Optimizing Biomedical Ontology Alignment through a Compact Multiobjective Particle Swarm Optimization Algorithm Driven by Knee Solution

Xingsi Xue 1,2,3,4, Xiaojing Wu 5, and Junfeng Chen 6

1 Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, Fuzhou 350118, China
2 Intelligent Information Processing Research Center, Fujian University of Technology, Fuzhou 350118, China
3 Fujian Key Lab for Automotive Electronics and Electric Drive, Fujian University of Technology, Fuzhou 350118, China
4 Guangxi Key Laboratory of Automatic Detecting Technology and Instruments, Guilin University of Electronic Technology, Guilin, Guangxi 541004, China
5 College of Information Science and Engineering, Fujian University of Technology, Fuzhou 350118, China
6 College of IOT Engineering, Hohai University, Changzhou 213022, China

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

Received 19 January 2020; Revised 15 March 2020; Accepted 19 March 2020; Published 1 May 2020

Guest Editor: Chi-Hua Chen

Copyright © 2020 Xingsi Xue et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nowadays, most real-world decision problems consist of two or more incommensurable or conflicting objectives to be optimized simultaneously, so-called multiobjective optimization problems (MOPs). Usually, a decision maker (DM) prefers only a single optimum solution in the Pareto front (PF), and the PF’s knee solution is logically the one if there are no user-specific or problem-specific preferences. In this context, the biomedical ontology matching problem in the Semantic Web (SW) domain is investigated. Although biomedical ontology is extensively used in the biomedical domain to model the biomedical knowledge, it is difficult to find a perfect alignment that could meet all DM’s requirements, and usually, the matching process needs to trade-off two conflict objectives, i.e., the alignment’s recall and precision. To this end, in this work, the biomedical ontology matching problem is first defined as a MOP, and then a compact multiobjective particle swarm optimization algorithm driven by knee solution (CMPSO-K) is proposed to address it. In particular, a compact evolutionary mechanism is proposed to efficiently optimize the alignment’s quality, and a max-min approach is used to determine the PF’s knee solution. In the experiment, three biomedical tracks provided by Ontology Alignment Evaluation Initiative (OAEI) are used to test CMPSO-K’s performance. The comparisons with OAEI’s participants and PSO-based matching technique show that CMPSO-K is both effective and efficient.

1. Introduction

Decision-making requires finding an optimum solution to a decision problem in the process of identifying and evaluating alternatives. Nowadays, most real-world decision problems consist of two or more incommensurable objectives to be optimized simultaneously, so-called multiobjective optimization problems (MOPs) [1]. In general, MOP does not have one optimum solution but a set of solutions, so-called Pareto optimal solutions, which are superior to the others in terms of one or more objectives. In many situations, a decision maker (DM) only prefers one single optimum solution in the Pareto front (PF) [2], and therefore, the optimization and the decision process are often combined. One of the multiobjective methods is driven by the PF’s knee solution which is the one for which an improvement in one of its objectives will result in a deterioration in another. Often, PF’s knee solution is logically preferred to the DM if there are no user-specific or problem-specific preferences [3].

