Data Science and Pattern Recognition
Ubiquitous International
 ISSN 2520-4165
Volume **3**, Number **1**, March **2019**

Using Population-based Incremental Learning Algorithm for Matching Class Diagrams

Xingsi Xue

College of Information Science and Engineering
Intelligent Information Processing Research Center
Fujian Provincial Key Laboratory of Big Data Mining and Applications
Fujian Key Lab for Automotive Electronics and Electric Drive
Fujian University of Technology
No.33 Xuefu South Road, University Town, Minhou, Fuzhou City, Fujian Province, 350118, China
jack8375@gmail.com

Haiyan Yang

College of Information Science and Engineering
Fujian University of Technology
No.33 Xuefu South Road, University Town, Minhou, Fuzhou City, Fujian Province, 350118, China
yanghy@fjut.edu.cn

Jie Zhang

School of Computer Science and Engineering
Yulin Normal University
No.299 Education Middle Road, Yulin City, Guanxi Province, 537000, China
jgxyzjzj@126.com

Abstract. *Model matching is able to determine the correspondences between two model's elements, which serves as a basis for such model management operations as model retrieval, consolidation, reuse, and evolution. In this work, we employ a Population-Based Incremental Learning algorithm (PBIL) in the class diagram matching context to improve the quality of class diagram alignments. In particular, a novel optimal model for the class diagram matching problem is constructed, a profile-based class similarity measure is presented, and a PBIL-based class diagram matching technique is proposed to efficiently determine the high quality class diagram alignments. The experimental results show the effectiveness of our proposal.*
Keywords: class diagram matching, Population-Based Incremental Learning algorithm, class similarity measure

1. **Introduction.** Models are of the utmost importance for the developers to have different viewpoints of visualization about a software system [1]. Model matching is able to determine the correspondences between two model's elements, which serves as a basis for such model management operations as model retrieval, consolidation, reuse, and evolution. In this work, we mainly focus on the model represented by the Unified Modeling Language (UML) [1] class diagram. Research on matching is concerned with the task of identifying relations between concepts in different artifacts [2], and in the past, researchers have developed a lot of matching approaches in the alongside related domain such as schema matching and ontology matching [3]. Since matching class diagrams is

[1]https://en.wikipedia.org/wiki/Unified_Modeling_Language

a complex (nonlinear problem with many local optimal solutions) and time-consuming task (large-scale problem), particularly when the number of classes is significantly large, approximate methods are usually used for determining the class alignments [4]. From this point of view, the Population-Based Incremental Learning algorithm [5] could represent an efficient approach for addressing this problem. In particular, our main contributions in this paper are as follows:

- An optimal model for the class diagram matching problem is constructed;
- A profile-based class similarity measure is presented;
- A PBIL-based class diagram matching technique is proposed to efficiently determine the high quality class diagram alignments.

The rest of the paper is organized as follows: Section 2 defines the class diagram, class diagram alignment and class diagram matching problem, and describes the class similarity measure; Section 3 presents the PBIL-based class diagram matching technique in details; Section 4 gives the experimental studies and analysis; finally, Section 5 draws the conclusions.

2. **Preliminary.**

2.1. **Class Diagram and Class Diagram Alignment.**

In this work, a class diagram D can be defined as a 2-tuples $D = (C, R)$, where C and R are respectively referred to class set and relationship set [6]. A class c/inC can be defined as 3-tuples $c = (name, P, M)$, where $name$, P and M are the name, property set and method set of c, respectively. In addition, a class diagram alignment A between two class diagram is a correspondence set, and each correspondence is a 4-tuples $(c_1, c_2, confidence, relation)$, where c_1 and c_2 are respectively the classes of two class diagrams, $confidence \in [0, 1]$ is a confidence value for the correspondence between c_1 and c_2 and $relation$ is the relation of equivalence.

