PARALLEL GENETIC ALGORITHM WITH ADAPTIVE GENETIC PARAMETERS TUNED BY FUZZY REASONING

YOICHIRO MAEDA
Dept. of Human and Artificial Intelligent Systems
Faculty of Engineering, University of Fukui
3-9-1, Bunkyo, Fukui, 910-8507 Japan
maeda@ir.his.fukui-u.ac.jp

QIANG LI
Dept. of Human and Artificial Intelligent Systems
Graduate School of Engineering, University of Fukui
3-9-1, Bunkyo, Fukui, 910-8507 Japan
qli@ir.his.fukui-u.ac.jp

ABSTRACT. Genetic algorithms (GAs) have several problems, the important of which is that the search ability of ordinary GAs is not always optimal in the early and final stages of the search because of fixed GA parameters. Therefore, we have already proposed the fuzzy adaptive search method for genetic algorithms which is able to tune the genetic parameters according to the search stage by the fuzzy rule.

In this paper, a fuzzy adaptive search method for parallel genetic algorithms is proposed, in which the high-speed search ability of fuzzy adaptive tuning by FASGA is combined with and the high-quality solution capacity of parallel genetic algorithms. The proposed method offers improved search performance, and produces high-quality solutions. Simulations are performed to confirm the efficiency of the proposed method, which is shown to be superior to both ordinary and parallel genetic algorithms.

Keywords: Parallel Genetic Algorithm, Migration, Fuzzy Reasoning, Adaptive Search.

1. Introduction. Genetic algorithms (GAs) are search algorithms based on the mechanics of natural selection and natural genetics [1, 2]. GAs can be applied to several types of optimization problems by encoding design variables to individuals. However, GAs also have several problems, the important of which is that the search ability of ordinary GAs is not always optimal. This is particularly important in the early and final stages of the search, and is due to the GA parameters (crossover rate, mutation rate etc.) being fixed. The large number of iterations required to find a solution using GAs also limits their utility. Thus, many types of modified GAs have been proposed in an attempt to improve the performance of this potentially useful technique.

Lee et al. proposed a method of dynamic control of GA parameters based on fuzzy logic techniques [3]. In this method, the population sizes, and crossover and mutation rates are decided from average and maximum fitness values and differentials of the fitness value by fuzzy rules. In our laboratory, a fuzzy adaptive search method for genetic algorithms (FASGA) has been developed as a modified GA [4, 5, 6]. By this method, efficient searching is realized by describing fuzzy rules to tune the GA parameters (crossover and mutation rates) based on maximum and average fitness values according to the search stage.

Parallel GA methods have also been proposed as effective methods for finding high-quality solutions using GAs [7]. In parallel methods, the total population is divided into independent sub-populations called islands. Three distribution models have been proposed: a master-slave model, a coarse-grained model (island model) [8], and a fine-grained model (cellular model) [9]. In the present research, the island model
is employed so as to avoid the propagation of local minimum solutions in a whole population, thereby yielding a high-quality solution. After a predetermined number of generations (the migration interval), genes are moved to another island at a predetermined migration rate defined as the number of genes migrating per migration event. Although the population size of each island is smaller than that of a GA to allow local solutions to be reached quickly, the existence of islands and the operation of migration ensure that the variety of solutions is kept comparatively high in this type of parallel genetic algorithm (PGA). Generally, PGAs are therefore capable of higher-quality solutions than ordinary GAs.

The disadvantage of PGAs is that parallel processing cannot always be used effective because the migration rate of PGA is a constant. Many modified methods have been proposed to overcome this problem, including a distributed GA with randomized migration rate method [10], and PGA with distributed environment scheme [11].

In the present study, a fuzzy adaptive search method for parallel genetic algorithms (FASPGA) is proposed, combining FASGA with an island-model PGA. It is expected that this FASPGA method will overcome both of these problems, the sub-optimality of GA search, and the effective utilization of parallel processing [12]. FASPGA is a PGA method offers both fast search ability and high-quality solutions, with tuning achieved by not only the crossover and mutation rates but also the migration rate via fuzzy reasoning. The fuzzy adaptive control of the migration rate of the PGA by evaluating the evolutionary degree for each island is the characteristic feature of this method. Computer simulations of the optimization of the Rastrigin function are performed to confirm the efficiency of the FASPGA approach.

