ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 4, Issue 2, February 2015*

# Aggregated Similarity Optimization in Ontology Alignment through Multiobjective Particle Swarm Optimization

**Ujjal Marjit**

Centre for Information Resource Management, University of Kalyani, India

**Abstract**: The basic idea behind the ontology is to conceptualize information that is published in electronic format. The problem of ontology alignment is defined as identifying the relationship shared by the set of different entities where each entity belongs to separate ontology. The amount of similarity between two entities from two different ontologies takes part into the ontology alignment process. There are several similarity measuring methods available in the existing literature for measuring the similarity between two discrete entities from different ontologies. To obtain a comprehensive and precise result, all the similarity measures are integrated. One of the ways to combine the various similarity measures is weight-based similarity aggregation. Usually the weights with respect to various similarity measures are assigned manually or through some method. But most of the existing techniques suffer from lack of optimality. Also many evolutionary based approaches are available to find the optimal solution for weight-based similarity aggregation but they are designed as single objective optimization problem. This fact has inspired us to develop a multiobjective particle swarm based optimization algorithm for generating optimal weight based similarity aggregation to get a optimal alignment. In this article, two objectives precision and recall are simultaneously optimized. Moreover a local search is conducted for replacing the worst population in the new generation by best population acquired from the history. The proposed study is evaluated using an artificial data set and performance of the proposed method is compared with that of its single objective versions.

**Keywords**: ontology alignment, particle swarm optimization, multiobjective optimization, f-score.

## I. INTRODUCTION

During the last few years, ontology has gained substantial popularity in the field of computer science. The Greek Philosophers Socrates and Aristotle were the first developing the foundations of ontology. Socrates established the notion of abstract ideas, a hierarchy among them, and class instance relations. Aristotle included logical associations. Computer scientists have borrowed the term ontology for their own requirements. Ontology is a shared understanding of some domain of interest [1]. It defines a set of entities and relations between them in a way that both humans and machines understand. A little updated version of Karlsruhe Ontology Model [2] is defined as follows:

An An Ontology is a tuple O = (C, R, I, $\leq_C$, $\leq_R$), where:

- C is a set of concepts, R is a set of relations, I is a set of Instances
- $\leq_C$ is partial order on *C* called concept taxonomy,
- $\leq_R$ is a partial order on *R* called relational hierarchy, where $r1 \leq_R r2$ iff $domain(r1) \leq_C domain(r2)$ and $range(r1) \leq_C range(r2)$.

Ontologies carry out the information sharing, reuse and integration in modern heterogeneous knowledge based system. Interoperability among the heterogeneous data sources are solved by using ontology alignment. Different ontologies comprised of several set of discrete entities. Identifying correspondences between the entities of the ontologies are very much essential to combine two or more ontologies in a single one. This mechanism is treated as ontology alignment [3][4][5]. Ontologies are provided to the ontology alignment mechanism and alignments are returned accordingly. Ontology Alignment can formally be defined as "An ontology alignment function, *Align*, based on the set *E* of all entities *e€E* and best on the set of possible ontologies *O* is a partial function *Align : E* X *O* X *O→E*" [4]. If we align the ontology in a manual way it will be complicated to implement when the ontology size is too large. Ontology alignment is a major part in the integration of heterogeneous applications. Over the last decade, many evolutionary based approaches have been implemented in [5], [6], [7] to optimize the quality of ontology alignment. But their design format is based on single objective optimization problem [8]. This reality motivated us to develop such an approach where multiple objectives are optimized in parallel. Particle Swarm Optimization (PSO) has been used for optimization purpose which is modeled as multiobjective problem. There are already many evolutionary based techniques available which have been adopted for optimizing the global quality of ontology alignment. But all the previously developed approaches produce only a single solution for ontology alignment because they are designed as singleobjective optimization problem. This fact has motivated us to develop an approach where PSO is modeled as multiobjective optimization problem for achieving more than one better ontology alignment solutions. G. Acampora et al [9] proposes a memetic algorithm to perform an automatic matching process

