Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5594553, 9 pages
https://doi.org/10.1155/2021/5594553

Research Article

Multiobjective Sensor Ontology Matching Technique with User Preference Metrics

Hai Zhu,[1] **Xingsi Xue**,[2] **Chengcai Jiang**,[3] **and He Ren**[3]

[1] *School of Network Engineering, Zhoukou Normal University, Zhoukou, Henan 466001, China*
[2] *Intelligent Information Processing Research Center, Fujian University of Technology, Fuzhou, Fujian 350118, China*
[3] *College of Electrical and Power Engineering, Taiyuan University of Technology, Taiyuan 030024, China*

Correspondence should be addressed to Xingsi Xue; jack8375@gmail.com

Received 28 January 2021; Revised 19 February 2021; Accepted 7 March 2021; Published 16 March 2021

Academic Editor: Chi-Hua Chen

Due to the problem of data heterogeneity in the semantic sensor networks, the communications among different sensor network applications are seriously hampered. Although sensor ontology is regarded as the state-of-the-art knowledge model for exchanging sensor information, there also exists the heterogeneity problem between different sensor ontologies. Ontology matching is an effective method to deal with the sensor ontology heterogeneity problem, whose kernel technique is the similarity measure. How to integrate different similarity measures to determine the alignment of high quality for the users with different preferences is a challenging problem. To face this challenge, in our work, a Multiobjective Evolutionary Algorithm (MOEA) is used in determining different nondominated solutions. In particular, the evaluating metric on sensor ontology alignment's quality is proposed, which takes into consideration user's preferences and do not need to use the Reference Alignment (RA) beforehand; an optimization model is constructed to define the sensor ontology matching problem formally, and a selection operator is presented, which can make MOEA uniformly improve the solution's objectives. In the experiment, the benchmark from the Ontology Alignment Evaluation Initiative (OAEI) and the real ontologies of the sensor domain is used to test the performance of our approach, and the experimental results show the validity of our approach.

1. Introduction

Nowadays, sensors are playing a more and more vital role in the distributed systems, and the sensor network has been broadly used in many areas of society, such as military, industrial, smart home, and many other fields [1, 2]. However, the lack of annotations on the sensor information with the semantic meaning is one of the main obstacles to implement the sensor infrastructure [3]. Due to the data heterogeneity issue in the sensor networks, the communications among different sensor network applications are seriously hampered. To integrate the information in different sensor networks, the Semantic Sensor Network (SSN) [4] is proposed, which provides the unified and standard sensor information styles to enhance the semantic connectivity, share sensor data, and improve interoperability. The Semantic Web (SW) provides a knowledge representation of a conceptual reference model, known as an ontology, to restrict domain terms [5, 6], and SSN [7] combines the traditional sensor network with SW's knowledge representation, reasoning, and organizational capabilities. As the key component of SSN, sensor ontology is regarded as the advanced knowledge model for exchanging sensor information.

For the last few years, many sensor ontologies are being progressed, among which the most famous one is SSN ontology [8]. SSN ontology describes the function and properties of sensors, measurement process, and sensor deployment, as well as the observation value of sensors and the method to get the observation value. There have been many kinds of research on sensor data query based on SSN ontology, such as environmental detection systems based on the Internet of Things (IoT) and agricultural ontology systems [9, 10]. SOSA (Sensor, Observation, Sample, Actuator) [11] provides SSN with a lightweight core that aims at expanding the target audience and application areas. In addition, SOSA defines common classes and attributes as the minimum level of interoperability fallback, that is, data for these common classes and attributes can be securely exchanged across all SSN, SSN modules, and SOSA usage. These sensor ontologies are able to describe the sensor's capabilities, performance, and use conditions, which allow different data to be found in different contexts and used for different purposes. However, since different sensor ontologies are built by domain experts with different backgrounds, the same concept might be represented by different names in different domains, or the same name may be depicted by different concepts in different domains, which generates the sensor ontology heterogeneity problem. The sensor ontology heterogeneity problem is the biggest barrier that hampers the interaction and collaboration between smart applications based on sensor ontologies. Sensor ontology matching is an effective way in solving the sensor ontology heterogeneity problem, which aims at confirming the corresponding relationship between entities of different ontologies. When there are thousands of entities in the ontologies, matching these ontologies manually is a long-time and error-prone task. Therefore, many automatic and semiautomatic matching systems have been developed, which utilize the similarity measures to determine the ontology alignment.

The similarity measure is critical to an ontology matching technique, which can be divided into three types of categories, i.e., string distance-based similarity measure, linguistic-based similarity measure, and structure-based similarity measure [12]. A single similarity measure is not able to effectively measure the similarity in various scenarios with different heterogeneity, so aggregating multiple similarity measures is a must to improve the result's confidence [13, 14]. A popular way of aggregating similarity measures is the weighted sum strategy, where different similarity values are integrated to form a comprehensive similarity. The process of determining the suitable aggregating weights is referred to as ontology metamatching.

The ontology metamatching problem as a nonlinear problem often has many locally optimal solutions, and from this point of view, Evolutionary Algorithms (EA) can be used as a good way to address it [15, 16]. GOAL (Genetics for Ontology Alignment) [17] is the first one that uses Genetic Algorithms (GA) to deal with similarity integration problems. Given a Reference Alignment (RA), GOAL determines the optimal parameter set through GA to integrate different similarity measures to obtain the optimal ontology matching results. Because GOAL needs to obtain the reference matching result in advance, so it is difficult to apply it in practical applications. To overcome this drawback, this work proposes user preference-based evaluation metrics on sensor ontology alignment's quality, which is able to work without the RA.