In this context, the biomedical ontology matching problem in the Semantic Web (SW) domain is investigated. Although biomedical ontology is extensively used in the biomedical domain to model the biomedical knowledge, it is
developed and maintained by different communities, and the same biomedical knowledge could be described with different terminologies or in different contexts. To bridge the semantic gaps between two biomedical ontologies and support their communications, it is necessary to find their identical concept mapping, which is the so-called biomedical ontology matching. Since biomedical ontologies often own large-scale concepts with rich semantic meanings, it is difficult to find a perfect alignment that could meet all DM’s requirements, and usually, the matching process needs to trade-off two conflict objectives, i.e., the alignment’s recall and precision. To this end, in this work, the biomedical ontology matching problem is defined as a MOP. Although there are many studies on the single-objective approaches for addressing the ontology matching problem [4–8], the research on multiobjective ontology matching techniques is still in this infancy [9]. Being inspired by the success of the particle swarm optimization algorithm (PSO) in the ontology matching domain [10], large-scale discrete optimization domain [11–13], and biomedical engineering [14], in this work, a compact multiobjective particle swarm optimization algorithm driven by knee solution (CMPSO-K) is further proposed to address this problem. In particular, CMPSO-K uses a compact multiobjective evolutionary mechanism to efficiently optimize the alignment’s quality and a max-min approach to determine the PF’s knee solution. The contributions made in this paper are as follows: (1) a discrete multiobjective optimal model for the biomedical ontology matching problem is constructed; (2) a hybrid biomedical concept similarity metric is proposed, which can effectively calculate the similarity value of two biomedical ontology concepts; and (3) a CMPSO-K is proposed to optimize the biomedical ontology alignment, which takes into consideration both DM’s preference and algorithm’s performance.

The rest of the paper is organized as follows: Section 2 overviews the SIA-based ontology matching techniques; Section 3 defines the biomedical ontology matching problem and presents the similarity measure on biomedical concepts; Section 4 presents the CMPSO-K-based biomedical ontology matching technique in detail; Section 5 shows the experimental results; and finally, Section 6 draws the conclusion and presents the future work.

2. Swarm Intelligence Algorithm-Based Ontology Matching Technique

The first generation of swarm intelligence algorithm (SIA-) based matcher dedicates to optimize the ways of aggregating various ontology matchers’ corresponding alignments. The very first matching system is Genetics for Ontology Alignments (GOAL) [7], which uses evolutionary algorithm (EA) to optimize the aggregating weight set of different ontology matchers. Later, Alexandru-Lucian and Iftene [15] further used EA to optimize one more parameter to filter the unavailable concept mappings to final alignment. Acampora et al. [4] introduced a local search process into EA’s evolving process to improve the algorithm’s performance. Xue and Wang [16] used a new metric as the fitness function to guide the algorithm’s search direction. Their approach can address the holistic matching problem and determine a universal weight configuration for matching several pairs of ontologies at a time. He et al. [6] proposed to utilize artificial bee colony algorithm (ABC) to optimize all the parameters in the matching process, whose results are better than the EA-based matchers. More recently, Xue et al. [17] proposed a new approach that uses NSGA-III [18] to combine various similarity measures without tuning the aggregating parameters. However, when the scale of the similarity measures becomes huge, e.g., more than 50 similarity measures, this approach could be inefficient.

The above matchers need to calculate all the matchers’ alignments and store them in the main memory before aggregating them, which requires huge memory consumption. Recently, the second generation of the SIA-based matcher tries to directly find an entity correspondence set that is close to the golden alignment. GAOM (genetic algorithm-based ontology matching) [8] regards two ontologies as two discrete concept sets and employs EA to determine the optimal entity mapping set. Alves et al. utilized a memetic algorithm (MA), which combines EA with a local search strategy, to execute the instance-based ontology matching process [5]. They first matched the instances and then propagated the instance pair’s similarity value to the corresponding concepts. More recently, MapPSO [10] uses PSO to determine the optimal entity correspondence set. In particular, MapPSO introduces a new quality measure on the ontology alignment, which depends on the statistical results on the alignment. More recently, Chu et al. [19] first modeled the ontologies in the vector space so that two entities’ similarity value can be calculated through the cosine function, and then EA was used to determine the optimal alignments. Xue [20] proposed a new similarity metric for measuring the biomedical concepts’ similarity value and then used the firefly algorithm (FA) to optimize the biomedical ontology alignment. However, their proposal suffers from the premature convergence when matching large-scale ontologies. Our proposal also belongs to this category, but different from the existing work, in this work, the biomedical ontology problem is regarded as a discrete MOP, and a multiobjective SIA is presented to address it, which takes into consideration the DM’s preference.