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 4, Issue 2, February 2015*

capable of computing a sub-optimal alignment between two ontologies. In article [8], J. Bock et al applied discrete particle swarm optimization for ontology alignment. Holistic ontology alignment by population based optimization is depicted in [10]. Existing variety of genetic algorithm based ontology alignments approaches are addressed in [5], [7], [11] and PSO based ontology alignment methods has been introduced in [8]. In this proposed study, PSO has been designed for encoding various weights as particles. The process of PSO is initialized with a population of random particles and the algorithm then searches for optimal solutions by continuously updating generations. In this proposed study, PSO has been designed for encoding various weights as particles. The single objective optimization yields a single best solution but multiobjective optimization (MOO) produces a set of solutions which contains a number of non-dominated solutions, none of which can be further improved on any one objective without degrading it in another [12], [13]. Multiobjective Optimization (MOO) can be defined as follows:

*Optimize* $Z = (f_1(x), f_2(x), f_3(x), \ldots\ldots, f_m(x))$, *where* $x = (x_1, x_2, x_3, \ldots, x_n) \in X$……………………………………….(1)

Here $X$ is $n$-dimensional decision vector solution and $X$ is decision space. The vector objective function $Z$ maps $X$ in $R^m$, where $m \geq 2$ is the number of objectives. Moreover multiobjective optimization problem has been modeled by applying PSO [14] in which fitness comparison takes Pareto dominance [15], [16] into account when moving the particles and non-dominated solution are stored in an archive to approximate the Pareto front. Furthermore, a local search is applied for giving the procedure a better direction by replacing the worst population with best population. The performance of this proposed algorithm has been demonstrated using an example of randomly generated seven similarity measures. Its performance has been compared with the single objective versions.

## I. RELATED WORK

Applying Genetic Algorithm (GA) or Particle Swarm Optimization (PSO) on ontology alignment problem provides several benefits in terms of good accuracy such as precision recall, f-measure for handling ontology alignment for the large ontologies. During the last few years many researchers have given a lot of attention in this field to identify the sensible alignment between the ontologies. Jorge Martinez-Gil et al [7], in their paper discussed how the Genetic Algorithm (GA) can be used to identify the optimal weight configuration for weighted average aggregation of several base matcher in the Goal system. Giovanni Acampora et al [9] propose a memetic algorithm to accomplish an automatic matching process capable of computing a suboptimal alignment between two ontologies. In this regard the authors have modeled ontology alignment problem as a minimum optimization problem where the objective function is based on fuzzy similarity. A memetic approach which is a combination of evolutionary and local search methods can also be used rather than GA for better performance. Jose Manuel Vazquez Naya et al [17] adopted GA based approach to detect how to aggregate different similarity measures into a single metric. But the proposed system only deals with classes in the ontologies rather than properties or instances. In paper [5], Alexandru-Lucian Ginsca et al addresses the growing challenges in the field of ontology alignment. They have formulated the basic similarity measures such as syntactic similarity represented by Levenshtein [18] or Jaro distance [19], semantic similarities which make use of WordNet and taxonomy similarities. In addition to that their developed system uses a genetics algorithm particularly designed for the task of optimizing the aggregation of these measures. Shailendra Singh et al [20] put forward a hybrid approach based on genetics algorithm that determines the best combination of algorithm to map the ontologies. Their method addresses the ontology mapping problem from several perspectives as opposed to a single perspective of ontology.

## II. MULTIOBJECTIVE OPTIMIZATION

Single objective optimization problem optimize a single goal and generate a solution regarding to the single optimizing criterion. But the fact is that in real world, there are different aspects of solutions which are partially or wholly in conflict. Therefore, the MultiObjective Optimization (MOO) is considered to estimate that different aspects of solutions. The multiobjective optimization can formally be stated as [21–23].