2.2. **Class Similarity Measure.**

2.3. **Biomedical Concept Similarity Measure.**

Class similarity measure is the foundation of class diagram matching [7]. In this work, we utilize a profile-based similarity measure [8] to calculate similarity value between two classes. First, for each class, we construct a profile for it by collecting the name, property name, and method name from itself and all its ascendants. Then, the similarity of two classes c_1 and c_2 is measured based on the similarity of their profiles p^1 and p^2 through the following equation:

$$\frac{\sum_{i=1}^{|p^1|} \max\limits_{j=1\cdots|p^2|}(sim'(p_i^1, p_j^2)) + \sum_{j=1}^{|p^2|} \max\limits_{i=1\cdots|p^1|}(sim'(p_j^2, p_i^1))}{2 \times min(|p^1|, |p^2|)} \tag{1}$$

where:

- $|p^1|$ and $|p^2|$ are the cardinalities of p^1 and p^2, respectively,
- p_i^1 is the ith element of p^1, and p_j^2 is the jth element of p^2,
- $sim'()$ calculates the similarity of p_i^1 and p_j^2 by SMOA [9] and WordNet [10], which is defined as follows:
  $sim'(p_i^1, p_j^2) = \begin{cases} 1, & \text{if two words are synonymous in Wordnet} \\ SMOA(p_i^1, p_j^2), & \text{otherwise} \end{cases}$

2.4. **Class Diagram Matching Problem.** Supposing the golden alignment is one to one, i.e. one class in source class diagram is matched with only one class in target class diagram and vice versa, based on the observations that the more correspondences found and the higher mean similarity values of the correspondences are, the better the alignment quality is [11], we use the following equation to measure a class diagram alignment's quality:

$$I(A) = 2 \times \frac{\phi(A) \times \frac{\sum_{i=1}^{|A|} \delta_i}{|A|}}{\phi(A) + \frac{\sum_{i=1}^{|A|} \delta_i}{|A|}} \tag{2}$$

where $|A|$ is the number of correspondences in A, ϕ is a function of normalization in [0,1], δ_i is the similarity value of the ith correspondence in A. On this basis, the optimal model of class diagram matching problem is defined as follows:

$$\begin{cases} max \quad I(X) \\ s.t. \quad X = (x_1, x_2, ..., x_{|D_1|}, x_{|D_1|+1})^T \\ \qquad x_i = 1, 2, \cdots, |D_2| \\ \qquad x_{|D_1|+1} \in [0, 1] \end{cases} \tag{3}$$

where the decision variable X represents an alignment between the class diagrams D_1 and D_2, x_i represent the ith correspondence between ith class in D_1 and x_ith class in D_2, $|D_1|$ and $|D_2|$ are the cardinalities of the class set in D_1 and D_2 respectively, and $x_{|D_1|+1} \in [0, 1]$ is the threshold to filter the final alignment.

3. **Population-based Incremental Learning Algorithm.** Model-based optimization using probabilistic modeling of the search space is one of the areas where research on EA has considerably advanced in recent years, and PBIL is one of the first algorithms of its kind [12]. PBIL uses the Probability Vector (PV) representation for defining a population. Rather than passively transforming each population into a probability vector, from which solution vectors are generated and recombined. Therefore, a run of PBIL is able to highly improve the performance of solving large scale matching problem in terms of both memory consumption and runtime [13].

In this work, PBIL's solution is encoded through the binary coding mechanism, which can be divided into two parts. The first part stands for the correspondences in the alignment, and the other one stands for a threshold. Given the total number of activities in source and target class sets C_1 and C_2, the first part of a chromosome consists of $|C_1|$ gene segments, and the Binary Code Length (BCL) of each gene segment is equal to $\lfloor \log_2(C_2) + 0.5 \rfloor$, which ensures each gene segment could present any target class's index. While, the second part of a chromosome has only one gene segment, whose BCL is equal to $\lfloor \log_2(\frac{1}{numAccuracy}) + 0.5 \rfloor$, which can ensure this gene segment can present any threshold value under the numerical accuracy $numAccuracy$. Thus, the total length of the chromosome is equal to $C_1 \times \lfloor \log_2(n_t) + 0.5 \rfloor + \lfloor \log_2(\frac{1}{numAccuracy}) + 0.5 \rfloor$. The aim of PBIL is to actively create a PV which, with high probability, represents a population of high evaluation solution vectors. Unlike the mechanisms inherent to a Genetic Algorithm (GA), operations are not defined on the population; rather, in PBIL, operations take place directly on the PV. These mechanisms are derived from those used in competitive learning. In this work, we use one PV to characterize the entire population, and the number of elements in PV is equal to the number of solution's gene bits and each element's value is in [0,1]. Since each element's value in PV represents the probability of being one, we can use PV to generate various solutions. In addition, PV can be updated based on

TABLE 1. Comparison with two state-of-the-art class diagram matchers in terms of alignment's quality.