Besides, many EA-based ontology matching techniques focus on a signal optimization objective, which ignores different users' preferences on the solutions. To meet different users' requirements, this paper proposes a multiobjective sensor ontology matching technique, which uses the Multiobjective Evolutionary Algorithm (MOEA) to simultaneously optimize the inflection point solutions [18], i.e., three solutions with the best recall, precision, and f-measure [19]. In particular, the main contributions of this paper are summarized as follows:

(i) The evaluating metrics on sensor ontology alignment's quality is proposed, which takes into consideration user's preferences and do not need to use RA

(ii) A sensor ontology optimization model is constructed to formally define the sensor ontology matching problem

(iii) An MOEA-based multiobjective sensor ontology matching technique is proposed, which uses a selection operator to improve the solution's objectives

The rest of the paper is organized as follows: Section 2 devoted to introducing the basic concepts on the sensor ontology and the evaluation metrics on ontology alignment; Section 3 gives an optimization model and describes how to use MOEA/D to select the user's three preferred solutions; Section 4 presents the experimental results and analysis; finally, Section 5 draws the conclusion and presents the future work.

2. Sensor Ontology Matching Problem and Similarity Measure

2.1. Sensor Ontology Matching Problem. The sensor ontology is defined as a triple $O = (C, P, I)$, where C is the set of classes, P is the set of properties, and I is the set of instances. Typically, class, property, and instance are collectively known as entities. Figure 1 shows the sensor ontology matching process, where O_1 and O_2 are two sensor ontologies to be matched, respectively; A is the standard results of ontology matching, namely, ontology alignment, and it is optional; A $'$ is the ontology aligned by the ontology matching system, which is a set of mapping elements; r is an external resource; and p is a set of parameters, including the weights of the various similarity measures and the threshold used to filter out matching pairs with low confidence. Each element of A' is a 4-tuple (e, e', n, r), where e and e' are entities in different ontologies, respectively, n is the confidence of similarity between two entities, and r is the relationship between two entities, typically the equivalence.

2.2. Similarity Measure

2.2.1. Lexical-Based Similarity Measure. The lexical-based similarity measure, or the string-based similarity measure, determines the semantic similarity between entities by calculating the morphological similarity of the entities. In particular, for sensor ontology modeled by OWL language, the terms that are used to calculate the similarity are as follows: (1) the ID of the entity in the sensor ontology, also known as the local name; (2) the tag of the ontology entity with OWL syntax "rdfs: label"; and (3) the comment of the ontology entity with OWL syntax "rdfs: comment." Typically, the ID, label, and comment of an entity within the sensor ontology are called the textual contents. There are many lexical-based similarity measures, such as N-gram [20], Similarity Measure for Ontology Alignment (SMOA) [21], and Levenshtein [22]. According to [23, 26], N-gram and SMOA have better performance when solving ontology matching

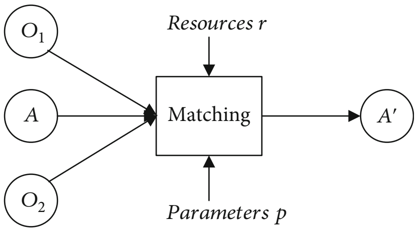

Figure 1: Sensor ontology matching process.

problems; therefore, they were used in this paper, which is, respectively, defined as follows:

$$\text{N-gram}(s_1, s_2) = \frac{2 \times \text{comm}(s_1, s_2)}{N_{s_1} + N_{s_2}}, \tag{1}$$

where s_1 and s_2 are two strings and split into substrings with three characters, $\text{comm}(s_1, s_2)$ is the number of the same substrings between s_1 and s_2, and N_{s_1} and N_{s_2}, respectively, represent the number of the substring in the two strings.

Unlike N-gram, SMOA takes into account the difference and same between the two strings, which is defined as follows:

$$\text{SMOA}(s_1, s_2) = \text{comm}(s_1, s_2) - \text{diff}(s_1, s_2) + \text{winklerImpr}(s_1, s_2), \tag{2}$$

where s_1 and s_2 are the two strings, $\text{comm}(s_1, s_2)$ is a measure of the similarity between s_1 and s_2, $\text{diff}(s_1, s_2)$ is a measure of their difference, and $\text{winklerImpr}(s_1, s_2)$ is an improved approach proposed in [24].

2.2.2. Linguistic-Based Similarity Measure. The linguistic similarity measure is used for determining the semantic distance between two words by considering the synonym or the relation between the hypernym and hyponym. It requires the utilization of external resources, such as WordNet [25]. Unlike other electronic dictionaries, WordNet is an English electronic dictionary based on cognitive linguistics, which not only ranks words but also forms a "web of words" according to their meanings. In WordNet, all words are constructed in hierarchical relationships, with the same depth representing a closer relationship. Conversely, the greater the depth difference, the more alienated the relationship between the two words.