3. Preliminaries

3.1. Biomedical Ontology Matching Problem. Ontology matching aims at determining the identical entity mappings, which is the so-called ontology alignment. In the past, the ontology alignment’s quality was often measured by f-measure [21], which is defined as follows:

\[
\text{recall} = \frac{|R \cap A|}{|R|} \\
\text{precision} = \frac{|R \cap A|}{|A|} \\
f - \text{measure} = \frac{\text{recall} \times \text{precision}}{a \times \text{recall} + (1 - a) \times \text{precision}}
\]
where \( R \) and \( A \) are, respectively, the alignments determined by the domain expert and ontology matcher. Recall and precision, respectively, measure \( A \)'s completeness and soundness, \( f \)-measure is their harmonic mean, and \( \alpha \in [0, 1] \) is the weight to trade-off recall and precision. However, \( f \)-measure requires domain experts to provide \( R \), which limits its application in real practice. Supposing a golden biomedical ontology alignment’s cardinality is 1:1, since the larger cardinality of the concept mapping set is and the higher the found correspondences’ mean similarity value is, the better the alignment quality is, three approximate measures, i.e., recall’, precision’, and \( f \)-measure’, are used to, respectively, approximate the original recall, precision, and \( f \)-measure [22]:
\[
\text{recall}’ = \frac{|A|}{\max(|C_1|, |C_2|)},
\]
\[
\text{precision}’ = \frac{\sum_{i=1}^{A} \text{simValue}_i}{|A|},
\]
\[
f - \text{measure}’ = \frac{\text{recall}’ \times \text{precision}’}{\alpha \times \text{recall}’ + (1 - \alpha) \times \text{precision}’},
\]
where \(|C_1| \) and \(|C_2| \) are two ontologies’ concept scale, \(|A| \) is \( A \)'s concept mapping number, and \( \text{simValue}_i \) is the \( i \)-th concept mapping’s similarity value.

Finally, biomedical ontology matching problem’s discrete multiobjective optimal model is defined as follows:
\[
\begin{align*}
\max & f(x) = (r(x), p(x)), \\
\text{s.t.} & X = \left[ x_1, x_2, \ldots, x_{|C_1|} \right]^T, \\
& x_i \in \{1, 2, \ldots, |C_2| \}, i = 1, 2, \ldots, |C_1|,
\end{align*}
\]
where \(|C_1| \) and \(|C_2| \) are, respectively, the concept number of two biomedical ontologies, \( x_i \) means the \( i \)-th pair of concept correspondence, i.e., \( i \)-th source concept is mapped to target \( x_i \)-th concept, and two objective functions \( r \) and \( p \) are to, respectively, maximize recall’ and precision’ of \( X \)'s corresponding alignment.

### 3.2. Similarity Measure on Biomedical Concept

In this work, a hybrid similarity measure is used to enhance the confidence of the calculated similarity value, which takes into consideration the concept’s syntax and linguistic and context information. First, for each biomedical concept, the information (the label, comment, and property name) from itself and its context concept are put into its separated profile. Then, two biomedical concepts’ similarity value is calculated according to the following equation:
\[
\text{sim}(p_1, p_2) = \frac{\sum_{i=1}^{f} \max_{j=1}^{g} \text{sim}(p_{1i}, p_{2j}) + \sum_{i=1}^{g} \max_{j=1}^{f} \text{sim}(p_{1i}, p_{2j}))}{f + g},
\]
where \( p_1 \) and \( p_2 \) are, respectively, two biomedical concept’s corresponding profiles and \( \text{sim}(p_{1i}, p_{2j}) \) is calculated by integrating the SMOA [23] based on syntax measure and Unified Medical Language System (UMLS) [24] based on linguistic measure.
\[
\text{sim}(p_{1i}, p_{2j}) = \begin{cases} 
1, & \text{if two elements are synonymous in UMLS} \\
\text{SMOA}(p_{1i}, p_{2j}), & \text{otherwise}
\end{cases}
\]

### 4. Compact Multiobjective Swarm Optimization Algorithm Driven by Knee Solution

This work proposes a CMPSO-K for solving the biomedical ontology matching problem, which approximates the population-based PSO’s evolving process through a probability vector (PV) [25]. In the next, the objective decomposition approach, the encoding mechanism, and knee solution determination are presented, respectively, and finally, the details of CMPSO-K are shown through the pseudo-code.