2. The FASPGA Method. The proposed method combines FASGA, which allows the genetic parameters to be tuned according to the search stage based on fuzzy rules, with a PGA, which produces high-quality solutions. FASPGA uses fuzzy rules to improve both the search performance of each sub-population by tuning the genetic parameters in each sub-population for every generation, and the search performance of the whole population by tuning the migration rate. The FASPGA method is therefore expected to achieve faster searches and afford higher-quality solutions.

2.1. General Concept. The setting of genetic parameters and crossover and mutation rates influences the behavior and performance of GAs greatly. These parameters relate directly to the performance of the algorithm: the higher the crossover rate, the faster the production of new individuals, but the more easily the genetic schema is broken, causing the construction of individuals with high fitness value to fail quickly. If the crossover rate is too low, the search will be so slow as to become stationary. Similarly, if the mutation rate is too small, the production of new individuals will be difficult. However, a high mutation rate causes the GA to become to a pure stochastic search algorithm. Finding robust genetic operators or parameter settings that allow the premature convergence problem to be avoided in any problem is not a trivial task, as the interaction of these settings with GA performance is complex and optimal values are often problem-dependent.

Many adaptive techniques have been suggested in order to adjust the GA configuration depending on parameters associated with GA performance. The FASGA method proposed by our laboratory is such an adaptive technique, in which the genetic parameters, including the crossover and mutation rates, are tuned based on fuzzy rules according to the search stage. In the early stage, the crossover rate should be small, the mutation rate should be large so as to maintain the species diversity. On the contrary, the mutation rate should be small in order to avoid breaking the schema of excellent individuals, and the crossover rate should be large for obtaining the good individuals quickly in final stage.

2.2. Fuzzy Rule in FASPGA. In this study, the parameters of FASPGA included crossover rate ($r_{c_i}$), mutation rate ($r_{m_i}$) and migration rate ($r_{e_i}$), are all decided by the fuzzy rule (see Figure 1). And the fuzzy rule is based on two variables, the average fitness value ($f_{a_i}$) and the difference between the maximum and average fitness value ($f_{m_i} - f_{a_i}$) in each sub-population $i$ (island $i$). Because of depending on these two variables ($f_{a_i}, f_{m_i} - f_{a_i}$), we are able to understand the states of each island that it is in the early stage or final stage. So, the fuzzy rule with three types of parameters (crossover rate, mutation rate and migration rate) is formed according to these two values, $f_{a_i}$ and ($f_{m_i} - f_{a_i}$).
Next, we explain about the fuzzy rule in FASPGA. To understand easily, let us take examples. The FS means the average fitness in an island is small, and it also means this island is in the early searching stage. The same to the FL, it means the average fitness in an island is large, and also means this island is in the final searching stage. On the other hand, the DS means the difference between the maximum and average fitness value \((f_m - f_a)\) in an island is small, at the same time it also implicate the individuals in this island is rather compact. The DL is just contrary. As above those, according to different states, different parameters are set such as CVS, MVL, and EVL are set in the state of FS and DS. By this way, we solved the problem referred in Section 1, which the search ability of the ordinary GA is not always optimal specially in the early and final search stage.

In FASPGA, the crossover rate \(r_{c_i}\), the mutation rate \(r_{m_i}\) and the migration rate \(r_{e_i}\) are not fixed. We realize to tune the parameters value of each island by using fuzzy inference. In this research, the membership function in \(IF\) part of fuzzy inference in FASPGA we proposed is as same as FASGA. It is composed of the average fitness value \(f_a\) and the difference between the maximum and average fitness value \((f_m - f_a)\). However, FASPGA add a parameter called the migration rate in \(THEN\) part. The resulting output is calculated by the weighted average based on the firing strength. Membership functions in \(IF\) part and singletons in \(THEN\) part are shown in Figure 1.

2.3. Migration Process in FASPGA. In fact, the migration is an operation that some individuals are selected to move from one island to another. By the migration, the better individuals are able to be
spread in all population quickly, and enhance the precision of solution. The migration of individuals from one island to another is controlled by these parameters: (a) a migration rate; (b) a migration interval; and (c) the topology that defines the connections between islands. Migration rate means the ratio of migrated per all individuals in a sub-population. In this FASPGA, because the migration rate is tuned by fuzzy rule, so the migration interval is not needed any more. As a migration topology, we used Random Ring model.

