Abstract—Fuzzy genetics-based machine learning (FGBML) is one of the representative approaches to obtain a set of fuzzy if-then rules by evolutionary computation. A number of FGBML methods have been proposed so far. Among them, Michigan-style approaches are popular thanks to its lower computational cost than Pittsburgh approaches. In this study, we introduce two simple modifications for our Michigan-style FGBML. One is related to heuristic rule generation. In the original FGBML, each rule in an initial population is generated from a randomly-selected training pattern in a heuristic manner. The heuristic rule generation also performs during evolution where each rule is generated from a misclassified pattern. As its modification, we propose the use of multiple patterns to generate each fuzzy if-then rule. The other is related to the fitness calculation. In the original FGBML, the fitness of each rule is calculated as the number of correctly classified training patterns, while the number of misclassified patterns is ignored. As its modification, we incorporate a penalty term into the fitness function. Through computational experiments using 20 benchmark data sets, we examine the effects of these two modifications on the search ability of our Michigan-style FGBML.

Keywords—Evolutionary fuzzy systems; fuzzy genetics-based machine learning; heuristic rule generation; fitness calculation.

I. INTRODUCTION

Evolutionary computation has frequently been incorporated into optimization or tuning of fuzzy rule-based systems under the name of genetic fuzzy systems (GFS) and evolutionary fuzzy systems (EFS) [1]-[6]. Fuzzy genetics-based machine learning (FGBML) is a kind of EFS which directly optimizes fuzzy if-then rules or a set of fuzzy rules. There are three types of FGBML. One is Pittsburgh-style approaches where a set of fuzzy if-then rules is coded as an individual and is optimized as a fuzzy rule-based system. Another one is Michigan-style approaches where each rule in a population is coded as an individual and is optimized as a member of a fuzzy system. The other is iterative rule learning approaches where each rule is coded as an individual and is iteratively optimized based on a fuzzy system with the previously obtained rules.

In our former study [7], [8], we proposed the Michigan-style FGBML for fuzzy classifier design. It was utilized as local search in the Pittsburgh-style FGBML [9]. In [10], the multiobjective formulation of hybrid FGBML (MoFGBML) was proposed and was used for the tradeoff analysis between the accuracy and complexity. The search performance of our hybrid FGBML was clearly examined through experiments comparing with some representative GBML methods [11].

In order to enhance the search ability of our MoFGBML, we revisit our Michigan-style FGBML and propose two simple modifications in this paper. One modification is related to heuristic rule generation. In the original FGBML, heuristic rule generation performs at population initialization and during evolution. For the population initialization, each rule in an initial population is generated from a randomly selected training pattern. During evolution, some rules with low fitness are replaced with ones by the heuristic rule selection. Each new rule is generated from a misclassified pattern. We proposed a simple modification of the heuristic rule generation in [12] where it is applied to MoFGBML. In this paper, the effect of the same idea is examined for our Michigan-style GBML. We use multiple patterns to generate a single rule instead of using a single pattern. We expect that newly generated rules from multiple patterns are more generalized. The other modification is related to the fitness calculation in our Michigan-style FGBML. In the original method, the fitness of each rule is calculated by the number of correctly classified training patterns. In the modified version, a penalty term is incorporated into the fitness function. The effect of this modification was already examined in [7], [8], but we examine the synergy effect between the use of the penalty-incorporated fitness function and the use of multiple patterns for the heuristic rule generation on the search ability of our Michigan-style FGBML.

This paper is organized as follows. First, we explain our Michigan-style FGBML in Section II. In Section III, we introduce the modified heuristic rule generation with multiple training patterns and the fitness calculation with the penalty term. In Section IV, we examine the synergy effects of two modifications through computational experiments. Finally, we conclude this paper in Section V.

II. MICHIGAN-STYLE FUZZY GENETICS-BASED MACHINE LEARNING

A. Fuzzy Rule-Based Classifier

For simplicity, we assume an M-class data sets with m training patterns. We also normalize all the attributes into a
real number in [0, 1]. For a fuzzy rule-based classifier, the following fuzzy if-then rules are used in this paper.