Find the vector $\vec{x}^* = [x^*_1, x^*_2, \ldots\ldots, x^*_n]^T$ of decision variables which satisfies $m$ inequality constraints:

$g_i(\vec{x}) \geq 0,\ i = 1; 2, \ldots\ldots, m$ …………………………… (2)

$p$ equality constraints

$h_i(\vec{x}) = 0,\ i = 1; 2, \ldots, p$…………………………………… (3)

and optimizes the vector function

$f(\vec{x}) = [f_1(x), f_2(x), \ldots\ldots, f_k(x)]^T$ ……………………………(4)

The constraints in eqns. (1) and (2) define the feasible region $F$ which contains all the admissible solutions. Any solution outside this region is inadmissible since it violates one or more constraints. The vector $\vec{x}^*$ describes an optimal solution in $F$. In the context of multiobjective optimization, the difficulty lies in the definition of optimality, since it is only rarely that we will find a situation where a single vector $\vec{x}^*$ *represents* the optimum solution to all the objective functions. However the meaning of optimization with respect to Multiobjective Optimization can be defined through Pareto optimality [15], [16]. Pareto optimal set of solutions consists of all those that it is impossible to improve any objective without simultaneous worsening in some other objective. It can be said that a vector of decision variables $\vec{x}^* \in F$ is Pareto optimal if there does not exist another $\vec{x}^*$ such that $fi(\vec{x}^*)$ _ $fi(\vec{x}^*)$ for all $i = 1 \ldots\ldots\ k$ and $f_j(\vec{x}^*) < f_j(\vec{x}^*)$ for at least one $j$ when the problem is minimizing one. Here, $F$ denotes the feasible region of the problem (i.e., where the constraints are satisfied). Pareto optimal set generally contains more than one solution because there exists different 'trade-off' solutions to the problem with

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 4, Issue 2, February 2015*

respect to different objectives. The set of solutions contained by Pareto optimal set are called nondominated solutions. The plot of the objective functions whose non-dominated vectors are in the Pareto optimal set is called the Pareto front. In fact, MOO generates the whole Pareto front [15], [16] or an approximation to it.

**Algorithm-1: Basic PSO**

1: The Swarm is initialized with random position and zero velocity.
2: for $n$ := 1 : Swarm-size do
3: The *fitness* is computed
4: end for
5: for $i$:= 1 : specified number of *iteration* do
6: for $j$ := 1 : Swarm-size do
7: *pbest* is updated
8: *Gbest* is updated
9: *position* and *velocity* are evaluated as new population
10: *fitness* is computed for new population

**11: end for**
12: end for

## III. THE DESIGN OF BASIC PSO ALGORITHM

Among various existing population based optimization techniques [6], [7], Particle Swarm Optimization [24] is a very well known optimization approach. In PSO, candidate solutions are called particles and a population of these particles is called a swarm. A swarm consists of $N$ particles moving around a $D$-dimensional search space. The population in PSO is initialized with random particles and the candidate solutions or particles move around the search space with the goal to acquire optimal fitness. Initially, each particle has a position and velocity and the position and velocity of each particle are updated according to a few formulae. Unlike other optimization techniques which tend to have premature convergence to local optimal solution, PSO is known for globalized searching. Existing variety of PSO based ontology alignment methods has been introduced in [25] and [26]. In this proposed study, PSO has been designed for encoding various weights as particles. The basic model of a PSO technique is described in Algorithm 1.

## IV. PROPOSED METHOD

Computing optimal similarity aggregation is a challenging task as it needs more robust and efficient techniques to acquire the comprehensive and precise alignments. Several similarity measure techniques [27], [28] are combined to a single metric during the process of ontology alignment. Different techniques [5], [6], [7] have already been developed in this regard. In this article, we have proposed PSO in the framework of multiobjective optimization [14]. Moreover non-dominated sorting and crowding distance sorting are applied to improve adaptive fit of the population to a Pareto front and to get better diversity of Pareto optimal front respectively. The proposed method is applied on a artificial data set where rows of the data set represent the different similarity measures and columns represent the associations between two different ontologies. Then for integrating these similarity measures into single metric optimal weights are generated. The proposed approach can find a set of weights correspond to those similarity measures which produce a optimal alignment. During PSO evaluation aggregate function $func_{agg}$ is calculated by multiplying output weight with similarity values as shown in equation 5.

