (criteria not met)
{
get a cuckoo randomly;
evaluate the fitness of it;
select a nest from the population randomly;
if(fitness of selected nest is high)
end
discard a fraction of worse nests and build new
ones at new locations;
keep the best nests (solutions);
rank the nests and discover the present best;
}
post process results and visualization;
End

5. Multi Objective Bat Algorithm

It is very shiny problem working algorithm and it is improved to apply more engineering problems [4].

Multi Objective Bat Algorithm (MOBA) is a best select for finding optimal solutions. A swarm of MOBA flows in the graph for detecting optimal or close optimal solutions.
MOBA algorithm consists of two major types of parameter:
• Static parameters
• Dynamic parameters

Algorithm 2: *Outline of discovery approach*
start
define objective function

initialize the population of the bats
define and initialize parameters
while(Termination criterion not met)
{
generate the updated solutions randomly
if (Pulse rate (rand) >current)
select a solution among the top solution generate the local solution around the chosen best ones.
end if
generate a updated solution by flying randomly
if (Pulse rate (rand) >current) select a solution among the best solution generate the local solution around the selected best ones.
end if
generate a updated solution by flying randomly
if (loudness & pulse frequency (rand) < current)
accept the updated solutions increase pulse rate and reduce loudness
end if
rank the bats and find the present best
}
Output

6. Importance of MOBA Algorithm

The following three reasons provide the necessary and grandness of the MOBA algorithm:

- It provides good quality answers using average values (AVG).
- MOBA algorithm has fast overlap when compared to other methods.
- It is also elastic in the dynamic environment and pop-up threats are easily merged.

7. Hybrid Algorithm

The HYBRID algorithm explains the following steps for combine the two (MOBA & MOCS) algorithms. The HYBRID algorithm is explained the following steps:

We test the effect of each search to discover the services of the proposed technique and existing WSM process by specifying the keyword in each test scenario. WSDL URL is used to utilize or integrate the service with user application. Description about the services is used to search the service. The existing WS matchmaking process and proposed techniques were used to find out the exact services from database. After searching the list of services, the services can be filtered based on the exact matching and number of hits.

The accuracy of a measurement system is the degree of closeness of measurements of a quantity to its actual (true) value [8]. The above HYBRID algorithm explains the steps of effective principle by grouping the Multi objective cuckoo search and Multi Objective Bat algorithm. First of all the query has been given, then expert seeker gives the trust query as soon as the query has been searched using the MOCS. At last in most situations MOBA will be used. Then the response will be displayed.