Rule $R_q$: If $x_1$ is $A_{q1}$ and ... and $x_n$ is $A_{qn}$

then Class $C_q$ with $CF_q$, \hspace{1cm} (1)

where $R_q$ is a rule label for the $q$th rule. $x_i$ is the $i$th attribute value of the pattern $x$. $A_{qi}$ is an antecedent condition for the $i$th attribute of the $q$th rule. $C_q$ and $CF_q$ are a consequent class and a rule weight of the $q$th rule, respectively. Fourteen fuzzy sets shown in Fig. 1 are used as antecedent conditions in this paper, while only five fuzzy sets shown in the bottom-right of Fig. 1 were used in [7], [8]. In addition to fuzzy sets, “don’t care” condition is also used to make a rule generalized.

![Fig. 1. Fourteen triangular fuzzy sets with different granularities.](image)

Once the antecedent part of a rule is specified by heuristic rule generation or genetic operations, the consequent part (i.e., class $C$ and rule weight $CF$) of its rule is determined according to the training patterns. First, the compatibility grade of the class $C_q$ rule generation or genetic operations, the consequent part (i.e., a consequent class and a rule weight of the $q$th rule, respectively. Fourteen fuzzy sets shown in Fig. 1 are used as antecedent conditions in this paper, while only five fuzzy sets shown in the bottom-right of Fig. 1 were used in [7], [8]. In addition to fuzzy sets, “don’t care” condition is also used to make a rule generalized.

The consequent class $C_q$ is determined by identifying the class with the maximum confidence as:

$$c(A_q \Rightarrow \text{Class } h) = \frac{\sum_{p=1}^{n} \mu_{A_q}(x_p)}{\sum_{p=1}^{n} \mu_{A_q}(x_p)}.$$ \hspace{1cm} (3)

The consequent class $C_q$ is determined by identifying the class with the maximum confidence as:

$$c(A_q \Rightarrow \text{Class } C_q) = \max_{h=1,2,\ldots,m} \{c(A_q \Rightarrow \text{Class } h)\}.$$ \hspace{1cm} (4)

If there exist multiple classes have the same maximum confidence, this rule is not generated. The rule weight $CF_q$ is determined by the difference in the confidence as:

$$CF_q = c(A_q \Rightarrow \text{Class } C_q) - \sum_{h=1 \atop h \neq C_q}^{M} c(A_q \Rightarrow \text{Class } h).$$ \hspace{1cm} (5)

A fuzzy classifier is composed of a prespecified number of fuzzy if-then rules explained above.

### B. Procedure of FGBML

The original procedure of our Michigan-style FGBML is shown as follows:

- **Step 1**: Initialize a population by heuristic rule generation,
- **Step 2**: Evaluate each fuzzy rule in the current population,
- **Step 3**: Generate new fuzzy rules by genetic operations and heuristic rule generation,
- **Step 4**: Replace a part of the current population with the newly generated rules.
- **Step 5**: Return to Step 2 until the stopping condition is satisfied.

Since the consequent class and rule weight of each rule are determined by (2)-(5), only a combination of antecedent conditions for its rule is coded as an individual.

In Step 1, $N_{pop}$ training patterns are randomly selected without replacement. $N_{pop}$ is the same as the population size. A single fuzzy rule is generated from each selected training pattern by probabilistically choosing one of 14 fuzzy sets $B_u (u = 1, 2, \ldots, 14)$ in Fig. 1 for each attribute. The selection probability of each fuzzy set $B_u$ is calculated as:

$$P(B_u) = \frac{\mu_{B_u}(x_{pi})}{\sum_{i=1}^{14} \mu_{B_u}(x_{pi})}.$$ \hspace{1cm} (6)

To make this rule generalized, each antecedent condition is replaced with a “don’t care” condition using the don’t care probability $P_{DC}$. This probability is empirically specified as $(n-5)/n$ ($n$: the number of attributes) [13]. Thus, each initial fuzzy rule is expected to have five fuzzy antecedent conditions, while “don’t care” condition is assigned to the other attributes. This operation is called heuristic rule generation.