$$func_{agg}(ontology1_i,\ ontology2_j) = \Sigma^7_{k=1}\ w_k \times F_k(smap_{ij}),$$

$$\text{where } \Sigma^7_{k=1}\ w_k = 1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(5)$$

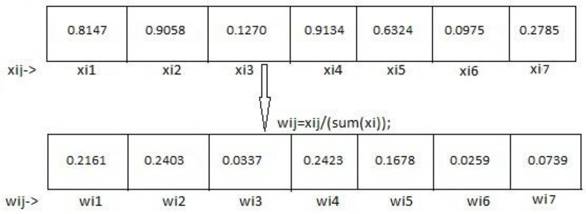


Figure 1: $i$ th particle with seven cells or potions is converted to seven weights using formulae $w_{ij} = x_{ij}/\Sigma^7_{i=1} xi$; so that $0 \leq wij\ \Sigma \leq 1$ and

Therefore, if $funcagg(ontology1i,\ ontology2j)$ is greater than a threshold value then $smap_{ij}$ is a valid mapping. Thus all valid mappings are calculated. Subsequently, using these valid mappings and reference alignments objective functions are calculated. The proposed method has been illustrated by following steps:

### *A) Encoding Scheme and Initializtion*

Here the population is called swarm and it consists of $m$ number of candidate solutions or particles. Each particle has $n$ cells or positions which contain $n$ weights correspond to $n$ various similarity measures considered by the algorithm. As for example, a particle encoding scheme with seven cells or positions which converted into seven weights (normalized cell value) for seven similarity measures is depicted in Fig.1. Initially each cell of a particle are randomly chosen values between 0 and 1. After the initial swarms are chosen, their corresponding fitness values are calculated. Then the velocity of each cell of the particle is initialized to zero. The inputs of the proposed technique are swarm size=50 and weighting factors c1 and c2 which are cognitive and social parameters respectively are set to 2. The threshold value for finding valid mapping is taken 0.5. The algorithm is executed for 30 iterations.

### Objective Function

The approach optimizes multiple objectives i.e., precision and recall are simultaneously optimized. Precision is a measure of correct alignment found from output alignment and recall is a measure of correct alignment found from a given reference alignment. In information retrieval

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 4, Issue 2, February 2015*

positive predictive value is called precision defined in Equation 6 and recall is defined in Equation 7. Using precision and recall, f-measure can be defined as Equation 10.

$$Precision = \frac{|A| - |A \cap R|}{|A|} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (6)$$

$$Recall = \frac{|A| - |A \cap R|}{|A|} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (7)$$

As our proposed multiobjective Particle Swarm Optimization is designed as minimization problem so first objective is computed as (1-precision) and second objective is computed as (1-recall).

*Next Generation Swarm is Produced by Evaluating the Position and Velocity*

Each cell or position represents a weight (normalized cell value) with respect to a similarity measure. The cells within a particle contain values between 0 and 1 and velocity of each gene is initialized to zero. Using the information obtained from the previous step the position of each particle and velocity of each cluster are updated [24] - [14]. Each particle keeps track of the best position it has achieved so far in the history, this best position is also called *pbest* or local best. In multiobjective perspective, that position is chosen for *pbest* for which fitness of that particle dominates other fitnesses acquired by that particle in the history, if there is no such fitness then random choices have been done between current and previous position of that particle. And the best position among all the particles is called global best or *gbest*. Actually whenever a particle moves to a new position with a velocity, its position and velocity are changed according to the equations 8 and 9 given below [24]:

$$v_{ij}(t+1) = w{*}v_{ij}(t) + c1{*}r1{*}(pbest_{ij}(t)x_{ij}(t)) + c2{*}r2{*}(gbest_{ij}(t) - x_{ij}(t)), \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (8)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (9)$$

Here, $t$ is the time stamp and $j$-th cluster of i-th particle has been considered. In equation 8 new velocity $v_{ij}(t+1)$ is acquired using velocity of previous time $v_{ij}(t)$, *pbest* and *gbest*. Then new position $x_{ij}(t+1)$ is obtained by adding new velocity with current position $x_{ij}(t)$ as shown in equation 9. $c1$, $c2$ are set to 2 and $r1$ and $r2$ are two random values in the range of 0 to 1.

