A New Model of Distributed Genetic Algorithm for Cluster Systems: Dual Individual DGA

Tomoyuki HIROYASU
tomo@is.doshisha.ac.jp

Mitsunori MIKI
mmiki@mail.doshisha.ac.jp

Masahiro HAMASAKI
ham102@iname.com

Yusuke TANIMURA
tanisuke@mikilab.doshisha.ac.jp

Department of Knowledge Engineering and Computer Sciences
Doshisha University
1-3 Tatara Miyakodani Kyotanabe-shi, Kyoto, 610-0321, JAPAN

Abstract *In this study, a new model of distributed genetic algorithm (DGA) for cluster systems is proposed. That is called Dual Individual Distributed Genetic Algorithm: DuDGA. DGA is a very good parallel model of genetic algorithms, because the necessary network traffic is not so heavy. On the other hand, DGA needs additional parameters to simple GA such as migration rate and migration intervals. In the DuDGA, each island of DGA has only two individuals. Because of this constitution, Necessary parameters decrease markedly. At the same time, the searching ability improves and the network traffic is also decrease. These characteristics are suitable for the PC cluster systems. Through the typical test functions, these characteristics are confirmed and discussed.*

Keywords: Distributed Genetic Algorithm, Parallel Processing, Distributed Processing

1 Introduction

Genetic Algorithm (GA) that mimics the mechanism of the creature's heredity and the evolution is one of the optimization tools [1]. Because GA is one of the multi point searching and the stochastic algorithm, GA can find the optimum even when there are several peaks of local minimums in searching field. At the same time, GA can deal with not only the continuos searching domain but also the discrete searching domain. On the other hand, there are several problems in GAs. One of the biggest problems is the necessity of huge computational cost. To solve this problem, the operation of GA in parallel is one of the solutions. Distributed Genetic Algorithm (DGA) is one of the models of the parallel genetic algorithms [2]. In this model, the necessary network traffic is very small. Therefore, DGA is a suitable model for PC cluster systems.

In this paper, the DGA is extended and a new model of distributed genetic algorithm (DGA) for cluster systems is proposed. That is called Dual Individual Distributed Genetic Algorithm: DuDGA. In this model, there are only two individuals in each island. Because of this constitution, the number of setting parameters is decreased. At the same time, the searching ability is increased to compare with the normal DGA. In the implementation of the parallel of DuDGA, each processor has several numbers of islands. In each processor, DuDGA is performed. After some iterations, an island is exchanged. There are a lot of islands in DuDGA, but only one island is moved to the other processor. Therefore, the network traffic is very small. It can be said that DuDGA is very suitable to PC cluster systems.

The characteristics of DuDGA are examined through the numerical examples. By dappling DuDGA to typical test functions, it is found that DuDGA has high searching ability and high parallel efficiency on PC cluster systems.

2 Distributed Genetic Algorithm

Distributed Genetic Algorithm (DGA) is one of the models of parallel GAs [3, 4, 5]. In the DGA, the following steps are performed.

1. *Initialize :* The total population is divided into sub populations
2. *SGA :* SGA (or other GA) is performed in each island for several iterations.
3. *Migration :* After several iterations, some of the individuals are chosen and are moved to the other island. This operation is called migration. The iteration is called migration interval. The number of chosen individuals is determined by the number of individuals and the parameter of migration rate. After the migration, the process is returned to the SGA process.

In this model, usually, each island is allocated to a processor and the network communication does not happened frequently. Therefore, this model is very suitable model to the cluster parallel computer systems. Moreover, the DGA can derived the good solutions with smaller calculation cost compared to the one population model [6, 7]. Therefore, this model is useful not only in the parallel processing systems but also in the sequential processing systems.

The GAs have several parameters and these parameters should be determined by users. In the DGAs, moreover, users should determine the additional parameters: island number, migration rate and migration interval. Cuntu-Paz investigated the topologies, migration rates and multi population of DGA [8]. However, the problem of the parameter setting is still existed.