In Step 2, the current population is regarded as a fuzzy classifier $S$. We employ a winner-take-all manner. When an input pattern $x_p$ is to be classified by $S$, a single winner rule $R_w$ is chosen from $S$ as:

$$\mu_{A_{q_w}}(x_p) \cdot CF_w = \max \{\mu_{A_{q}}(x_p) \cdot CF_q \mid R_q \in S\}.$$ \hspace{1cm} (7)

The input pattern $x_p$ is classified as the class $C_{w}$ of the winner rule $R_w$. When multiple rules with the maximum value of (7) have different consequent classes, the classification of $x_p$ is rejected. If there is no compatible fuzzy rule with $x_p$, its classification is also rejected. Rejected patterns are regarded as misclassified patterns in this study.

After all the training patterns are classified by $S$, the fitness of each fuzzy rule is calculated as follows:

$$\text{fitness}(R_q) = NCP(R_q),$$ \hspace{1cm} (8)

where $NCP(R_q)$ is the number of correctly classified training patterns by $R_q$. 
In Step 3, \( N_{rep} \) fuzzy rules are generated by genetic operations and the heuristic rule generation. We specify \( N_{rep} \) as 20% of the population size in this paper. As genetic operations, binary tournament selection, uniform crossover, and random replacement mutation are performed. In random replacement mutation, a few antecedent conditions including don’t care are randomly replaced with another condition.

The heuristic rule generation in Step 3 is almost the same as in Step 1, but the difference is that the base patterns are randomly selected from the misclassified patterns in Step 2. Basically, at most \( N_{rep}/2 \) rules are generated by the heuristic rule generation. If the number of misclassified patterns is less than \( N_{rep}/2 \), all the misclassified patterns are used for the heuristic rule generation. The remaining rules are generated by the genetic operations.

In Step 4, \( N_{rep} \) fuzzy rules with lower fitness values in the current population are replaced with the newly generated fuzzy rules in Step 3.

The final output (i.e., a fuzzy classifier) by the Michigan-style FGBML is the classifier with the highest fitness as:

\[
Fitness(S) = \sum_{q \in S} NCP(R_q). \tag{9}
\]

III. SIMPLE MODIFICATIONS OF MICHIGAN-STYLE FGBML

In this paper, we examine the synergy effects of the heuristic rule generation with multiple patterns and the modified fitness calculation with a penalty term on the search ability of our Michigan-style FGBML.

A. Heuristic Rule Generation with Multiple Patterns

As mentioned earlier, a new rule is generated from only a randomly selected training pattern in the original heuristic rule generation at initialization and during evolution. Each attribute condition is probabilistically specified using the compatibility grade with the selected pattern. Then, to avoid the over-specialization, some attribute conditions are replaced with “don’t care”. In our proposed approach, instead of a single pattern, multiple patterns are used for generating a fuzzy rule.

In the proposed rule generation, one base pattern and \((H - 1)\) support patterns are randomly selected to generate a single fuzzy rule. If the \(j\)th support pattern has a different class label from that of the base pattern, the compatibility grade for the \(j\)th support pattern is changed as follows:

\[
\mu_{B_i}(x_{ji}) \leftarrow 1 - \mu_{B_i}(x_{ji}). \tag{10}
\]

In order to select an appropriate antecedent fuzzy set, the minimum compatibility grade for the \(H\) patterns is calculated as follows:

\[
\mu^*_{B_i} = \min\{\mu_{B_i}(x_{1i}), \mu_{B_i}(x_{2i}), \ldots, \mu_{B_i}(x_{Hi})\}, \tag{11}
\]

\[
\mu^*_{B_i} = \begin{cases} 
\mu_{B_i} & \text{if } \mu_{B_i} \geq 0.5 \\
0 & \text{otherwise}
\end{cases} \tag{12}
\]

The selection probability \(P(B_a)\), \(a = 1, 2, \ldots, 14\) of antecedent fuzzy sets for the \(i\)th attribute is calculated as:

\[
P(B_a) = \frac{\mu^*_{B_a}}{\sum_{i=1}^{14} \mu^*_{B_i}}. \tag{13}
\]

If all \(P(B_a)\) are 0 for the \(i\)th attribute, “don’t care” is assigned to that attribute. Therefore, the don’t care probability is not necessary to specify beforehand. This rule will properly cover the selected \(H\) patterns and some neighbors.