8. Implementation

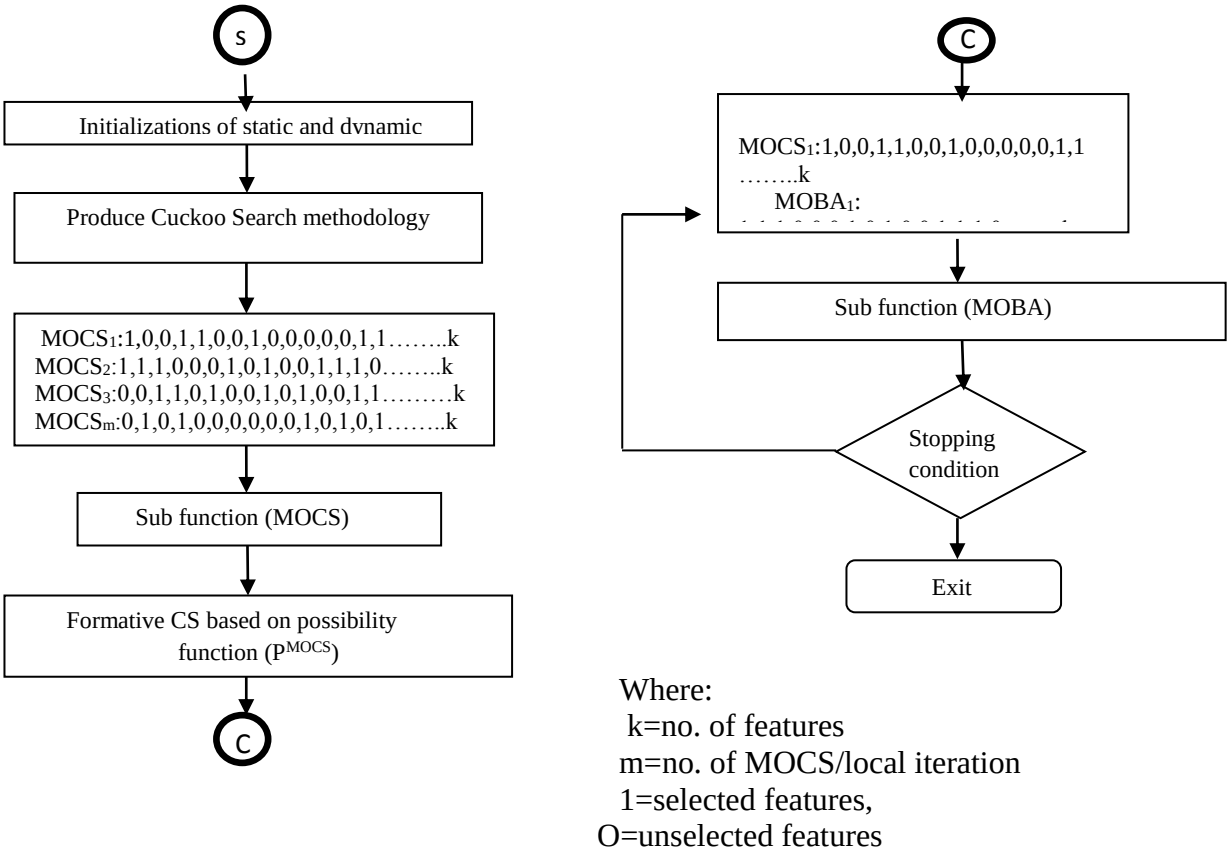
In our proposed system we are facilitating web services discovery using Hybrid algorithm (MOCS and MOBA) and optimizing the web services discovery. In this process the user profile is maintained for privacy issues. And a separate UDDI is created to publish the web services.

In this Hybrid web services discovery, the query of the user is processed from client side. The Hybrid algorithm (MOCS and MOBA algorithm) which is at server side helps to process the query request by considering Multi objective searching technique. This technique helps to retrieve services from UDDI by considering the multi objective (keyword) which is specified in the query request.

This multi objective consideration will overcome topic drifting problem. And MOBA algorithm helps to prioritize the web services according to the velocity of the services. Here the date and hits of a web service is considered as velocity fields. Based on the velocity rank, the web services are displayed. The response time is reduces due to effective discovery and by quick ranking process. By this our proposed system provides the required web services to the user efficiently.

```
  Hybrid (Query1, Query2) {
  ObjectExpansionMOCS(Mixquery1,Mixquery2);
  AddExtraObjectMOBA(Mixqurey1, Mixqurey2);
  For Object=Mixqurey1 (MOCS), Mixquery (MOBA);
  Final result (URL 1, URL 2, Ext);
}
  And the Result function is defined as:
  Result (Mixquery1, Expant query) {
  Re (Mixquery1); Re1 (Expert query); Mix Re, Re1
  (State1); AddResultQuery (State1);
  Final Result ();
}
```

Pseudo code 1. Pseudo code for HYBRID



Where:
k=no. of features
m=no. of MOCS/local iteration
1=selected features,
O=unselected features