This work uses the Wup similarity measure, which is provided by Wu and Palmer [26] and works based on this principle that the path length is from the closest common ancestor node to the root node. Given two strings s_1 and s_2, their similarity computing method is as follows:

$$\text{Wup}(s_1, s_2) = \frac{2 \times \text{depth}(\text{LCA}(s_1, s_2))}{\text{depth}(s_1) + \text{depth}(s_2)}, \tag{3}$$

where $\text{LCA}(s_1, s_2)$ stands for the closest common parent concept between s_1 and s_2 in WordNet, and $\text{depth}(s)$ is the hierarchy depth of s in the WordNet.

2.2.3. Structure-Based Similarity Measure. Structure-based similarity measure makes use of the context information to calculate two classes' similarity. This work uses the SimRank model [27] to calculate the structure-based similarity value, which is a graph-based topological information model for measuring similarity between two classes. The motivation behind SimRank is that if two classes are referenced by similar classes (that is, they have similar upper and lower nodes), they are also similar, i.e., the similarity is contagious in SimRank. This work uses the matrix form of SimRank [28] which is defined as follows:

$$\mathbf{S} = C \cdot \left(\mathbf{W}^T \cdot \mathbf{S} \cdot \mathbf{W}\right) + (1 - C) \cdot \mathbf{I}. \tag{4}$$

$\mathbf{S}$ is the SimRank similarity matrix whose elements and two variables $\mathbf{S}$ are, respectively, an updated one (on the right-hand side) and an old one (on the right-hand side); $[\mathbf{S}]_{a,b}$ is the distance $s(a, b)$ between note a and note b from two ontologies, respectively; $\mathbf{W}$ is a column-normalized adjacency matrix whose element is $[\mathbf{W}]_{a,b} = 1/|\mathfrak{T}(b)|$ where the mark $\mathfrak{T}(b)$ denotes the set of nodes that point to a node, if there is a directed edge (a, b), otherwise 0; $\mathbf{I}$ is an identity matrix, while C is an attenuation factor, which usually takes 0.8. The similarity value is initialized as follows:

$$s_0(a, b) = \begin{cases} 1, a = b, \\ 0, a \neq b. \end{cases} \tag{5}$$

Equation (4) is used for updating the value of the similarity value in the matrix, and the number of cycles $k = 5$.

A single similarity measure is not able to ensure its effectiveness in all matching tasks, which is due to the complex intrinsic of entity heterogeneous features. To enhance the confidence of the similarity value, it is necessary to aggregate multiple similarity measures. Next, we formally define the sensor ontology metamatching problem, which studies how to determine the optimal aggregating weights for various similarity measures.

3. Sensor Ontology Metamatching Problem

In general, the optimization problem can be divided into the unconstrained optimization problem and constrained optimization problem according to whether there are constrained conditions. The unconstrained optimization problem is referred for solving the optimal objective under the condition of infinite resources, while the constrained optimization problem is to solve the optimal objective under the condition of limited resources. The unconstrained optimization problem as a special circumstance of the constrained optimization problem is short of restricted condition. In this work, the sensor ontology matching problem is regarded as a constrained optimization problem, and the following three points shall be considered before determining the optimization model:

(1) Decision variables, which refer to the undetermined constants and variables related to the constraints

and objective functions involved in the sensor ontology metamatching optimization problems

(2) The objective function, which refers to the function related to a variable and for which the extreme value (maximum or minimum) is to be found

(3) Constraint conditions, which refer to the conditions that variables must meet when finding the extreme value of an objective function

3.1. Decision Variable. The process of ontology metamatching is displayed in Figure 2, where $\mathbf{M}_1$, $\mathbf{M}_2$, $\mathbf{M}_3$, and $\mathbf{M}_4$ are the similarity matrices corresponding to four different similarity measures, respectively, and w_1, w_2, w_3, and w_4 are, respectively, their aggregating weights. The final similarity matrix $\mathbf{M}$ can be obtained by integrating different similarity matrices and filtered by threshold v. In this work, we need to optimize four aggregating weights and one threshold, which consists of the decision variables.

3.2. Objective Function. The traditional evaluating metrics on alignment's quality are recall, precision, and f-measure, respectively, [17], which are defined as follows:

$$\text{recall}(A, R) = \frac{|R \cap A|}{|R|}, \tag{6}$$

$$\text{precision}(A, R) = \frac{|R \cap A|}{|A|}, \tag{7}$$

$$f\text{-measure}(A, R) = \frac{\text{recall}(A, R) \times \text{precision}(A, R)}{\alpha \times \text{recall}(A, R) + (1-\alpha) \times \text{precision}(A, R)}, \tag{8}$$

where A is the RA which contains a set of standard entity pairs and R is the matching results we found which contains a set of founded matching pairs. The recall is used to test the completeness of alignment, and precision of the correct rate of found pairs. To get a more practical solution, recall and precision are integrated to form the f-measure metric. The α in Formula (8) is a preference coefficient, and when α is close to 1.0, the f-measure is closer to recall, and vice versa to precision. However, it is unrealistic to obtain the standard matching results in advance, especially when the ontology boasts a huge amount of data.