**Revising Archieve**

The repository where the non-dominated population in the history has been kept called archive. The current population is merged with the next generation swarm to evaluate the archive. Subsequently, non-dominated solutions have been yielded for next generation. First the archive is initialized with non-dominated population, then next generation population is added, finally again non-dominated sorting and crowded distance sorting is also evaluated for this combined population to obtain better diversity of the Pareto optimal front.

*E. Algorithm for Proposed Multiobjective PSO with Local Search for Ontology Alignment*

In this proposed algorithm, Multi-Objective particle swarm optimization has been modeled to produce optimal weights maximizing the f-measure and minimizing the fall-out. The adopted method technique is illustrated in Algorithm 2. The population is initialized by randomly chosen values between 0 and 1 and population fitness values are calculated using output alignment and reference alignment described in equations 6 and 7. The archive *A* is initialized by the population after non-dominated sorting of the initial population. Velocity and position are updated using equations 8 and 9 respectively. Thereafter, a boundary constraint for each cell is set in the range of 0 to 1. In the algorithm given below, the number of cells is *C* because the number of weights for corresponding similarity measure is *C*. Local best *P* is updated comparing the current fitness and previous fitness of a particle and global best *G* is updated according to random choice of particle from the archive. After applying non-dominated sorting and crowding distance sorting to the archive, a Local Search is conducted for obtaining the better approximation of weights regarding optimal alignment. The Local Search algorithm is described in algorithm 3. In the Local-Search algorithm, the best particle replaces the worst particle of the new generation.

*Algorithm 2: Multi-Objective PSO with Local Search for Ontology Alignment*

Input: Similarity matrix *dt*, *C*=number of cell, *N*= number of particle.

Output: archive *A*

1: $[x_n, v_n, G_n, Pn]^N_{n=1}$ := initialize(*dt*) Random locations between 0 and 1 and velocities

2: $A := ndsort(x_n)$ (if $x_n \not\succ u;\ \forall u \in A$) //Initialize archive A by first non-dominated $x_n$

3: **for** $n := 1 : N$ **do**

4: $w := 1{:}1 - (G_{nd}{=}P_{nd})$

5: **for** $d := 1 : C$ **do**

6: $v_{nd} := w{:}v_{nd} + r1{:}(P_{nd} - x_{nd}) + r2{:}(G_{nd} - x_{nd})$

7: $x_{nd}: = x_{nd} + v_{nd}$ 8: **if** $x_{nd} > 1$ **then** // position set between 0 and 1

9: $x_{nd}: = 1$

10: **else**

11: **if** $x_{nd} < 0$ **then**

12: $x_{nd} := 0$

13: **end if**

14: **end if**

15: **end for**

16: **end for**

17: **for** $n := 1 : N$ **do**

18: $y_n: = f(x_{obj})$ // Evaluate objectives

19: $A := A \cup x_n obj$ // Add $x_n obj$ to $A$

20: $A := ndsort(A)$ if $x_n obj \not\succ u;\ \forall u \in A$ // Non-dominated sorting is applied to the updated archive

21: CrowdingSort(A) // crowding distance sorting for archive

22: **for** $n := 1 : N$ **do**

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 4, Issue 2, February 2015*

23: **if** $x_n$ < *Pn*(*fitnesses*($x_n$) ⊁ *fitnesses*($P_n$)) **then** // Update personal best
24: $P_n := x_n$ 25: **if** Non-dominated fitnesses **then**
26: Random-choice [$x_n$, $P_n$]
27: **else**
28: $G_n$ := *random-select*($x_n$, *A*)
29: **end if**
30: **end if**
31: **end for**
32: *x* = *Local-Search*(*x*,*A*); //Update x by local search
33: **end for**