2.3.1. Decision of Migration Rate. To our knowledge, it is difficult to decide the migration rate properly, but it also is very important because it concerns the performance of PGA directly. Generally, the individuals of migration are almost the best individuals in each sub-population, so if the migration is frequent, it advantages to spread the advance individuals in all population and improve the speed of convergence. However, at the same time it causes the decrease of population diversity, it disadvantages to explore different regions of the search space.

On the other hand, the migration rate is a constant in ordinary PGA. In the other words, individuals of each sub-population are migrated in the same size, regardless of the sub-population with the delayed evolutionary condition or with the advanced evolutionary condition. This is not obviously effective by using parallel processing. It disadvantages to improve the efficiency of PGA as spreading of individuals with the delayed evolutionary condition.

Therefore, in FASPGA proposed in this paper, we didn’t set migration rate to a constant and set migration rate in a range (see Figure 2). It is tuned by fuzzy rule according to states of each sub-population. It makes the migration process more efficient. In this method, the migration rate \( r_c \) is decided by fuzzy rule based on the average fitness value \( f_a \) and the difference between the maximum and average fitness value \( (f_m - f_a) \) in each island \( i \). In the process of the migration, some individuals in sub-population with the advanced evolutionary condition are easy to be spread in all population. On the contrary, some individuals in sub-population with the delayed evolutionary condition are difficult to be spread in whole population under the tuning of fuzzy rule. So the fuzzy rule plays a good part in guiding the evolitional direction for improving the quality of solution effectively.

![Figure 2. Migration Process with Random Ring Model in FASPGA](image)

2.3.2. Selection of Migration Individuals. We used the roulette wheel selection as the selection method to select migration individuals. Probability of the roulette wheel selection for selecting individuals with high fitness value is used high in the sender island and low in the receiver island as shown in the following equations. In these equations, \( p \) means the number of individuals of each-population.
**Probability of Individual Selection in Sender Island:**

\[
\frac{f_i}{\sum_{i=0}^{p} f_i}
\]

**Probability of Individual Selection in Receiver Island:**

\[
\frac{f_{m_j} - f_j}{\sum_{j=1}^{p}(f_{m_j} - f_j)}
\]

2.4. **Algorithm Flow of FASPGA.** At first, the initial individuals are generated at random. Then the fitness value of each individual is calculated. Next, the initial population is divided into \(n\) sub-populations (islands). After the selection by using the roulette wheel selection method, the average fitness value \(f_{ai}\) and the maximum fitness value \(f_{mi}\) are calculated in each sub-population \((i=1, 2, \ldots, n)\). By estimating a progress degree of the evolution with the average fitness value \(f_{ai}\) and the difference between the maximum and average fitness value \((f_{mi} - f_{ai})\), the migration rate \(r_e\) in each sub-population are decided by fuzzy rule. The migration rate \(r_e\) is bigger, the size of individuals of the migration is larger. And the migration is executed with the random ring model. Before the operation of crossover and mutation, \(f_{ai}\) and \(f_{mi}\) are calculated once more. Because the fuzzy rule depends on current \(f_{ai}\) and \((f_{mi} - f_{ai})\), the crossover rate \(r_c\) and mutation rate \(r_m\) must be successfully tuned. Finally, the terminate condition

![Algorithm Flowchart of FASPGA](image)

**Figure 3. Algorithm Flowchart of FASPGA**
of the evolution is checked. If it is contented then terminate the evolution, or else return to operation of selection, and execute once more in the same step. We can regard this process as FASGA algorithm applied to each sub-population.

Algorithm flowchart of FASPGA proposed in this paper is shown in the Figure 3. Tuning processes of the crossover rate $r_{c_i}$, the mutation rate $r_{m_i}$ and the migration rate $r_{e_i}$ in each island by fuzzy reasoning are executed in the dotted line area.

3. Simulation. The computer simulation was performed in this research to confirm the efficiency of FASPGA proposed in this paper. In this section, we explain the precondition of simulation, simulation method and report the results of simulation.