#### 4.1. Objective Decomposition

In this work, the weighted sum approach is used to transform a MOP into a set of subproblems and solves them simultaneously.
\[
\max g^w(x) = \sum_{i=1}^{N} w_i \cdot f_i(x),
\]
\[
s.t. \ x \in \Omega \subset \mathbb{R}^n
\]
where \( N \) is the number of decomposed problems and \( w_i \in [0, 1] \), \( i = 1, \ldots, N \), \( \sum_{i=1}^{N} w_i = 1 \). Two objectives of our problem are to maximize recall’ and precision’, and the \( i \)-th subproblem’s objective is defined as follows:
\[
\max \frac{\text{recall}’ \times \text{precision}’}{\alpha_i \times \text{recall}’ + (1 - \alpha_i) \times \text{precision}’},
\]
where \( \alpha_1 = 0 \) and \( \alpha_i = i - 1/N - 1, \ldots, \infty, \alpha_N = 1 \).

#### 4.2. Encoding Mechanism

In this work, the Gray coding, a binary encoding mechanism, is used to encode an alignment. As can be seen from Figure 1, an example of the encoding mechanism is shown, in which the source concept “uterine gland” with index 8 is mapped to target concept “Uterine Gland” with index 6 whose Gray code is 110. In particular, Gray code 000 means an individual by increasing (or decreasing) its element value if the corresponding element of that individual is 1 (or 0).
PV: \{ 0.1, 0.3, 0.5, 0.9 \}

Random numbers: 0.2, 0.4, 0.6, 0.1

0.2 > 0.1 0.4 > 0.3 0.6 > 0.5 0.1 < 0.9

Generated individual 0001

Figure 1: An example of the encoding mechanism.

Figure 2: An example of generating an individual through PV.

Input: two individuals \textit{ind}_1 and \textit{ind}_2, crossover probability \( p_{cr} \)
Output: a new individual \( \textit{ind}_{new} \)

1. \( \textit{ind}_{new} = \textit{ind}_1 \);
2. \textbf{for} \( i = 1; i \leq \textit{ind}_1\.length; i++ \) \textbf{do}
3. \hspace{1em} \textbf{if} \( \text{rand}(0,1) < p_{cr} \) \textbf{then}
4. \hspace{2em} \( \textit{ind}_{new,i} = \textit{ind}_{2,i} \);
5. \hspace{1em} \textbf{end if}
6. \textbf{end for}
7. \textbf{return} \( \textit{ind}_{new} \);

Algorithm 1: Uniform crossover.

Figure 3: An example of knee solution and user preferred region.
4.3. Crossover Operator. The crossover operator generates one child individual by mixing the information of two parent individuals, which is carried out according to the crossover probability. In this work, the uniform crossover operator is used, whose pseudo-code is shown in Algorithm 1. Given two parent solutions \( \text{ind}_1 \) and \( \text{ind}_2 \), each gene bit value of their offspring \( \text{ind} \) new is the same with the corresponding gene bit value of \( \text{ind}_1 \) (or \( \text{ind}_2 \)) when a random number \( \text{rand} \) \((0,1)\) is larger than (or smaller than) the crossover rate \( p_c \).