Testing Case	GA-based matcher $f(r,p)$	ACO-based matcher $f(r,p)$	PBIL-based matcher $f(r,p)$
M1M2	0.82 (0.85,0.80)	0.85 (0.90,0.80)	0.92 (0.95,0.90)
M1M3	0.72 (0.70,0.75)	0.90 (0.90,0.90)	0.93(0.95,0.92)
M1M4	0.72(0.70,0.75)	0.82 (0.85,0.80)	0.93(0.92,0.94)
M1M5	0.70(0.65,0.75)	0.90 (0.85,0.95)	0.94 (0.88,1.00)
M2M3	0.70 (0.70,0.70)	0.95 (0.90,1.00)	0.96 (0.93,1.00)
M2M4	0.70(0.70,0.70)	0.90(0.90,0.90)	0.94 (0.97,0.92)
M2M5	0.67 (0.60,0.75)	0.80(0.80,0.80)	0.86 (0.85,0.88)
M3M4	0.67(0.60,0.75)	0.85 (0.80,0.90)	0.91(0.85,0.97)
M3M5	0.77 (0.80,0.75)	0.95 (0.90,1.00)	0.95(0.91,1.00)
M4M5	0.75 (0.70,0.80)	0.85(0.80,0.90)	0.87 (0.82,0.92)
Average	0.72(0.70,0.75)	0.88(0.86,0.90)	0.92(0.90,0.95)

the better solution in terms of its fitness value, with the aim to move the PV toward the better solution. In the following, the pseudo-code of the improved PBIL is given:

4. **Experimental Studies and Analysis.** In the experiment, we test the performance of PBIL-based matcher by comparing with two state-the-of-art class diagram matching techniques, i.e. the Genetic Algorithm (GA) based matcher [14] and the Ant Colony Optimization algorithm (ACO) based matcher [15]. We use the class diagrams generated by reverse engineering an open source system, ezmorph [2]. The system has 12 releases, and to allow for more variation between the generated class diagrams, five non-consecutive releases are selected. The smallest diagram has 49 classes and the largest diagram has 71 classes.

PBIL uses the following parameters which represent a trade-off setting obtained in an empirical way to achieve the highest average alignment quality on all exploited testing cases:

- Numerical accuracy = 0.01;
- Learning rate = 0.1;
- Crossover probability = 0.6;
- Mutation probability = 0.03;
- Mutation shift = 0.05;
- Maximum generation = 3000.

The experimental results of GA-based matcher and ACO-based matcher are from their literatures, and the experimental results PBIL-based matcher in the table are the average values over thirty independent runs. The symbols r, p and f in the table are recall, precision and f-measure [16] of the obtained alignment, respectively.

As can be seen from Table 1 and Figure 1, PBIL-based matcher can efficiently determine better alignments on all testing cases than GA-based matcher and ACO-based matcher. In particular, the precision of our approach is in generally high, which show the effectiveness of the class similarity measure. PBIL combines the mechanisms of a classic Genetic Algorithm (GA) with a competitive learning, and the results achieved by PBIL are better

[2]http://sourceforge.net/projects/ezmorph/?source=directory

Algorithm 1 Population-based Incremental Learning Algorithm

Require:

- $maxGen$: maximum number of generations;
- LR: learning rate;
- p_c: crossover probability;
- p_m: PV mutation probability;
- MS: mutation shift.

Ensure: ind_{elite}: the solution with best objective value.

{Step 1. Initialization}
$gen = 0$;
for $i = 0$; $i < PV.length$; i++ **do**
  $PV[i] = 0.5$;
end for
generate an individual through PV to initialize ind_{elite};
{Step 2. Evolving Process}
{Step 2.1 Crossover}
generate an individual ind_{new} through PV;
$[winner, loser] = compete(ind_{elite}, ind_{new})$;
if $winner == ind_{new}$ **then**
  $ind_{elite} = ind_{new}$;
end if
for $i = 0$; $i < PV.length$; i++ **do**
  if winner[i]==1 **then**
    $PV[i] = PV[i] + LR$;
    if $PV[i] > 1$ **then**
      $PV[i] = 1$;
    end if
  else
    $PV[i] = PV[i] - LR$;
    if $PV[i] < 0$ **then**
      $PV[i] = 0$;
    end if
  end if
end for
{Step 2.2 Mutation}
for $i = 0$; $i < num$; i++ **do**
  if $(rand(0, 1) < p_m$ **then**
    PV[i]=PV[i]$\times(1 - MS) + rand(0$ or $1) \times MS$;
  end if
end for
{Step 3. Termination}
if $maxGen$ is reached or each bit of PV is either 1 or 0 **then**
  return ind_{elite};
else
  $gen = gen + 1$;
  go to Step 2;
end if