In this work, a group of evaluating metrics is proposed on the alignment's quality without using RA, which is taken as the optimization objective. There are three user preferences on the alignment, i.e., completeness, accuracy, and unbiasedness [29], which are defined, respectively, as follows:

$$\text{completeness}(\mathbf{M}) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\Gamma\left(\left[\mathbf{M}_{ij}\right]\right)}{\min\,(m, n)}, \tag{9}$$

$$\Gamma\left(\left[\mathbf{M}_{ij}\right]\right) = \begin{cases} 1, & \text{when element} - \boldsymbol{M}_{ij} \text{ are largest in } i\text{th line and } j\text{th column in } \mathbf{M} \text{ matrix,} \\ 0, & \text{otherwise,} \end{cases} \tag{10}$$

where $\mathbf{M}$ is the similarity matrix after aggregation and $[\mathbf{M}_{ij}]$ is the similarity of the ith row and the jth column from the $\mathbf{M}$.

$$\text{accuracy}(\mathbf{M}) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\varphi\left(\left[\mathbf{M}_{ij}\right]\right)}{\min\,(m, n)}, \tag{11}$$

$$\varphi\left(\left[\mathbf{M}_{ij}\right]\right) = \begin{cases} \left[\mathbf{M}_{ij}\right], & \text{when element} - \mathbf{M}_{ij} \text{ are largest in } i\text{th line and } j\text{th column in } \mathbf{M} \text{ matrix,} \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

After the synthesis matrix $\mathbf{M}$ is obtained by integrating various similarity measurement methods, if the accuracy value of the matrix $\mathbf{M}$ is higher, it can indirectly reflect the higher matching accuracy.

$$\text{unbiasedness}(\mathbf{M}) = 2 \times \frac{\text{completeness} \times \text{accuracy}}{\text{completeness} + \text{accuracy}}. \tag{13}$$

Finally, the objective to be optimized are defined as follows:

$$\max\ F(\mathbf{M}) = (\text{completeness}(\mathbf{M}), \text{accuracy}(\mathbf{M}), \text{unbiasedness}(\mathbf{M}))^T. \tag{14}$$

3.3. Constraint Condition. In this work, there are two constraints, which are on the aggregating weights and the threshold, respectively. According to the weighted sum strategy, the aggregating weight w_i corresponding to the ith similarity measure should comply with the constraint $w_i \in [0, 1]$, $\sum_i w_i = 1$. When the threshold's upper limit is higher than 0.8 or the lower limit is smaller than 0.1, it is not easy to ensure the alignment's quality. Thus, threshold v's range should be $[0.1, 0.8]$.

4. Interactive Multiobjective Sensor Ontology Matching Technique

4.1. Multiple-Objective Evolutionary Algorithm with Decomposition. Multiple-Objective Evolutionary Algorithm with Decomposition (MOEA/D) [30] as a popular MOEA algorithm decomposes the multiobjective optimization problem into a variety of single-objective optimization subproblems and optimizes them at the same time. Because of the decomposition operation, MOEA/D has a great advantage in keeping the distribution of the solution. However, when moving the solutions to the Pareto Front (PF), it might sacrifice some objectives. To uniformly improve the solution's objectives, this work proposes a selection operator, which prevents excessive deterioration of the final solution. The flowchart of MOEA/D is shown in Figure 3.

4.2. Objective Decomposition. In this work, there are two objectives that need to optimize, i.e., maximize the completeness and accuracy of the alignment. To this end, we define the number of the subproblems as N, and their distributed weight vectors are $(\alpha_i, \beta_j)^T = ((i-1)/(N-1), (N-i)/(N-1))^T$, $i = 2, 3, \cdots, N-1$. If the two weight vectors $\lambda_i = (\alpha_i, \beta_i)^T$ and $\lambda_j = (\alpha_j, \beta_j)^T$ are close to each other which depends on

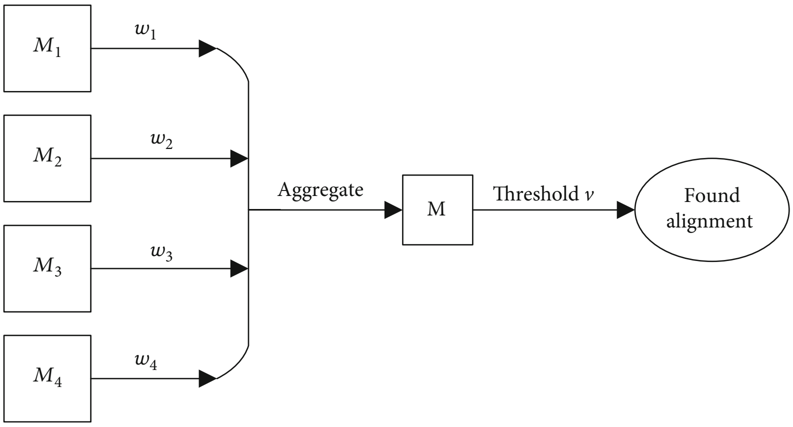

Figure 2: Similarity matrix aggregation.