Algorithm 3 *Local-Search*

Input : Non-dominated Archive *A* New Generation Swarm *x*
Output: Archive *A*
1: Fitness for archive *A fit-A* calculated
2: Fitness for new generation swarm *x fit-x* calculated
3: [*A-max, id*] = *maximum*(*fit-A*); // *id* have the best fitness in *A*
4: [*x-min, idd*] = *minimum*(*fit-x*); // *idd* have the worst fitness among the *x*
5: *x*(*idd*, :) = *A*(*id*, :); // The worst particle in *x* is replaced by best particle from *A*

## V. DATA DESCRIPTION

The proposed algorithm has been applied on a randomly created synthetic dataset. Let us assume two ontologies with the form as depicted in Figure 2. It is evident from the figure that ontology ***a*** has six entities and ontology ***b*** has four entities. Each entity of ontology ***a*** has link with every other entities of ontology ***b***. As there are four links for every entity of ontology *a*, hence a total of twenty four pair wise links are presented by associations. Although only the associations [*a*1, *b*1] and [*a*1, *b*2] are shown in the figure. The associations are given weight by the similarity value computed from the corresponding entities. That means similarity value between *a*1 and *b*1 defines the weight for the association [*a*1, *b*1]. Then we randomly generate a similarity versus association matrix where seven similarity measures and twenty four associations are considered. The data matrix contains values between 0 to 1. It is assumed that an association is a correspondence if the mean of the seven similarity measures regarding the associations exceeds a threshold value 0.8.

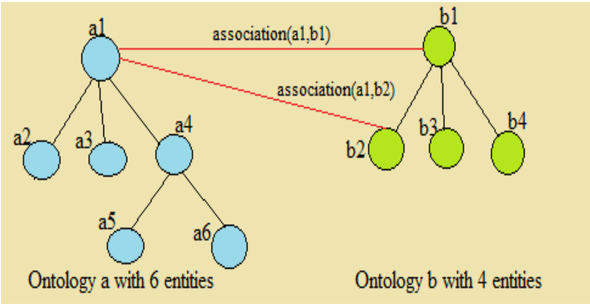


Figure 2: The adapted two ontologies namely *a* with 6 entities and *b* with 4 entities and for example two associations are (*a*1, *b*1) and (*a*1, *b*2) are shown.

## VI. RESULTS AND DISCUSSION

Here, we first describe the performance metrics followed by the results of different algorithms.

### A. *Performance Metrics*

Performance is evaluated using precision, recall, f-measure and fallout. In equations 6 and 7 of section, precision and recall are already described as the measure of correct alignment found from output alignment and the measure of correct alignment found from a given reference alignment respectively. F-measure is a weighted harmonic mean of precision and recall defined in equation 10. Thereafter, fall-out is a measure of incorrect alignment found from the output alignment. Given a reference alignment *R* and some alignment *A*, fall-out can be defined as in equation 11.

$$\text{f-measure} = \frac{2 * Precision \ * Recall}{Precision \ + Recall} \text{……..(10)}$$

$$\text{fall-out} = \frac{|A| - |A \cap R|}{|A|} \text{…………………….….(11)}$$

From the equations 10 and 11, it is clear that high f-measure as well as low fall-out is always giving the best alignment solution. The maximum *F*-score generating candidate solution should have highest precision and highest recall.

### *B. Score Analysis*

Table 1: Scores on Data for Proposed Method and its Single Objective Versions

| **Algorithms** | **Precision** | **Recall** | **F-measure** | **Fall-out** |
|---|---|---|---|---|
| Proposed method | 0.81428 | 1.00 | 0.8976 | 0 |
| Single objective (precision) | 0.7142 | 0.86 | 0.7803 | 0.0031 |
| Single objective (recall) | 0.3333 | 1.00 | 0.5 | 0.0073 |
| Single objective (f-measure) | 0.71286 | 1.00 | 0.8333 | 0 |