3 Dual Individual Distributed Genetic Algorithm

In this section, a new model of distributed genetic algorithm: Dual Individual Genetic Algorithm (DuDGA) is explained. At first, the over view of DuDGA is explained. Then, the implementation of parallel processing of DuDGA is explained.

3.1 Overview of DuDGA

In the DuDGA, there are only two individuals in each island. The concept is shown in Figure 5.

In the proposed DuDGA, the following operations are performed.

- The number of population in each island: There are only two.
- Selection: The individuals whose fitness values are best in the present generation and one previous generation are remained.
- Migration method: The migration individual is chosen randomly.
- Migration topology: It is a stepping stone. The destination of the migration is determined randomly at every migration event.

DGA can derived the better solutions with the smaller calculation costs, compared to the simple GA. However, in the DGA, there are several parameters. Since the parameters affected to the quality of the solutions, the users should determine these parameters carefully.

In DuDGA, most of the parameters are determined automatically, since there are only two individuals in each island. Such as,

- crossover rate: 1.0
- number of islands: total population size/2
- migration rate: 0.5 .

This is one of the advantages of the DuDGA. Users are free from some parameter settings.

3.2 Parallel Implementation of DuDGA

Since there are a lot of individuals in DuDGA, the parallel process of DuDGA is performed as follows.

1. The islands are divided into sub groups. Each group is assigned to one processor.
2. In each group, DuDGA is performed. In this step, the migration is occurred just in the group.
3. After some iteration, one of the islands in each group is chosen and is moved to the other group.
4. The step is returned to step 2.

Because just one island is chosen to move to the other group the network traffic is very small. The concept of this parallel implementation is shown in Figure 6. In this Figure, there is an example that has two processors.

4 Numerical Examples

4.1 Test Function and Used parameters

The discussion is performed through the following four types of numerical test functions.

$$F_1 = 10N + \sum_{i=1}^{N}\{x_i^2 - 10cos(2\pi x_i)\}$$
$$(-5.12 \leq x_i < 5.12) \quad (1)$$
$$F_2 = \sum_{i=1}^{N}\{100(x_{2i-1} - x_{2i}^2)^2 + (x_{2i} - 1)^2\}$$
$$(-2.048 \leq x_i < 2.048) \quad (2)$$
$$F_3 = 1 + \sum_{n=i}^{10} \frac{x_i^2}{4000} - \prod_{n=1}^{N} cos(\frac{x_i}{\sqrt{i}})$$
$$(-512 \leq x_i < 512) \quad (3)$$
$$F_4 = \sum_{i=1}^{N}(\sum_{j=1}^{i} x_j)^2$$
$$(-64 \leq x_i < 64) \quad (4)$$

The number of design variables (ND) and the number of bits (NB) of the test functions are summarized in Table 1.

Because the Rastrigin function (F_1) is the function whose design variable is independence respectively to the objective function, it is easy for GA to derive the solutions. On the other hand, the Rosenbrock function (F_2) and the Ridge function (F_4) are difficult for the GA to find the solutions because the design variables are not independent to the objective function. The Griewank function (F_3) is the problem of the difficult of the middle degree between F_1 and F_2.

Table 1: Test Functions

	Function Name	ND	NB
F_1	Rastrigin	20	200
F_2	Rosenbrock	5	50
F_3	Griewank	10	100
F_4	Ridge	10	100

Used parameters are summarized in Table 2. The results are derived with these parameters.

Table 2: Used Parameters

	DGA	DuDGA
Crossover rate	1.0	1.0
Population size	240	240
Mutation rate	$1/L$	$1/L$
Number of islands	4, 8, 12, 24	120
Migration rate	0.3	0.5
migration interval	5	5

L : Chromosome length

Besides these parameters, the algorithm is terminated, when the number of generation is over 5000. All of the results are the average of 20 trials.

In the DGA, there are several parameters. However, in the island of the DuDGA, because there are only two individuals, the parameters automatically determined except the population size and the migration interval.

4.2 Cluster system

In this paper, the simulation is performed on the PC cluster that is constructed with 16 PCs. The detail speck of this PC cluster system is shown in Table 3.