4.4. Determination of Knee Solution. Figure 3 shows an example of knee solution and user preferred region. As can be seen from the figure, the knee solution, a subproblem’s elite solution, and the PF form a user preferred region for that subproblem. In each generation, the newly generated solution tends to move toward that region to meet a DM’s requirement. Thus, the knee solution’s determination is of utmost importance for the algorithm’s search performance. In this work, for those solutions in the PF, a max-min approach is used to determine the knee solution. In particular, suppose \( \text{ind}_1 \) and \( \text{ind}_2 \) are two solutions in the PF, and \( \text{recall}_i \) and \( \text{precision}_i \), \( i = 1, 2 \), are, respectively, their recall and precision, and they can be compared according to the following formula:

\[
x_i = \arg \max \{ \min \{ \text{recall}_i, \text{precision}_i \} \}.
\]

On this basis, a best solution can be selected from PF solutions, and this procedure is shown in Algorithm 2.

4.5. Pseudo-Code of Compact Multiobjective Particle Swarm Optimization Algorithm Driven by Knee Solution. The pseudo-code of CMPSO-K is presented in Algorithm 3. CMPSO-K first divides the problem into three subproblems that, respectively, maximize \( f_{a=0}, f_{a=0.5}, \) and \( f_{a=1} \). First, three PVs and local best individuals are, respectively, initialized for three subproblems, and then the knee solution (or global best individual) is initialized by first using the nondominated sorting algorithm [26] on the population and then the knee solution in the PF is determined. In each generation, CMPSO-K tries to solve each subproblem by approximating PSO’s position updating strategy, i.e., crossover an individual with the local best individual and global best individual to obtain a new individual, and then use the new one to update the local best individual and PV. After solving each subproblem, a population in the current generation is obtained by 3 local best individuals and 6 individuals in total generated through \( PV_{a=0}, PV_{a=0.5}, \) and \( PV_{a=1} \), respectively, and then the nondominated sorting algorithm is used to determine its PF and the current generation’s knee solution \( \text{ind}_{\text{elite}} \). Through comparing \( \text{ind}_{\text{elite}} \) with historical knee solution \( \text{ind}_{\text{elite}} \) through the max-min approach, the latter can be updated. Finally, when the generation approaches the maximum generation \( \text{maxGen} \), the algorithm terminates and returns \( \text{ind}_{\text{elite}} \).

5. Experiment

To test CMPSO-K’s performance, the experiment exploits three biomedical tracks in Ontology Alignment Evaluation Initiative (OAEI), i.e., anatomy track, large biomed track, and disease and phenotype track. The testing cases in these tracks are all practical ontologies that are widely used in the biomedical domain and open to achieve, which have lots of overlapping information with different representations. OAEI provides the reference alignments for each track to test a matcher’s performance, and Table 1 briefly describes the ontologies in these tracks.

In terms of the alignment’s quality, the EA-based matcher [8], ABC-based matcher [6], PSO-based matcher [10], and OAEI’s participants are compared in Tables 2 and 3, and also, in terms of the memory consumption and converging speed, CMPSO-K-based matcher and other SIA-based matchers are compared in Figures 4 and 5. All the SIAs’ results are the mean values of 30 independent executions. EA, ABC, and PSO’s configurations are referred to their literatures, and CMPSO-K uses the following configuration: number of decomposed problems: \( N = 3 \); maximum generation: \( \text{maxGen} = 3000 \); crossover probability: \( p_c = 0.6 \); and step length for updating PV: \( \text{step} = 0.1 \). This configuration is determined in an empirical way, which represents a trade-off setting in the experiment to obtain the highest average results on all testing cases.