In this proposed work, the comparison among the proposed method and its single objective versions are performed with respect to precision, recall, f-measure and fall-out. The proposed method optimized precision and recall simultaneously and results a set of non-dominated solutions stored in archive. Then the highest f-score (which is calculated using precision and recall) generating particle is selected as the final solution. Therefore, precision, recall, f-measure and fall-out are calculated for the final solution and depicted in Table 5.1. The table reveals that the proposed method produces 0.81428 as precision which is better than other single objective versions. The recall value produced by the proposed method is 1 which is equal to Single objective (recall) and Single objective (f-measure) but better than Single

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 4, Issue 2, February 2015*

objective (precision). Again with respect to the f-measure the table shows that our method outperforms other single objective versions. The fall-out for the proposed method is 0 which is less than good for on Single objective (precision) and Single objective (recall). Therefore, the proposed method establishes its efficiency.

## VII. CONCLUSION

In this article, multiobjective Particle Swarm Optimization based approach with a local search is proposed for generating weight vectors correspond to different similarity measures. Then using these weights, different similarity measures are aggregated to improve the ontology alignment problem. Here, an artificial dataset has been used for analyzing the performance of the proposed technique. Therefore, a comparison is carried out among the proposed study and its single objective versions. In near future, we plan to apply this arrangement to the very popular OAEI datasets.