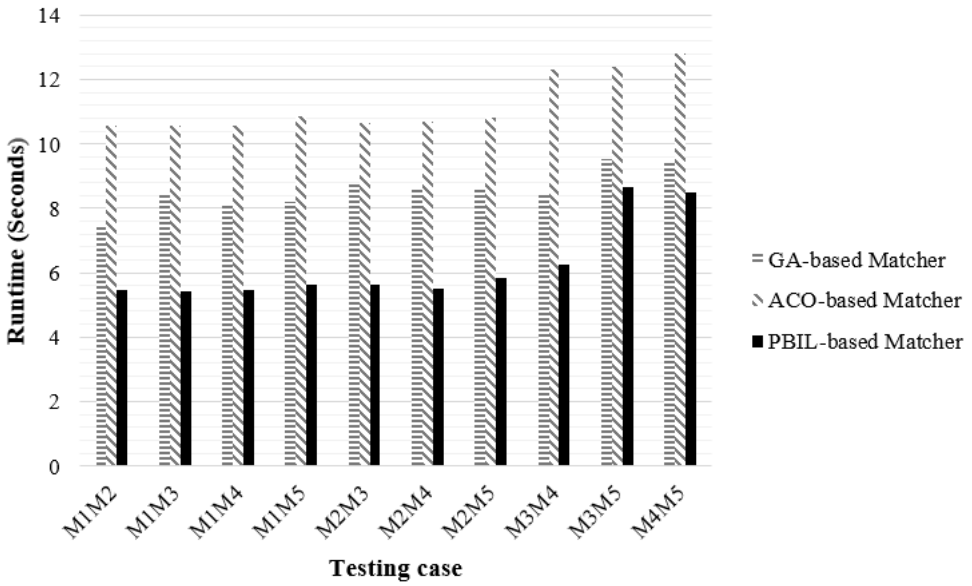


FIGURE 1. Comparison with two state-of-the-art class diagram matchers in terms of runtime (seconds).

and are attained faster than classic GA. The gains in solution quality and runtime are achieved respectively due to PBIL's particular competitive learning, which is effective to lead the algorithm to determine the optimal solution, and the simplicity of PBIL, which does not require all the mechanisms of a GA, rather the few steps in the algorithm are small and simple. When applied to complex optimization problems, the performance of state-of-the-art evolutionary computation techniques, such as GA and ACO, depends largely on the adequate setting of parameters like the probabilities of cross and mutation, size of the population, rate of generational reproduction and so forth, and they show a poor performance when the designed operators do not guarantee the correct movements of the populations in the search space. Comparing with state-of-the-art evolutionary computation techniques, PBIL works based on the probabilistic modeling of promising solutions, which makes it easier to predict the movements of the populations in the search space as well as to avoid the need for many parameters. It has been underlined that when optimization problems contain dependencies between variables, PBIL can more effectively learn on the learning step and propose the new generation of individuals accordingly.

5. **Conclusion.** To efficiently match class diagrams, in this paper, a PBIL-based class diagram matching technique is proposed. To this end, a single-objective optimal model is constructed for the class diagram matching problem, a class similarity measure is presented to distinguish the identical classes and a problem-specific PBIL is proposed to efficiently solve the class diagram problem. The experimental results show that PBIL-based matcher can significantly improve the quality of GA-based matcher and ACO-based matcher.

Acknowledgment. This work is supported by the Natural Science Foundation of Fujian Province (Nos. 2016J05145 and 2015J01652), 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), Scientific Research Foundation of Fujian University of Technology (No. GY-Z17162), the National

Natural Science Foundation of China (No. 61841603) and the Guangxi Natural Science Foundation (No. 2018JJA170050).

REFERENCES

[1] J. Kienzle, G. Mussbacher, B. Combemale, and J. Deantoni, "A unifying framework for homogeneous model composition," *Software and Systems Modeling*, vol. 2019, pp. 1–19, 2019.

[2] A. Bhattacharjee and H. Jamil, "A schema matching system for on-the-fly autonomous data integration," *International Journal of Information and Decision Sciences*, vol.4(2-3), pp. 167–181, 2017.