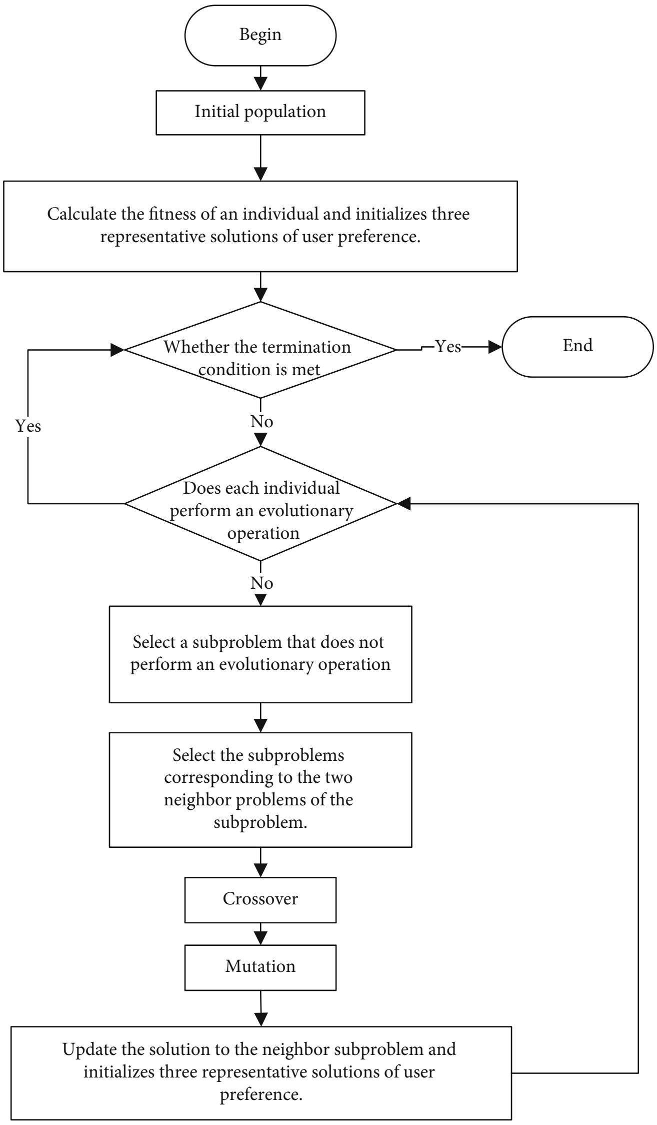
Begin
Initial population
Calculate the fitness of an individual and initializes three representative solutions of user preference.
Whether the termination condition is met
Yes
End
No
Yes
Does each individual perform an evolutionary operation
No
Select a subproblem that does not perform an evolutionary operation
Select the subproblems corresponding to the two neighbor problems of the subproblem.
Crossover
Mutation
Update the solution to the neighbor subproblem and initializes three representative solutions of user preference.

Figure 3: The flowchart of MOEA/D.

Euclidian distance, ith and jth subproblems are called neighbor problems, which can help each other to achieve optimization during the evolving process. The weighted integration strategy adopted in this paper is the Tchebycheff approach [28], which is able to convert the approximation problem of Pareto Front into several scalar optimization problems:

$$\begin{cases} \min \ g^{te}(x \mid \lambda, z^*) = \max \{\lambda_i |f_i(x) - z_i^*|\} \\ \text{s.t.} x \in \Omega, \end{cases} \tag{15}$$

where $z^* = (z_1^*, z_2^*)^T$ is the reference point, and we set $z^* = (1.0,1.0)^T$; $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)^T$ is a weight vector, and $\sum_{i=1}^{2} \lambda_i = 1$; Ω is the feasible solution space. Equation (15) is the criterion for updating the neighbor solution, i.e., supposing the newly generated solution is y and the original solution is x, if $g^{te}(y \mid \lambda, z^*) \leq g^{te}(x \mid \lambda, z^*)$, x should be replaced by y.

4.3. Encoding Mechanism. This paper adopts the binary coding mechanism, in which a chromosome is divided into two parts. The first one is the combined weights of various similarity measures, respectively, and the second part is the threshold. The decoding process is shown in Figure 4. Let p be the number of similarity measures, the group of segmentation points can be expressed as $c' = \{c_1', c_2', \cdots, c_{p-1}'\}$. When decoding, firstly, the elements in c' are arranged in ascending order in this form $c' = \{c_1, c_2, \cdots c_{p-1}\}$, and then, the weight is calculated according to Equation (16). In particular, the second part does not need to execute the special decoding process.

$$w_k \begin{cases} c_1, & k = 1, \\ c_k - c_{k-1}, & 1 < k < p, \\ 1 - c_{p-1}, & \text{other}. \end{cases} \tag{16}$$

4.4. Genetic Operator. EA's selection operator aims at selecting individuals with high fitness and removing individuals with low fitness. According to MOEA/D, the selection operator works basing on the neighborhood of an individual in the population. First, the completeness and accuracy values of the current individual's neighbor individuals are obtained; then, the distance (d_i) of the solution of the neighbor individual from the origin point (see also Figure 5) is calculated; and finally, two neighbor individuals with a relatively long distance are selected for cross operation. In this work, the selection operator dedicates to improve the solution's quality evenly in terms of two objectives. The new selection strategy will be described in more detail below.

Because the user preference information is considered in this paper, it is necessary to select a better trade-off solution as an alternative and then perform crossover and mutation operations on such an alternative. In terms of the trade-off between solutions, uniform weight and solution are combined into an objective function. Firstly, the corresponding solutions and their neighbor solutions are selected, and then, the scores of the three solutions under the uniform weights are calculated. The calculation of the score is as follows: If there is a uniform weight vector $[h, 1-h]^T$, the score corresponding to the three solutions is score $= [h, 1-h]^T \times$ [completeness, accuracy]. Through the above method, the scores of each solution under a uniform weight can be calculated, and the two solutions with high scores are selected as the candidate solutions for the crossing and mutation operator.