Figure 1. Hybrid Algorithm Flow chart

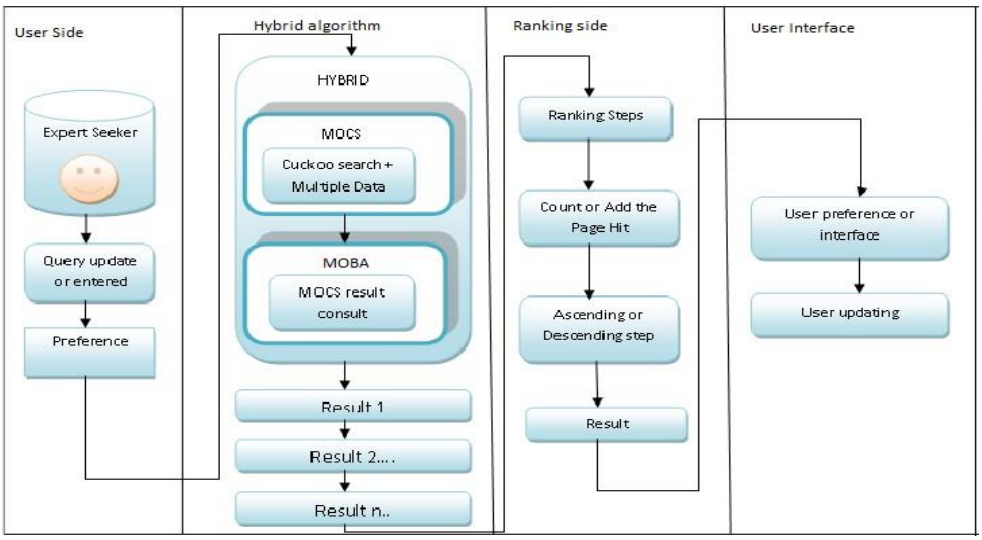


Figure2. System Architecture

9. System Architecture

System Architecture explains about how the rank or page ranking will be given and the working of Hybrid algorithm. The Hybrid is the combination of MOCS and MOBA. At MOCS algorithm Multiple Objects are combined with cuckoo search. The result of MOCS should be given to the MOBA. Then these results are forwarded to the ranking steps. The rank will be given based on which one is given to the first link. Then at last the User Interface working will be done. The QoS-based service range trouble involved in service discovery is, how to select one check for each concerning task from its corresponding existing nominee service group, so that the overall QoS of the construct complex service can be maximized. The values for QoS parameters are randomly chosen for the first time and Updated on every single rating. For each web service rating process three lift web services and four candidate web helps are counted.

The most suited web services will be recalled from the database, based on the fitness assess. Figure of the database includes the values put for each arguments of the tested web service. The arguments considered for rating are cost, response time, availability and dependability.

QoS (IM) =<Qcost MOBA(I),
QresponseMOCS(M)QavailabilityMOBA(I),
QreliabilityMOCS(M)

The MOCS says low level formatting of the nests and the arbitrary primary solution is performed. The current best nest is chosen by the arbitrary walk and then the development of the choice of fitness is done. Else apply levy's flight for the rating. Execution is bore out till all the result is made.

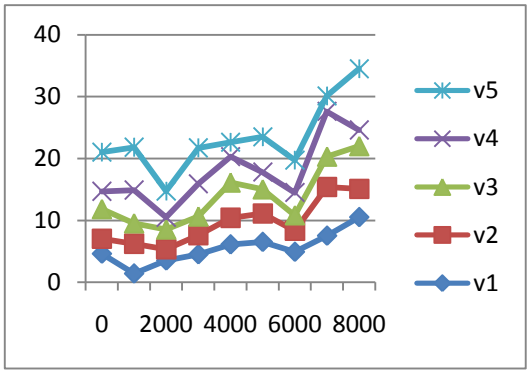


Figure 3. Individual graph for* MOBA *with using MOCS

The flow chart is the Hybrid algorithm, thus the intifada of the argument that need to be scheduled is performed. Then the count of labor that has to be planned is keyed. Scheduling of imagination to the tax is carried out with the avail of Hybrid algorithm. The above Diagram explains about the practical value examples. The v1, v2, v3, v4, v5 and 0-8000 is the x and y axis and the values will be assigned.