First, Friedman’s test [32] is used to figure out whether all the matchers present any difference and then determines whether one matcher statistically outperforms others through Holm’s test [33]. In Friedman’s test, the null
Input: maximum generation maxGen, crossover rate $p_c$, step length for updating PV step $p_{ste}$
Output: knee solution $ind_{elite}$
(1) "Initialization"
(2) initialize generation $t = 0$;
(3) initialize $PV_{a=0}$, $PV_{a=0.5}$ and $PV_{a=1}$ by setting all the elements inside as 0.5;
(4) initialize three local best individuals $ind_{0,elite}$, $ind_{0.5,elite}$ and $ind_{1,elite}$;
(5) generate 5 individuals through $PV_{a=0}$, $PV_{a=0.5}$ and $PV_{a=1}$;
(6) NonDominateSort();
(7) initialize the knee solution (or global best individual) $ind_{elite}$;
(8) "Evolving Process"
(9) while $t < maxGen$ do
(10) "Updating $ind_{0,elite}$"
(11) generate an individual $ind_{0,new}$ through $PV_{a=0}$;
(12) $ind_{0,new} = crossover(ind_{0,new},ind_{0,elite})$;
(13) $ind_{0,5,new} = crossover(ind_{0,new},ind_{0.5,elite})$;
(14) [winner,loser] = compete($ind_{0,new}$, $ind_{0,elite}$);
(15) updatePV(winner,loser);
(16) if winner == $ind_{0,new}$ then
(17) $ind_{0,elite} = ind_{0,new}$;
(18) end if
(19) "Updating $ind_{0.5,elite}$"
(20) generate an individual $ind_{0.5,new}$ through $PV_{a=0.5}$;
(21) $ind_{0.5,new} = crossover(ind_{0,new},ind_{0.5,elite})$;
(22) $ind_{0.5,elite} = crossover(ind_{0,new},ind_{0.5,elite})$;
(23) [winner,loser] = compete($ind_{0,new}$, $ind_{0.5,elite}$);
(24) updatePV(winner,loser);
(25) if winner == $ind_{0,new}$ then
(26) $ind_{0.5,elite} = ind_{0,new}$;
(27) end if
(28) "Updating $ind_{1,elite}$"
(29) generate an individual $ind_{1,new}$ through $PV_{a=1}$;
(30) $ind_{1,new} = crossover(ind_{1,new},ind_{1,elite})$;
(31) $ind_{1,elite} = crossover(ind_{1,new},ind_{1,elite})$;
(32) [winner,loser] = compete($ind_{1,new}$, $ind_{1,elite}$);
(33) updatePV(winner,loser);
(34) if winner == $ind_{1,new}$ then
(35) $ind_{1,elite} = ind_{1,new}$;
(36) end if
(37) "Updating $ind_{elite}$"
(38) generate 2 individuals through $PV_{a=0}$, $PV_{a=0.5}$ and $PV_{a=1}$ respectively;
(39) NonDominateSort;
(40) determine current generation’s knee solution $ind_{elite}'$;
(41) [winner,loser] = compete($ind_{elite}$, $ind_{elite}'$);
(42) if winner == $ind_{elite}'$ then
(43) $ind_{elite} = ind_{elite}'$;
(44) end if
(45) $t = t + 1$;
(46) end while
(47) return $ind_{elite}$;

Algorithm 3: Compact multiobjective particle swarm optimization algorithm driven by knee solution.

Table 1: Description on the OAEI’s tracks.