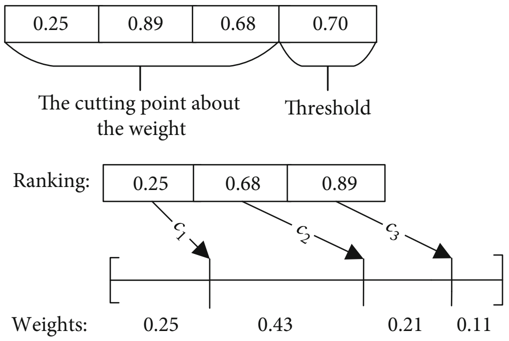

Figure 4: Decoding operation.

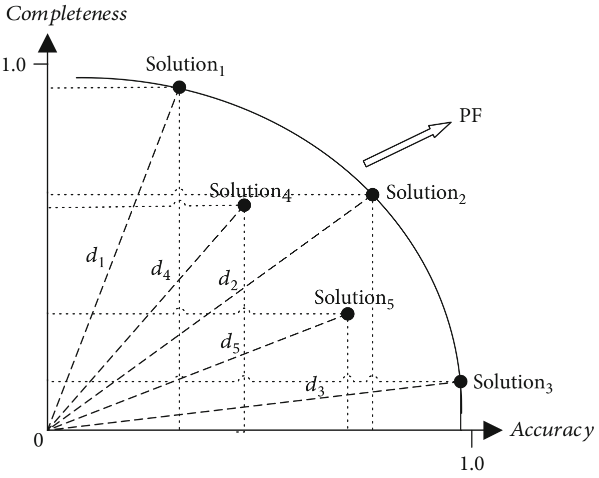

Figure 5: Process of selecting solutions.

The single point crossover and locus mutation were used as crossover and mutation strategy, respectively.

5. Experiment

5.1. Experiment Configuration. In this work, the testing cases we used consist of Ontology Alignment Evaluation Initiative (OAEI)'s (http://oaei.ontologymatching.org) benchmark and three real sensor ontologies, to test the effectiveness of our approach, whose brief descriptions are shown in Table 1.

The comparison systems in the experiment are from OAEI's participants and can be found on OAEI's official website, while the results of our approach are the mean value of 30 independent runs. The configuration of our approach is as follows:

(i) Similarity measures: N-gram, SMOA, WuAndPamer, and SimRank

Table 1: The brief description of testing cases.

ID	Brief description
101-104	The target ontology is identical to the source or different with some OWL constraints.
201-210	The linguistic features are different between the target ontology and the source ontology, but their structures are the same.
221-247	The conceptual structure is different between the target ontology and the source ontology, but their linguistic features are the same.
248-266	The target ontology and source ontology have different lexical, linguistic, and structural features.
301-304	Both the target ontology and the source ontology are derived from the ontology used in reality.
Old SSN	The descriptions of sensors, data, and concepts.
New SSN	There is some improvement on the old SSN ontology, making it more applicable.
SOSA	The lightweight core of SSN.

Table 2: Comparison in terms of precision.

	edna	AML	LogMap	LogMapLt	XMap	LogMapBio	IOMap	CroLOM	Our approach
101	0.64	1	0.94	0.56	0.94	0.5	1	1	1
201-202	0.259	1	0.962	0.307	0.959	0.343	1	1	0.994
221-247	0.754	1	0.889	0.6	0.97	0.518	1	1	1
248-266	0.285	1	0.959	0.348	0.975	0.256	1	1	0.898
Average	0.336	1	0.947	0.389	0.972	0.312	1	1	0.927

Table 3: Comparison in terms of f-measure.

	edna	AML	LogMap	LogMapLt	XMap	LogMapBio	IOMap	CroLOM	Our approach
101	0.78	0	0.95	0.71	0.97	0.62	0	0	1
201-202	0.315	0.231	0.474	0.345	0.391	0.267	0	0	0.745
221-247	0.831	0.754	0.814	0.726	0.961	0.598	0	0	1
248-266	0.318	0.36	0.436	0.338	0.413	0.202	0	0	0.556
Average	0.41	0.41	0.51	0.409	0.51	0.28	0	0	0.656

(ii) Number of subproblems: 3

(iii) Population scale: 40

(iv) Crossover probability: 0.5

(v) Mutation probability: 0.01

We use the above configuration empirically, which ensures our approach achieves the highest average results in all testing cases.

5.2. Results and Analysis

5.2.1. Testing on OAEI's Benchmark. The result we compared is from OAEI's participants in Tables 2–4, where the testing cases are divided into four parts, i.e., 101, 201-202, 221-247, and 248-266, according to the heterogeneity features of the ontologies. In Table 5, it is obvious that both edna and LogMapLt perform well in the testing cases 101 and 221 to 247, and our method's results are good as well. In the testing cases 201 to 202, none of the matching systems outperform others while our method's results are close to 0.7, which shows that our method is effective than other competitors in terms of recall. The testing cases in 201-210 are heterogeneous in terms of both lexical and linguistic. All the testing cases own different heterogeneous characteristics, which require cooperation among different similarity measures. As to why the IOMap and CroLOM systems have achieved such poor results, our analysis is as follows. The IOMap is a system to solve the cross-lingual problem. Similarly, the full name of the CroLOM system is the Cross-Lingual Ontology Matching System and it uses Yandex translators, NLP techniques, and similarity methods based on word and synonym categories. Both systems have achieved excellent results in dealing with cross-language ontology, but they cannot achieve satisfactory

Table 4: Real sensor ontology alignment.