At initialization, \(N_{rep}\) patterns are first randomly selected from the training data without replacement as base patterns. Then \((H - 1)\) support patterns for each base pattern are randomly selected from the training patterns. After that, the selection probability is calculated using (10)-(13). During evolution, almost the same procedure is performed. The difference from the initialization is the base pattern selection. At most \(N_{rep}/2\) misclassified patterns are randomly selected without replacement as base patterns.

B. Modified Fitness Calculation with a Penalty Term

In the original fitness calculation, only the number of correctly classified training patterns is calculated, while the number of misclassified patterns is ignored. In this case, undesirable situations may happen sometimes. Let us assume two rules, \(R_a\) and \(R_b\). \(R_a\) is selected as a winner ten times and correctly classifies ten patterns. On the other hand, \(R_b\) is selected as a winner 15 times and correctly classifies eleven patterns. \(R_b\), however, misclassifies four patterns. If one of them has to be removed from a classifier, \(R_b\) will be removed according to (8). But, from the classifier viewpoint, \(R_b\) should be removed because \(R_b\) deteriorates the classification accuracy of the classifier.

To tackle the above-mentioned problem, the simplest idea is to incorporate the penalty term into (8) as follows:

\[
\text{fitness}(R_q) = NCP(R_q) - w_{\text{pen}} \cdot MCP(R_q), \tag{14}
\]

where \(w_{\text{pen}}\) and \(MCP(R_q)\) are the penalty factor and the number of misclassified training patterns by \(R_q\). When \(w_{\text{pen}}\) is specified as 1 or more, \(R_b\) will be removed from the classifier in the above example.

IV. COMPUTATIONAL EXPERIMENTS

In this section, we examine the synergy effects of the modified heuristic rule generation and fitness calculation on the search ability of our Michigan-style FGBML through computational experiments with some benchmark data sets.

A. Experimental Settings

The genetic operators and parameters of our Michigan-style FGBML used in this paper are listed as follows:

- Population size \((N_{pop})\): 30,
- Number of replaced rules \((N_{rep})\): 6,
- Parent selection: Binary tournament selection,
- Crossover: Uniform crossover,
- Crossover probability: 0.9,
- Mutation probability: \(1/n\) \((n: \text{Number of attributes})\),
- Termination condition: 5,000 generations,
- Number of runs: 50 (10-fold cross validation x five times).
We specified the number of selected patterns for heuristic rule generation \( (H) \) and the penalty \( (w_{\text{pen}}) \) as follows:

\[
H: 1, 2, 3, 4, 5, 10, 15, 20, \quad w_{\text{pen}}: 0, 1, 2, 3, 5, 10.
\]

It should be noted that, in the case of \( H = 1 \), each new rule is generated from only a single pattern and the probabilistic replacement with “don’t care” conditions used in other studies. In this case, we specified \( P_{\text{DC}} \) as \((n-5)/n\) as in [13]. If \( w_{\text{pen}} = 0 \) and \( H = 1 \), the algorithm is the same as the original one. We examined all the combination of \( H \) and \( w_{\text{pen}} \) in this paper.

Table I shows the data sets used in this study, which are available from the KEEL dataset repository [14]. All attributes were normalized into \([0, 1]\) in our computational experiments. The average results over 50 runs are shown in the next subsection.

**TABLE I. TWENTY DATA SETS USED IN THIS STUDY**

<table>
<thead>
<tr>
<th>Data</th>
<th>Patterns</th>
<th>Attributes</th>
<th>Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Australian</td>
<td>690</td>
<td>14</td>
<td>2</td>
</tr>
<tr>
<td>Bal</td>
<td>625</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Bupa</td>
<td>345</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>Contraceptive</td>
<td>1473</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>Flare-solar</td>
<td>1066</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>Glass</td>
<td>214</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Hayes-roth</td>
<td>132</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Heart</td>
<td>270</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>351</td>
<td>34</td>
<td>2</td>
</tr>
<tr>
<td>Newthyroid</td>
<td>215</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Pima</td>
<td>768</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>Saheart</td>
<td>462</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>Sonar</td>
<td>208</td>
<td>60</td>
<td>2</td>
</tr>
<tr>
<td>Spectfheart</td>
<td>267</td>
<td>44</td>
<td>2</td>
</tr>
<tr>
<td>Tae</td>
<td>151</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Vehicle</td>
<td>846</td>
<td>18</td>
<td>4</td>
</tr>
<tr>
<td>Vowel</td>
<td>990</td>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>Wdbc</td>
<td>569</td>
<td>30</td>
<td>2</td>
</tr>
<tr>
<td>Wine</td>
<td>178</td>
<td>13</td>
<td>3</td>
</tr>
<tr>
<td>Wisconsin</td>
<td>683</td>
<td>9</td>
<td>2</td>
</tr>
</tbody>
</table>