[3] K. Ramar and G. Gurunathan, "Technical review on ontology mapping techniques," *Asian Journal of Information Technology*, vol. 15(4), pp.676–688, 2016.

[4] H. O. Salami and M. A. Ahmed, "A Framework for Class Diagram Retrieval Using Genetic Algorithm," *The 24th International Conference on Software Engineering and Knowledge Engineering*, pp.737–740, 2012.

[5] X. Xue and J. Chen, "Optimizing ontology alignment through hybrid population-based incremental learning algorithm," *Memetic Computing*, vol. 2018, pp. 1–9, 2018.

[6] M. Zapata-Barra, A. Rodriguez, A. Caro, and E. B. Fernandez, "Towards Obtaining UML Class Diagrams from Secure Business Processes Using Security Patterns," *Journal of Universal Computer Science*, vol. 24(10), pp. 1472–1492, 2018.

[7] M. A. R. Al-Khiaty and M. Ahmed, "Similarity assessment of UML class diagrams using simulated annealing," *5th International Conference on Software Engineering and Service Science*, pp. 19–23, 2014.

[8] X. Xue and Y. Wang, "Using memetic algorithm for instance coreference resolution," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28(2), pp. 580–591, 2016.

[9] X. Xue and J. S. Pan, "A Compact Co-Evolutionary Algorithm for Sensor Ontology Meta-Matching," *Knowledge and Information Systems*, vol. 56(2), pp. 335–353, 2018.

[10] G. A. Miller, "WordNet: a lexical database for English," *Communications of the ACM*, vol. 38(11), pp. 39–41, 1995.

[11] X. Xue and X. Yao, "Interactive ontology matching based on partial reference alignment," *Applied Soft Computing*, vol.72, pp. 355–370, 2018.

[12] S. H. S. Lee, J. D. Deng, M. K. Purvis, M. Purvis, and L. Peng, "An improved PBIL algorithm for optimal coalition structure generation of smart grids," *The Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 345–356, 2018.

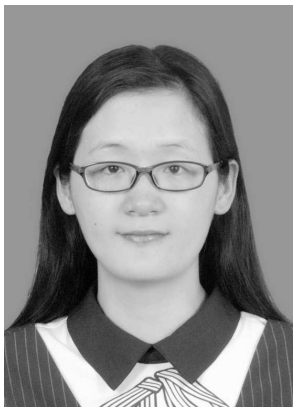
[13] M. H. da Silva, A. P. Legey, and A. C. D. A. Mol, "The evolution of PBIL algorithm when used to solve the nuclear reload optimization problem," *Annals of Nuclear Energy*, vol. 113, pp. 393–398, 2018.

[14] M. A. R. Al-Khiaty and M. Ahmed, "Matching UML class diagrams using a Hybridized Greedy-Genetic algorithm," *The 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies*, pp. 161–166, 2017.

[15] M. A. R. Al-Khiaty, "Ant Colony Optimization for Matching Class Diagrams," *The 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies*, pp. 132–135, 2018.

[16] C. J. Van Rijsberge, "Foundation of evaluation," *Journal of documentation*, vol. 30(4), pp. 365–373, 1974.

Xingsi Xue received the B. S. degree in Software Engineering from Fuzhou University, China in 2004, the M. S. degree in Computer Application Technology from Renmin University of China, China in 2009, and the Ph.D. degree in Computer Application Technology from Xidian University, China in 2014. He is a professor at the College of Information Science and Engineering, Fujian University of Technology, and the director of Intelligent Information Processing Research Center, Fujian University of Technology. He is also the kernel members in Intelligent Information Processing Research Center, Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian Key Lab for Automotive Electronics and Electric Drive at Fujian university of Technology. His research interests include intelligent computation, data mining and large-scale ontology matching technology. He is a member of IEEE and ACM, and won 2017 ACM Xian Rising Star Award and IIH-MSP 2016 excellent paper award.



Haiyan Yang received the B. S. degree in computer Application Technology from Xiangtan University, China in 2002, the M. S. degree in communication and information system from Central South University, China in 2007, and the Ph.D. degree in Computer Application Technology from Central South University, China in 2015. She is an associate professor at the College of Information Science and Engineering, Fujian University of Technology, her research interests include intelligent computation, image process and pattern recognition.

Jie Zhang received the Ph.D. degree in computer science and technology from Xidian University, China in 2013. He is currently an associate professor at Yulin Normal University, Yulin, China. His current main research interests include brain network analysis, single-cell data analysis and evolutionary computation.