<table>
<thead>
<tr>
<th>Track</th>
<th>Subtask</th>
<th>Ontology</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anatomy</td>
<td>MA-HA</td>
<td>Adult Mouse Anatomy (MA)</td>
<td>2,744 classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Human Anatomy (HA)</td>
<td>3,304 classes</td>
</tr>
<tr>
<td>Large biomed</td>
<td>FMA-NCI</td>
<td>Foundation Model of Anatomy (FMA)</td>
<td>78,989 classes</td>
</tr>
<tr>
<td></td>
<td>FMA-SNOMED</td>
<td>Systemized Nomenclature of Medicine (SNOMED)</td>
<td>122,464 classes</td>
</tr>
<tr>
<td></td>
<td>SNOMED-NCI</td>
<td>National Cancer Institute Thesaurus (NCI)</td>
<td>66,724 classes</td>
</tr>
<tr>
<td>Disease and phenotype</td>
<td>HP-MP</td>
<td>Human Phenotype Ontology (HP)</td>
<td>33,205 classes</td>
</tr>
<tr>
<td></td>
<td>DOID-ORDO</td>
<td>Mammalian Phenotype Ontology (MP)</td>
<td>32,298 classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Human Disease Ontology (DOID)</td>
<td>24,034 classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Orphanet Rare Disease Ontology (ORDO)</td>
<td>68,009 classes</td>
</tr>
</tbody>
</table>
Table 2: Friedman’s test on the alignment’s quality. Each value represents the f-measure, and the number in round parentheses is the corresponding computed rank.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Anatomy</th>
<th>FMA-NCI</th>
<th>FMA-SNOMED</th>
<th>NCI-SNOMED</th>
<th>HP-MP</th>
<th>DOID-ORDO</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>AML [27]</td>
<td>0.94 (1.5)</td>
<td>0.93 (1.5)</td>
<td>0.83 (2)</td>
<td>0.80 (2)</td>
<td>0.84 (3)</td>
<td>0.64 (7)</td>
<td>0.83 (2.83)</td>
</tr>
<tr>
<td>LogMap [28]</td>
<td>0.89 (4)</td>
<td>0.92 (3)</td>
<td>0.79 (3)</td>
<td>0.77 (3)</td>
<td>0.85 (2)</td>
<td>0.84 (2)</td>
<td>0.84 (2.83)</td>
</tr>
<tr>
<td>XMap [29]</td>
<td>0.89 (4)</td>
<td>0.86 (5.5)</td>
<td>0.77 (4)</td>
<td>0.69 (4)</td>
<td>0.47 (8.5)</td>
<td>0.70 (6)</td>
<td>0.73 (2.33)</td>
</tr>
<tr>
<td>DOME [30]</td>
<td>0.76 (6)</td>
<td>0.86 (5.5)</td>
<td>0.33 (9)</td>
<td>0.64 (6)</td>
<td>0.47 (8.5)</td>
<td>0.60 (9)</td>
<td>0.61 (7.33)</td>
</tr>
<tr>
<td>POMAP++ [31]</td>
<td>0.89 (4)</td>
<td>0.88 (4)</td>
<td>0.40 (8)</td>
<td>0.68 (5)</td>
<td>0.68 (6)</td>
<td>0.83 (3)</td>
<td>0.73 (5.00)</td>
</tr>
<tr>
<td>EA [8]</td>
<td>0.68 (9)</td>
<td>0.73 (8)</td>
<td>0.64 (6)</td>
<td>0.53 (8)</td>
<td>0.70 (5)</td>
<td>0.61 (8)</td>
<td>0.65 (7.33)</td>
</tr>
<tr>
<td>ABC [6]</td>
<td>0.72 (7)</td>
<td>0.75 (7)</td>
<td>0.64 (6)</td>
<td>0.62 (7)</td>
<td>0.62 (7)</td>
<td>0.72 (5)</td>
<td>0.68 (6.50)</td>
</tr>
<tr>
<td>PSO [10]</td>
<td>0.70 (8)</td>
<td>0.72 (9)</td>
<td>0.64 (6)</td>
<td>0.51 (9)</td>
<td>0.72 (4)</td>
<td>0.75 (4)</td>
<td>0.67 (6.67)</td>
</tr>
<tr>
<td>CMPSO-K</td>
<td>0.94 (1.5)</td>
<td>0.93 (1.5)</td>
<td>0.85 (1)</td>
<td>0.82 (1)</td>
<td>0.86 (1)</td>
<td>0.86 (1)</td>
<td>0.88 (1.17)</td>
</tr>
</tbody>
</table>

Table 3: Holm’s test on the alignment’s quality.