## REFERENCES

1. Uschold, Mike and Michael Gruninger. "Ontologies: Principles, Methods and Applications." The Knowledge Engineering Review, Vol-11, No- 02, pp. 93-136, 1996.
2. Martinez-Gil Jorge, Enrique Alba and José F. Aldana-Montes "Optimizing Ontology Alignments by using Genetic Algorithms." In Proceedings of the Workshop on Nature based Reasoning for the Semantic Web. Karlsruhe, Germany, 2008.
3. Ding Ying and Schubert Foo. "Ontology Research and Development. Part 2-A Review of Ontology Mapping and Evolving." Journal of Information Science, Vol-28, No. 5, pp. 375-388, 2002.
4. Ehrig Marc "Ontology Alignment: Bridging the Semantic Gap", Vol-4, Springer, 2006.
5. Lambrix, Patrick, and He Tan. "Sambo-A System for Aligning and Merging Biomedical Ontologies." Web Semantics: Science, Services and Agents on the World Wide Web 4, No-3 pp.196-206, 2006.
6. Wang, Junli, Zhijun Ding and Changjun Jiang. "GAOM: Genetic Algorithm based Ontology Matching." IEEE Asia-Pacific Conference on Services Computing, APSCC'06, pp. 617-620, 2006.
7. Martinez-Gil, Jorge, Enrique Alba, and José F. Aldana-Montes. "Optimizing Ontology Alignments by using Genetic Algorithms." In Proceedings of the Workshop on Nature based Reasoning for the Semantic Web. Karlsruhe, Germany. 2008.
8. Bock, Jürgen and Jan Hettenhausen "Discrete Particle Swarm Optimization for Ontology Alignment." Journal of Information Sciences Vol-19, No-2 pp. 152-173, 2012.
9. Rahm, Erhard, and Philip A. Bernstein. "A Survey of Approaches to Automatic Schema Matching." the VLDB Journal, Vol-10, No. 4, pp.334-350, 2001.
10. Bock, Jürgen, Sebastian Rudolph, and Michael Mutter "More than the Sum of Its Parts–Holistic Ontology Alignment by Population-Based Optimization." In Foundations of Information and Knowledge Systems, pp. 71-90, Springer Berlin Heidelberg, 2012.
11. Cañas, A. J., J. D. Novak, F. M. González, Francisco Edson, Lopes Rocha, Júlio Valente and Eloi Luiz Favero "A New Approach to Meaningful Learning Assessment Using Concept Maps: Ontologies and Genetic Algorithms." In proceedings of the First International Conference on Concept Mapping, 2004.
12. Coello Coello, Carlos A "Evolutionary Multi-objective Optimization: A Historical View of the Field." IEEE Computational Intelligence Magazine, Vol- 1, pp.28-36, 2006.
13. Deb, Kalyanmoy "Multi-Objective Optimization using Evolutionary Algorithms. Vol. 2012. Chichester: John Wiley and Sons, 2001.
14. Reyes-Sierra, Margarita, and CA Coello Coello. "Multi-objective Particle Swarm Optimizers: A Survey of the State-of-the-Art." International Journal of Computational Intelligence Research, Vol-2, No-3, pp. 287-308, 2006.
15. F Zitzler and L Thiele "An Evolutionary Algorithm for Multi-Objective Optimization: The Strength Pareto Approach." Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology Zürich (ETH), Technical Report 43, 1998.
16. M H Cheok, W. Yang, C H. Pui, J R Downing, C. Cheng, C W Naeve, M V Relling and W.EEvans "Characterization of Pareto Dominance." Operations Research Letters, Vol-31, No-1, pp.7-11, 2003.
17. J. M. V. Naya, M. M. Romero, J. P. Loureiro, C. R. Munteanu, and A. P. Sierra "Improving Ontology Alignment through Genetics Algorithm,," Soft Computing Methods for Practical Environment Solutions: Techniques and Studies, Chapter 15, IGI Global, pp. 241–259, 2010.
18. Euzenat, Jérôme, and Pavel Shvaiko "Ontology Matching" Vol. 18. Heidelberg: Springer, 2007.
19. Maedche Alexander and Steffen Staab "Measuring Similarity between Ontologies." In Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web, pp. 251-263. Springer Berlin Heidelberg, 2002.
20. Singh, Shailendra, and Y. Cheah "A Hybrid Approach for Ontology Mapping via Genetic Algorithm." In Proceedings of Third IEEE International Conference on Computational Intelligence, Modeling and Simulation (CIMSiM-11), pp. 80-83, 2011.
21. Reyes-Sierra, Margarita, and CA Coello Coello. "Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art." International Journal of Computational Intelligence Research, Vol-2, No. 3, pp. 287-308, 2006.
22. Deb Kalyanmoy, Amrit Pratap, Sameer Agarwal and T. A. M. T. Meyarivan "A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II." IEEE Transactions on Evolutionary Computation, Vol-6, No. 2, pp.182-197, 2002.
23. Deb, Kalyanmoy Multi-Objective Optimization using Evolutionary Algorithms. Vol. 2012. Chichester: John Wiley and Sons, 2001.
24. Parsopoulos, Konstantinos E, and Michael N. Vrahatis "Particle Swarm Optimization and Intelligence: Advances and Applications. Hershey: Information Science Reference, 2010.
25. Berners-Lee, Tim, James Hendler, and Ora Lassila. "The Semantic Web."Scientific American 284, no. 5, pp. 28-37, 2001.
26. Gruber, Thomas R "Toward Principles for the Design of Ontologies used for Knowledge Sharing?." International Journal of Human-Computer Studies, Vol-43, No. 5, pp. 907-928, 1995.
27. Acampora, Giovanni, Pasquale Avella, Vincenzo Loia, Saverio Salerno, and Autilia Vitiello. "Improving Ontology Alignment through Memetic Algorithms." In Proceedings of IEEE International Conference on Fuzzy Systems (FUZZ-11), pp. 1783-1790, 2011.
28. Stoilos Giorgos, Giorgos Stamou and Stefanos Kollias. "A String Metric for Ontology Alignment." In Proceedings of International Semantic Web Conference (ISWC2005), pp. 624-637. Springer Berlin Heidelberg, 2005.

## BIOGRAPHY

**Ujjal Marjit** is the System-in-Charge at the C.I.R.M (Centre for Information Resource Management), University of Kalyani. He obtained his M.C.A. degree from Jadavpur University, India. His vast areas of research interest reside in Web Service, Semantic Web, Semantic Web Service, Ontology, Knowledge Management, Linked Data etc. More than 42 papers have been published in the several reputed national and international conferences and journals.


DOI 10.17148/IJARCCE.2015.4257