3.1. Precondition of Simulation. In this simulation, we used Rastrigin function as a test function to confirm the efficiency of FASPGA. Rastrigin function is the $n$-dimensional function with multiple peaks as shown in the equation (3) and the function which has Lattice-shaped semi-optimum solutions around an optimum solution and is no dependence between design parameters. As the simplest example, 2-dimensional Rastrigin function is shown in Figure 4.

$$F_{Rastrigin}(x) = 10n + \sum_{i=1}^{n} \{x_i^2 - 10\cos(2\pi x_i)\}$$

$$(-5.12 \leq x_i < 5.12)$$

$$\min(F_{Rastrigin}(x)) = F(0,0,\ldots,0) = 0$$

**Figure 4. Overview of 2-Dimensional Rastrigin Function**

For example in Figure 4, Rastrigin function has two design parameters which are shown as two horizontal axes and the fitness value as the vertical axis. The optimum solution in this function is a point with zero fitness value, that is, a bottom of the valley in the origin of Figure 4.

In this simulation, we used 20 design parameters and standard binary coding. The elitist strategy is exploited in GA, PGA and FASPGA with one elite. The way of selecting elite is that selects the fittest individual in the island as the elitist individual and the way of returning elite is that replaces the worst individual in the island. Figure 5 shows the process of elitist strategy used in this simulation. The value of default parameters in GA, PGA and FASPGA are shown in Table 1. These parameters were selected after many simulations for GA and PGA. We used the best value of proper parameters in our simulations which we executed changing each parameter; crossover rate, mutation rate and migration rate. The parameters of fuzzy inference in FASPGA are set as Figure 6.

3.2. Simulation Method. In this research, two cases of simulations with the following range of parameter values were executed in GA, PGA and FASPGA. The simulations were carried out in a partial fashion, exploring the effect of varying one parameter while fixing the other at their default values.

- Case 1 - Island population size: 10, 30, 50, 100, 150, 200; Number of islands: 20 (fixed);
- Case 2 - Island population size: 10 (fixed); Number of islands: 10, 50, 100, 150, 200, 400.
PARALLEL GENETIC ALGORITHM WITH ADAPTIVE GENETIC PARAMETERS

![Diagram showing the process of Elitist Strategy](image)

**Figure 5.** Process of Elitist Strategy

<table>
<thead>
<tr>
<th>Parameters</th>
<th>GA</th>
<th>PGA</th>
<th>FASPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Island Population Size</td>
<td>Case 1: 10, 30, 50, 100, 150, 200</td>
<td>Case 2: 10</td>
<td></td>
</tr>
<tr>
<td>Number of Islands</td>
<td>–</td>
<td>Case 1: 20</td>
<td>Case 2: 10, 50, 100, 150, 200, 400</td>
</tr>
<tr>
<td>Generations</td>
<td>1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chromosome Length (L)</td>
<td>200</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Selection Method</td>
<td>Roulette Wheel</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Crossover Rate</td>
<td>0.6</td>
<td>0.6</td>
<td>Tuning by Fuzzy Rule</td>
</tr>
<tr>
<td></td>
<td>(One Point Crossover)</td>
<td>(One Point Crossover)</td>
<td>(One Point Crossover)</td>
</tr>
<tr>
<td>Mutation Rate</td>
<td>1/L</td>
<td>1/L</td>
<td>Tuning by Fuzzy Rule</td>
</tr>
<tr>
<td></td>
<td>Tuning by Fuzzy Rule</td>
<td>Random Ring</td>
<td>Random Ring</td>
</tr>
<tr>
<td>Migration Method</td>
<td>–</td>
<td>Random Ring</td>
<td></td>
</tr>
<tr>
<td>Migration Rate</td>
<td>–</td>
<td>0.5</td>
<td>Tuning by Fuzzy Rule</td>
</tr>
<tr>
<td>Migration Interval</td>
<td>–</td>
<td>5 (generations)</td>
<td>Changed</td>
</tr>
</tbody>
</table>

In the first simulation (Case 1), we compare the results of simulation with GA, PGA and FASPGA in the different population size. It is in order to confirm the performance of FASPGA subject to different population size. The second simulation (Case 2) is preformatted in the small population size with GA, PGA and FASPGA. The purpose of this simulation is confirming the performance of FASPGA in the case of that the hardware circumstance is finite. In this case we have to utilize small individuals size or short generations to obtain the optimum solution in short time.