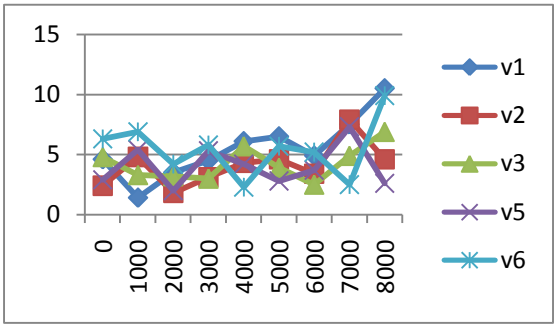


Figure 4. Function diagram for MOBA with using MOCS

Once the resource is allotted, the scheduler will allot the task to the doer and the slaying of task time is computed. Hybrid algorithm uses the MOCS and MOBA algorithm. In MOBA algorithm, the last fitness rate is the optimum value for a good result to the given problem.

The experimental results display that the Hybrid algorithm method is a workable and effective path to detect the optimum result. Multiple Data object will be combined. The cuckoo search contains the three constrains which are explained earlier. Then Multiple Data testing results of both the MOBA and MOCS are produced.

Contains X variables	**Quality of optimum solution**	**The solution Quality of MOBA vs MOCS**	**Average no. of Iterations of the MOBA Vs MOCS**
10 × 6	3800	3800	3.3
10 × 10	8706.1	8706.1	12.9
10 × 15	4015	4015	30.9
10 × 20	6121	6121	18.7
10 × 28	12400	12400	11.9
5 × 39	10617	10536.6	100
5 × 50	16537	16407	100
6×45	10756	1234	400
7×55	1234	14579	258
8×69	1567	23513	45.06
9×72	14578	1235	345.91
10×87	10234	15986	455.97
10×90	12096	13075	509
10×40	13086	12096	408.9
12×10	17479	19867	108.4
12×30	18654	17547	30.5
14×50	154769	12654	109.3
16×60	19756	18973	98.0
18×80	186754	198645	186.3

Table 1. The Quality and Average values of MOBA Vs MOCS

Based on these results the graph is drawn as shown below. The table contains the following columns such as X variables, Quality of optimum solution, Solution quality of MOBA vs MOCS and Average no of Iteration of the MOBA vs MOCS. The results are based on the experimental results.

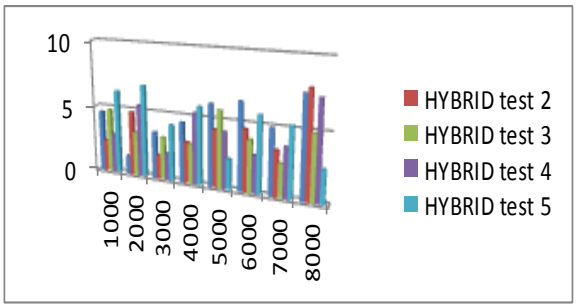


Figure 5. HYBRID test

The result of Hybrid Algorithm (MOBA &MOCS) shows that the hybridization provides more accuracy with considerably avoidance of Topic drifting and which in-turn minimizes Response time. The algorithm was tested with different user queries and Hybrid algorithm has produced effective search optimization in web based searching.

10. Conclusion

In this paper our approach of combining the two algorithms is done. By this Hybrid algorithm we have increased the efficiency of the web based searching. We have combined MOBA and MOCS, which helps in optimization of the web services discovery and it also over comes topic drifting problem in HITS algorithm and makes an effective search optimization. Hybrid is based on the context or content based search. Here we have concentrated only on the discovery approach. In the future if web services composition is integrated then performance level will be increased and an effective web services can be achieved.