	Recall	Precision	f-measure
SSN (old)-SSN (new)	0.97	1.0	0.98
SSN (old)-SOSA	0.97	1.0	0.98
SSN (new)-SOSA	1.0	1.0	1.0

Table 5: Comparison in terms of recall.

	edna	AML	LogMap	LogMapLt	XMap	LogMapBio	IOMap	CroLOM	Our approach
101	1	0	0.96	0.99	1	0.56	0	0	1
201-202	0.404	0.139	0.372	0.396	0.291	0.228	0	0	0.683
221-247	1	0.669	0.858	0.995	0.959	0.71	0	0	1
248-266	0.395	0.264	0.323	0.387	0.314	0.181	0	0	0.461
Average	0.505	0.317	0.426	0.498	0.429	0.28	0	0	0.582

results in the benchmark test set because the major of IOMap and CroLOM is cross-lingual. From the experimental results, we can see that our approach is able to better aggregate different similarity measures to determine more correct correspondences. This argument is further supported by the average recall.

In Table 2, we compare our approach in terms of precision. It can be seen that AML, IOMap, and CroLOM are able to determine the alignments with precision value 1.0. However, they achieve this goal by sacrificing recall. There is a contradiction between recall and precision, and when a matching technique tries to optimize one metric, it might deteriorate the other one. Since our approach is able to trade-off recall and precision, and although our approach's precision values are not the best, they are in general high. In particular, our approach's mean precision is only a bit lower than other competitors.

Table 3 presents all competitor's f-measure values. As can be seen from the table, IOMap and CroLOM, which have high precision and low recall, own low f-measure, and so do AML. Our approach is far ahead of other matching systems, which has the best results on the testing cases 101, 201 to 202, 221 to 247, and 248 to 266. In particular, our approach's results in the testing cases 101 and 221 to 247 are almost perfect, and even in the highly heterogeneous groups (201 to 202 and 248-266), our approach's f-measure values are still high. In contrast, none of the other matching systems' f -measure values are higher than 0.5. It can be seen from the average value that the alignments obtained by our method are much better than other matching systems. The reason behind this is that the suitable selection operator makes the nondeteriorating solution filtered out in the evolution process, which improves the result's quality. Furthermore, the solution selection strategy of inflection point based on user preference makes our approach be able to better trade-off the solution's recall and precision.

5.2.2. Testing on Real Sensor Ontologies. Three real sensor ontologies used in this work are, respectively, SSN (both new and old versions) and SOSA. SSN ontology is used to describe the processes and characteristics of the sensors and their observations, which follows a horizontal and vertical modular architecture. SSN and SOSA vary in scope and degree of the axiom and support a variety of applications, such as satellite imagery, large-scale scientific monitoring, industrial, smart-home infrastructure, social perception, citizen science, observation-driven ontology engineering, and the Internet of Things (IoT). The new SSN is different from the original SSN; the new SSN simplifies the relationship between device, platform, and system classes on the old SSN. New SSN supports humans and other animals as agents better permitting all major classes to be virtual.

Three pairs of real sensor ontology alignment are shown in Table 4. It can be seen that our approach can determine the perfect alignment when matching the new SSN (new) with SOSA. Since our method can only find the one-to-one matching relationship between the two ontologies, its recall and precision are affected on the other matching tasks. In general, our approach can determine high-quality sensor ontology alignments.

6. Conclusion and Future Work

Sensor ontology matching as a challenging problem needs to be solved, a new optimization model is constructed, and a multiobjective optimization algorithm based on user preference is used to find the appropriate solution. Our method is aimed at simultaneously determining different alignments for the users with different preferences. In order to test the performance of our approach, the benchmark provided by OAEI and three pairs of sensor ontologies are used. The experimental results show that our method is effective.

In tomorrow's work, we are interested in further lifting the performance of the algorithm by getting the expert involved. Besides, our method is not scalable, whose performance significantly drops with the growing ontology scale. A feasible solution would be the introduction of some divide-and-conquer strategy to improve the scalability of our approach.

Data Availability

The data used to support this study can be found in http://oaei.ontologymatching.org.

Conflicts of Interest

The authors declare that they have no conflicts of interest in the work.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Nos. 61801527 and 61103143) and the Natural Science Foundation of Fujian Province (No. 2020J01875).

References

[1] A. Bröring, J. Echterhoff, S. Jirka et al., "New generation sensor web enablement," *Sensors*, vol. 11, no. 3, pp. 2652–2699, 2011.

[2] D. J. Russomanno, C. R. Kothari, and O. A. Thomas, "Building a sensor ontology: a practical approach leveraging ISO and OGC models," in *Proceedings of the 2005 International Conference on Artificial Intelligence*, pp. 637–643, Las Vegas, Nevada, USA, 2005.