<table>
<thead>
<tr>
<th>i</th>
<th>Approach</th>
<th>z value</th>
<th>Unadjusted p value</th>
<th>α/k − i , α = 0.05</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>XMap</td>
<td>2.33</td>
<td>0.045</td>
<td>0.050</td>
</tr>
<tr>
<td>7</td>
<td>AML</td>
<td>2.83</td>
<td>0.004</td>
<td>0.025</td>
</tr>
<tr>
<td>6</td>
<td>LogMap</td>
<td>2.83</td>
<td>0.004</td>
<td>0.016</td>
</tr>
<tr>
<td>5</td>
<td>POMAP++</td>
<td>5.00</td>
<td>1.28 × 10⁻⁷</td>
<td>0.012</td>
</tr>
<tr>
<td>4</td>
<td>ABC</td>
<td>6.50</td>
<td>9.82 × 10⁻⁸</td>
<td>0.010</td>
</tr>
<tr>
<td>3</td>
<td>PSO</td>
<td>6.67</td>
<td>3.79 × 10⁻⁸</td>
<td>0.008</td>
</tr>
<tr>
<td>2</td>
<td>EA</td>
<td>7.33</td>
<td>5.38 × 10⁻¹¹</td>
<td>0.007</td>
</tr>
<tr>
<td>1</td>
<td>DOME</td>
<td>7.33</td>
<td>5.38 × 10⁻¹¹</td>
<td>0.006</td>
</tr>
</tbody>
</table>

Figure 4: Comparison on the converging speed, (a) comparison on the anatomy track, (b) comparison on the large biomed track, and (c) comparison on the disease and phenotype track.
hypothesis is that all the matchers are equivalent, and if the computed value $\chi^2$ must be equal to or greater than the tabled critical chi-square value at the specified level of significance [34], this hypothesis can be rejected. In this work, a level of significance $\alpha = 0.05$ is chosen, and the critical value for 8 degrees of freedom (since there are 9 matchers), i.e., $\chi^2_{0.05} = 15.507$, needs to be considered.

In Table 2, computed $\chi^2 = 54.43$, which is greater than 15.507, and therefore, the null hypothesis can be rejected, and Holm’s test can be further carried out. As shown in Table 2, since the CMPSO-K-based matcher ranks the lowest value, it is set as a control matcher that will be compared with others.

In Holm’s test, $z$ value is the testing statistic for comparing the $i$-th and $j$-th matchers, which is used for finding the $p$ value that is the corresponding probability from the table of the normal distribution. $p$ value is then compared with $\alpha = 0.05$, which is an appropriate level of significance. According to Table 3, it is possible to state that our approach statistically outperforms other biomedical ontology matchers on f-measure at 5% significance level. Since the multiobjective evolving mechanism can better trade-off two objectives and the knee solution can effectively guide the algorithm’s search direction, CMPSO-K’s solutions are much better than other SIAs.

Figures 4 and 5, respectively, compare CMPSO-K with other SIAs on the memory consumption and converging speed. As can be seen from the figures, CMPSO-K can significantly improve the converging speed and reduce the memory consumption, which shows the effectiveness of the compact encoding mechanism and the compact evolutionary operators. To sum up, CMPSO-K-based ontology matching technique can efficiently optimize the biomedical ontology alignments.

### 6. Conclusion and Future Work

To further improve the matching process’s efficiency, in the future, we will be interested in developing a biomedical ontology partitioning technique to split two ontologies into disjoint ontology segments so that the large-scale problem can be converted into several small-scale segment matching problems. After that, the parallel computation can also be further utilized to match the similar segments and improve the matching process’s performance. With respect to the similarity measure, we would like to develop an adaptive biomedical concept similarity measure framework, which can instantiate an effective similarity metric according to two biomedical ontologies’ heterogeneous characteristics.
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 was supported by the National Natural Science Foundation of China (no. 61503082), the Natural Science Foundation of Fujian Province (no. 2016J05145), the Program for New Century Excellent Talents in Fujian Province University (no. GY-Z18155), the Program for Outstanding Young Scientific Researcher in Fujian Province University (no. GY-Z160149), and the Scientific Research Foundation of Fujian University of Technology (nos. GY-Z17162 and GY-Z15007).

References