3.3. **Simulation Results.** We performed the optimization simulation using Rastrigin function and compared the result of simulation based on maximum fitness value. All figures in this section display maximum fitness on the y-axis, and generations on the x-axis. For the requirement of our computer program, we modified the Rastrigin function value \( F_{\text{Rastrigin}} \) as following equation. So the optimum solution value is also changed to 810 because it is equal to the worst value of the Rastrigin function.

\[
F_{\text{max_fitness}} = 810 - \min(F_{\text{Rastrigin}})
\]

The results of the simulation Case 1 are shown in Figure 7. These figures show the maximum fitness values in case that islands have various population size and the number of islands are fixed on 20. From these figures, we confirmed that the performance of FASPGA is the best, and GA is extremely the worst.
On the search capability of the early search stage, FASPGA is almost the best in any cases of population size. There is only tiny difference between PGA and FASPGA in the final search stage, but in case of over 50 and 100 individuals, only FASPGA has already obtained the best solution in about 400 generations. FASPGA has the best performance in these cases and becomes worse in case of over 150 individuals. So, in the simulation of the fixed island size, we confirmed that 50 or 100 individuals per an island are the best selection as the island population size.

Figure 8 shows the results of the simulation Case 2. In this simulation, we have the island population size fixed in 10 individuals and the number of islands is tried in various island sizes. The performance of GA is also the worst in this simulation. And FASPGA has the better performance than PGA in the early search stage, but it is small. In case of 10 islands, PGA and FASPGA have the similar performance. However, in case of over 50 islands, FASPGA is better than PGA in final search stage. In case of over 100 islands, FASPGA found the optimal solution between 100 and 200 generations. Specially, in case of 100 islands, PGA could not find the optimal solution until 1000 generations. So, in the simulation of the fixed island population size, we confirmed that 100 islands are the best selection as the minimum number of islands. However, there is not nearly difference in the final search stage. In addition, we could also find the difference between PGA and FASPGA becomes small along with the island size becoming small.

Figure 9 shows the results of the simulation with the best parameters selected in Case 1 and Case 2. In this simulation, FASPGA also has the best performance. However, FASPGA in case of over 100 islands in the simulation Case 2 obtained the optimal solution at the earliest generations. We consider the case of 100 islands with 10 individuals is the best parameters.

3.4. Remarks. We think the reason why FASPGA has good performance in the early search stage of both first and second simulation is that the speed-up of evolution was realized by the effect of tuning genetic parameters by using fuzzy reasoning rules.

Furthermore, in the simulation Case 1 and Case 2, FASPGA finally obtained the high-quality optimal solution as compared with PGA. We think this performance was able to realize by maintaining population diversity tuning the migration rate in each search stage by fuzzy rules. In addition, FASPGA seems to be able to realize the better performance in 50 to 100 of the island population size and the large number of islands over 100.
However, in the simulation Case 2, the difference between FASPGA and PGA became small in case of small island size. We consider that a reason of causing this state is that total migration individuals also decreased because the number of islands decreased. Thus lead the tuning capability of fuzzy rule in the migration rate to be weakened, because the island size is too small to find obvious difference in the migration individual size between large migration rate and small migration rate.

As a result, totally, we could confirm that FASPGA with large number of islands and limited individuals is able to obtain the optimum solution faster and higher quality than PGA.

4. Conclusions. A fuzzy adaptive search method for parallel genetic algorithms was proposed, in which the genetic parameters are adaptively tuned by fuzzy rules in accordance with the search stage. This method combines the fast search ability of a fuzzy adaptive search method with the capacity of parallel genetic algorithms. The FASPGA method therefore offers improved search efficiency and higher-quality solutions.

The performance of FASPGA was evaluated through optimization using the Rastrigin function with a range of parameter settings and comparison with the results for an ordinary GA and PGA. The FASPGA method was confirmed to reach the optimum solution faster and to produce higher-quality solutions than a PGA in the case of a large number of islands and limited individuals. These results suggest that a large number of islands is required to obtain good solutions. In the case of small populations, FASPGA also provided good performance in the early search stage, but offered no improvements in the final search stage using a small island population and small island size. This result demonstrates that the island population size and the number of islands have a substantial effect on the performance of FASPGA when the total population size is small.

Future work will include further research to confirm the performance of FASPGA using other testing functions, and the consideration of new optimum parameters. The relatively poor performance of FASPGA for a small populations will also be addressed. The FASPGA method is currently being investigated for application to motion learning for a robot manipulator.

REFERENCES


**Figure 7.** Simulation Results (Case 1)
Figure 8. Simulation Results (Case 2)
Figure 9. Simulation Results

a) 100 islands with 50 individuals

b) 100 islands with 100 individuals