[3] A. Bröring, K. Janowicz, C. Stasch, and W. Kuhn, "Semantic challenges for sensor plug and play," in *Web and Wireless Geographical Information Systems. W2GIS 2009*, pp. 72–86, Springer, 2009.

[4] A. Sheth, C. Henson, and S. S. Sahoo, "Semantic sensor web," *IEEE Internet Computing*, vol. 12, no. 4, pp. 78–83, 2008.

[5] C. W. B. Smith, "Formal ontology and information systems," *Telemedicine & Virtual Reality*, vol. 3, no. 5, pp. 309–321, 1993.

[6] W. Kuhn, "Semantic engineering," in *Research Trends in Geographic Information Science*, Springer, Berlin Heidelberg, 2009.

[7] X. Xue, "A compact firefly algorithm for matching biomedical ontologies," *Knowledge and Information Systems*, vol. 62, no. 7, pp. 2855–2871, 2020.

[8] M. Compton, "What now and where next for the W3C semantic sensor networks incubator group sensor ontology," in *International Conference on Semantic Sensor Networks*, pp. 1–8, Bonn, Germany, 2011.

[9] F. Lécué, S. Tallevi-Diotallevi, J. Hayes et al., "Smart traffic analytics in the semantic web with STAR-CITY: scenarios, system and lessons learned in Dublin City," *Journal of Web Semantics*, vol. 27-28, pp. 26–33, 2014.

[10] A. Khamparia and B. Pandey, "Performance analysis on agriculture ontology using SPARQL query system," in *2014 International Conference on Data Mining and Intelligent Computing (ICDMIC)*, pp. 1–5, Delhi, India, 2014.

[11] K. Janowicz, A. Haller, S. J. D. Cox, D. Le Phuoc, and M. Lefrançois, "SOSA: a lightweight ontology for sensors, observations, samples, and actuators," *Journal of Web Semantics*, vol. 56, pp. 1–10, 2019.

[12] J. David, F. Guillet, and H. Briand, "Matching directories and OWL ontologies with AROMA," in *Proceedings of the 15th ACM international conference on Information and knowledge management - CIKM '06*, pp. 830-831, Arlington Virginia, USA, 2006.

[13] X. Xue and J. Chen, "Optimizing ontology alignment through hybrid population-based incremental learning algorithm," *Memetic Computing*, vol. 11, no. 2, pp. 209–217, 2019.

[14] X. Xue, C. Yang, C. Jiang, P. W. Tsai, G. Mao, and H. Zhu, "Optimizing ontology alignment through linkage learning on entity correspondences," *Complexity*, vol. 2021, 12 pages, 2021.

[15] X. Xue and Y. Wang, "Optimizing ontology alignments through a memetic algorithm using both MatchFmeasure and unanimous improvement ratio," *Artificial Intelligence*, vol. 223, pp. 65–81, 2015.

[16] X. Xue, X. Wu, C. Jiang, G. Mao, and H. Zhu, "Integrating sensor ontologies with global and local alignment extractions," *Wireless Communications and Mobile Computing*, vol. 2021, 10 pages, 2021.

[17] M. G. Jorge, E. Alba, and J. F. Aldana-Montes, "Optimizing ontology alignments by using genetic algorithms," in *Proceedings of the First International Workshop on Nature Inspired Reasoning for the Semantic Web*, pp. 1–15, Karlsruhe, Germany, 2008.

[18] S. Bechikh, L. B. Said, and K. Ghédira, "Searching for knee regions of the Pareto front using mobile reference points," *Soft Computing*, vol. 15, no. 9, pp. 1807–1823, 2011.

[19] V. Rijsberge, *Information Retrieval*, University of Glasgow, 1975.

[20] V. Mascardi, A. Locoro, and P. Rosso, "Automatic ontology matching via upper ontologies: a systematic evaluation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 5, pp. 609–623, 2010.

[21] X. Xue and J. S. Pan, "An overview on evolutionary algorithm based ontology matching," *Journal of Information Hiding and Multimedia Signal Processisng*, vol. 9, no. 1, pp. 75–88, 2018.

[22] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Soviet Physics Doklady*, vol. 1965, no. 10, pp. 707–710, 1965.

[23] G. Stoilos, G. Stamou, and S. Kollias, "A string metric for ontology alignment," in *Proceedings of 4th Int. Semantic Web Conference*, pp. 623–637, Karlsruhe, Germany, 2005.

[24] W. E. Winkler, "The state of record linkage and current research problems," Tech. Rep. US Census Bureau, Suitland-Silver Hill, MD, USA, 1999.

[25] G. A. Miller, "WordNet," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[26] Y. Wang, "Verb semantics and lexical selection," *Computer Science*, vol. 14, no. 101, pp. 325–327, 1994.

[27] J. Glen and J. Widom, "SimRank: a measure of structuralcontext similarity," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 538–543, Edmonton, Canada, 2002.

[28] C. Li, J. Han, G. He et al., "Fast computation of simrank for static and dynamic information networks," in *Proceedings of the 13th International Conference on Extending Database Technology - EDBT '10*, pp. 465–476, Lausanne, Switzerland, 2010.

[29] X. Xue, J. Lu, and J. Chen, "Using NSGA-III for optimizing biomedical ontology alignment," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 3, pp. 135–141, 2019.

[30] Q. Zhang and H. Li, "MOEA/D: a multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2008.