**B. Experimental Results**

We observed the synergy effect of two modifications on the search ability of our Michigan-style FGBML for 13 data sets: Bal, Flare-solar, Glass Hayes-roth, Ionosphere, Newthyroid, Pima, Saheart, Sonar, Vehicle, Wdbc, Wine, Wisconsin. Due to the page limitation, we show the training data accuracy and the test data accuracy for some data sets in Figs. 2-10. The red bar and line represent the result by the original FGBML (i.e., \( w_{\text{pen}} = 0 \) and \( H = 1 \)). The blue bar represents the best classification rate among all specifications.

Fig. 2 shows the results on the Bal data set. Better classification rates for both training and test data sets were obtained by the two modifications around \( w_{\text{pen}} = 1 \) and \( H = 3 \). We also observed that large \( w_{\text{pen}} \) and \( H \) led to the deterioration of the search ability of our FGBML. The similar synergy effects were observed for the glass data, the Pima data (Fig. 6), the Vehicle data (Fig. 8) and the Wisconsin data (Fig. 10). Especially, for the Pima data set, we observed a large improvement by the modified fitness function with a penalty term (i.e., from 69.69% to 75.61% for the test data).

Fig. 3 shows the results on the Flare-solar data set. The best training data accuracy was obtained by \( w_{\text{pen}} = 10 \) and \( H = 4 \), while the best test data accuracy was obtained by \( w_{\text{pen}} = 5 \) and \( H = 20 \). Although the synergy effects were observed, the effects of the modifications were different between the training data and the test data. We observed the same effects for the Hayes-roth data, the Saheart data (Fig. 7), the Sonar data, and the Wine data. Interestingly, for the Saheart data, even if we choose the best specifications with respect to the training data accuracy, the test data accuracy is clearly better than the original FGBML.

Fig. 4 shows the results on the Ionosphere data set. The highest classification rates on both training and test data sets were obtained by a large penalty \( (w_{\text{pen}} = 10) \) and five patterns \( (H = 5) \). The similar synergy effects were observed for the Newthyroid data (Fig. 5), and the Wdbc data (Fig. 9). We should examine larger penalty values for these data sets.

We observed the positive effect of one of two modifications for four data sets: Bupa (Fig. 11), Heart, Tae, Vowel (Fig. 12). Only the modified fitness function with a penalty term worked well for the Bupa data, while only the heuristic rule generation with multiple patterns worked well for the Heart data, the Tae data, and the Vowel data.
Fig. 3. Error rates for the Flare-solar data set.

Fig. 4. Error rates for the Ionosphere data set.

Fig. 5. Error rates for the Newthyroid data set.

Fig. 6. Error rates for the Pima data set.
Fig. 7. Error rates for the Saheart data set.

(a) Training data

(b) Test data

Fig. 9. Error rates for the Wdbc data set.

(a) Training data

(b) Test data

Fig. 8. Error rates for the Vehicle data set.

(a) Training data

(b) Test data

Fig. 10. Error rates for the Wisconsin data set.

(a) Training data

(b) Test data
We observed the negative or slightly positive effect of two modifications for three data sets: Australian, Contraceptive, Spectfheart. Fig. 13 shows the results on the Australian data. Slightly better results on the training data were obtained when we specified $w_{pen}$ and $H$ as 1 and 0, the original FGBML was the best with respect to the test data.

Table II shows the comparison of the test data accuracy between the original Michigan-style FGBML (original) and the proposed FGBML with the best specification of $w_{pen}$ and $H$ (proposed). The best specification was selected comparing the classification rates on the training data. A better result for each data set is highlighted by bold face. To confirm the statistical difference, we performed the Wilcoxon signed-rank test [15] to compare between the original FGBML and the proposed FGBML with two modifications over 20 data sets. The $p$-value was 0.037. Thus, we can say that the difference between the original FGBML and the proposed FGBML was statistically significant with a significant level 0.05.