References

[1] Ben Hamida S., Schoenauer M. (2000) "*An adaptive algorithm for constrained optimization problems*". in Proc. of Parallel Problem Solving from Nature, vol. VI, pp 529–538. http://dx.doi.org/10.1007/3-540-45356-3_52

[2] D. Brabham, (2008) "*Crowdsourcing as a Model for Problem Solving: An Introduction and Cases*," junction, vol. 14, no. 1, pp. 75-90. http://dx.doi.org/10.1177/1354856507084420

[3] Cai et. al., (2010); Wang and Singh, 2007; "*Automated web services*," Vinothkumar and Selvan.

[4] Cai J. J., Ma XQ, Li Q, Li LX and Peng HP, (2010)." A *multi-objective chaotic ant swarm optimization for environment/economic dispatch. Electrical Power and Energy System*," Vol. 32, pp. 337-344. http://dx.doi.org/10.1016/j.ijepes.2010.01.006

[5] P. Dharanyadevi, P. Dhavachelvan, S. K. V. Jayakumar, P. Sujatha (2011) "*Context Aware SOA Based Discovery in Mobile Environment by Bi-Matchmaking Technique*" IEEE-International Conference on Recent Trends in Information Technology, ICRTIT 2011, IEEE MIT Anna University, Chennai. http://dx.doi.org/10.1109/icrtit.2011.5972326

[6] P. Dharanyadevi, P. Dhavachelvan, S. K. V. Jayakumar (2011) "*Event Matchmaking Technique in Dynamic Web Service Discovery*" IEEE-International Conference on Recent Trends in Information Technology, ICRTIT 2011, IEEE MIT, Anna University, Chennai. http://dx.doi.org/10.1109/icrtit.2011.5972407

[7] Khosravifar, B., Bentahar, J., Moazin, A., &Thiran, (2010) "*On the reputation of agent-based on web service*".In proceeding of the 24'Th AAAI Conference on Artificial Intelligence (AAAI). pp. 1352-1357.

[8] D. Schall, H.-L. Truong, and S. Dustdar, (2008)" *Unifying Human and Software Services in Web- ScaleCollaborations*," IEEE Internet Computing, vol. 12, no. 3, pp. 62-68. http://dx.doi.org/10.1109/mic.2008.66

[9] J. M. Kleinberg, (1999) "*Authoritative Sources in a Hyperlinked Environment*," J. ACM, vol. 46, no. 5, pp. 604-632. http://dx.doi.org/10.1145/324133.324140

[10] Tavakkoli-Moghaddam, Azarkish, (2011) "*A new hybrid mutli-objective Pareto archive PSO algorithm for a bi- Objective job shop scheduling problem*", Elsevier Expert Systems with Applications. http://dx.doi.org/10.1016/j.eswa.2011.02.050

[11] S. Rao Rayapudi, (2011)" *An Intelligent Water Drop Algorithm for Solving Economic Load Dispatch Problem*" International Journal of Electrical & Electronics Engineering; Feb2011, Vol. 5 Issue 1, pp.43.

[12] Shah-Hosseini, H. (2008). "*Intelligent water drops algorithm: A new optimization method for solving the multiple knapsack problem*" International Journal of Intelligent Computing and Cybernetics, 1, pp.193-212. http://dx.doi.org/10.1108/17563780810874717

[13] Shah-Hosseini, H. (2009). "The intelligent water drops algorithm: A nature-inspired swarm-based optimization algorithm" International Journal of Bio-Inspired Computation, 1, 71–79. http://dx.doi.org/10.1504/ijbic.2009.022775

[14] R. Kumar, Y. Lifshits, and A. Tomkins, (2010) "*Evolution of Two-Sided Markets"* Proc. third ACM Int'l Conf. Web Search and Data Mining (WSDM '10), pp. 311-320. http://dx.doi.org/10.1145/1718487.1718526

[15] Malik, Z., & Bouguettaya, A. (2007) "*Evaluating rater credibility for reputation assessment of web services*" In 8'th international conference on web information systems engineering (WISE) .pp. 38–49.
http://dx.doi.org/10.1007/978-3-540-76993-4_4

Received: December 15, 2014; Published: April 20, 2015