As shown in Figs. 3 and 7, for some data sets, the effects of two modifications were different between the training data accuracy and the test data accuracy. Thus, it is very difficult to select appropriate specification of $w_{pen}$ and $H$ from the results on the training data. For those data sets, nested cross validation or repeated-double cross-validation [16] should be applied to find an appropriate specification.
TABLE II. COMPARISON OF TEST DATA ACCURACY (%) BETWEEN THE
CONVENTIONAL METHOD AND THE PROPOSED METHOD WITH BEST TRAINING
DATA ACCURACY.

<table>
<thead>
<tr>
<th>Data</th>
<th>Original</th>
<th>Proposed</th>
<th>Best ($w_{pen}$ $H$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Australian</td>
<td>86.638</td>
<td>85.913</td>
<td>(1.1)</td>
</tr>
<tr>
<td>Bal</td>
<td>83.837</td>
<td>85.566</td>
<td>(1.4)</td>
</tr>
<tr>
<td>Bupa</td>
<td>58.449</td>
<td>67.157</td>
<td>(2.1)</td>
</tr>
<tr>
<td>Contraceptive</td>
<td>50.686</td>
<td>50.876</td>
<td>(0.2)</td>
</tr>
<tr>
<td>Flare-solar</td>
<td>66.750</td>
<td>66.113</td>
<td>(0.4)</td>
</tr>
<tr>
<td>Glass</td>
<td>64.303</td>
<td>65.927</td>
<td>(1.2)</td>
</tr>
<tr>
<td>Hayes-roth</td>
<td>75.356</td>
<td>76.051</td>
<td>(1.2)</td>
</tr>
<tr>
<td>Heart</td>
<td>81.630</td>
<td>81.556</td>
<td>(2.4)</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>89.176</td>
<td>91.008</td>
<td>(10.5)</td>
</tr>
<tr>
<td>Newthyroid</td>
<td>87.649</td>
<td>95.173</td>
<td>(10.2)</td>
</tr>
<tr>
<td>Pima</td>
<td>69.688</td>
<td>75.054</td>
<td>(2.2)</td>
</tr>
<tr>
<td>Saheart</td>
<td>66.552</td>
<td>69.728</td>
<td>(2.2)</td>
</tr>
<tr>
<td>Sonar</td>
<td>78.362</td>
<td>79.710</td>
<td>(3.5)</td>
</tr>
<tr>
<td>Specfheart</td>
<td>79.416</td>
<td>78.652</td>
<td>(5.4)</td>
</tr>
<tr>
<td>Tae</td>
<td>55.667</td>
<td>52.467</td>
<td>(1.2)</td>
</tr>
<tr>
<td>Vehicle</td>
<td>65.184</td>
<td>65.233</td>
<td>(1.5)</td>
</tr>
<tr>
<td>Vowel</td>
<td>50.848</td>
<td>53.677</td>
<td>(0.3)</td>
</tr>
<tr>
<td>Wdbc</td>
<td>93.247</td>
<td>95.041</td>
<td>(10.3)</td>
</tr>
<tr>
<td>Wine</td>
<td>96.039</td>
<td>94.595</td>
<td>(5.1)</td>
</tr>
<tr>
<td>Wisconsin</td>
<td>96.344</td>
<td>96.692</td>
<td>(3.3)</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>74.790</td>
<td>76.309</td>
<td>-</td>
</tr>
</tbody>
</table>

V. CONCLUSION

In this paper, we examined the synergy effects of our Michigan-style FGBML with two modifications on the search performance. One modification is the use of multiple patterns for the heuristic rule generation. The other is the modified fitness calculation with a penalty term. The experimental results showed that the appropriate specifications of $w_{pen}$ and $H$ depended on the data sets. We observed the synergy effects of these two modifications or the positive effects by one of them for many data sets.

We need more detailed analysis on these modifications. We have to clarify what kind of data sets these modifications work well and how to specify appropriate parameters for each data.

The two modifications in this paper (i.e., rule generation from multiple patterns and fitness calculation with penalty) can be incorporated into various GBML methods XCS, UCS, fuzzy UCS, and so on [17]-[20]. This is also left for a future research topic.

REFERENCES