Table 3: Speck of 16 PC cluster system

CPU	Pentium II (400 MHz) $\times$ 16
Memory	128 MB
OS	Linux 2.2.10
Network	FastEthernet Switching HUB TCP/IP
Communication library	MPICH 1.1.2

This cluster system is a kind of Beowulf cluster and has normal networks. Therefore, when the network traffic is heavy, the parallel efficiency becomes very low.

4.3 Searching Ability of DuDGA

To find the searching ability of DuDGA, the covering rate and the number of evaluations are shown in Figure 1 and 2 respectively.

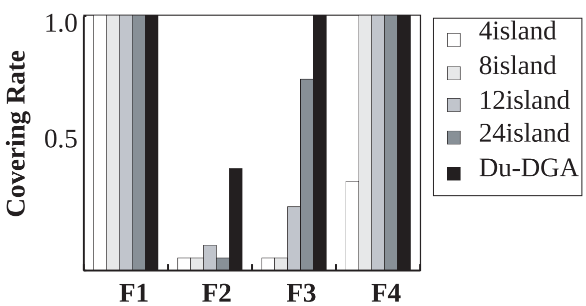

Figure 1: Covering Rate

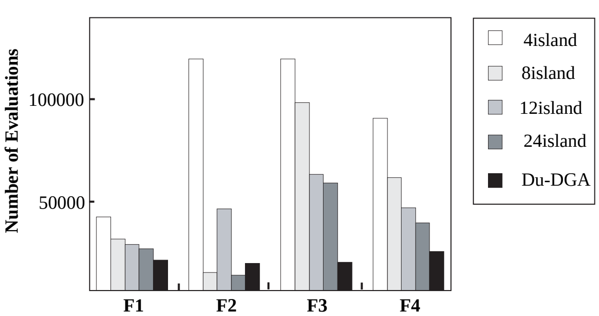

Figure 2: Number of Evaluations

The covering rate is the success rate of finding the optimum of each problem. Therefore, 1.0 is the best value for the covering rate. In GAs, usually, the evaluation part takes a lot of time. Therefore, the number of evaluation is very important index and the smaller values are better.

The DuDGA has good results in Figure 1 and 2 in every test function. It suggests that the number of island becomes bigger in DGA, the ability becomes better. We did not examine other model of DGA such as that whose population size of each island is 4 or 8. However, the DuDGA can be said the reasonable model, because users need not to determine the island number, the crossover rate and migration intervals.

Figure 7 and 8 are the history of evaluation values and the hamming distance to the optimum solution respectively. These are the results of the Rastrigin function (F_1).

The progress of the evaluation values of the DuDGA is not good in the first half part of the generation compared to the other DGAs. This is because the DuDGA is searching not in the local but in the global. On the other hand, in the latter half of the part, the values of the DuDGA are converged quickly and found the better solutions than those of the other DGAs. Therefore, in the latter half of the part the DuDGA is searching in local.

From these results, it can be said that the searching ability of DuDGA is made progress compared to the DGA.

4.4 Parallel Efficiency

To find the parallel efficiency, the speed up of DuDGA is shown in Figure 3. This is also a result of the Rastrigin function (F_1). The speed up means the rate with the calculation time of one processor and the multi processors. These results are derived by the DuDGA whose population size and the number of islands are the same but the number of groups is different.

From Figure 3, DuDGA can speed up in good rate and some reasons are considered as the followings. First of all, the network traf-

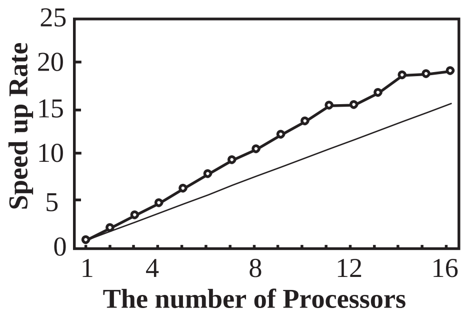

Figure 3: Speed Up (one process per one processor)

fic of DuDGA is very small so that there is no affection by the network traffic jam. The other reason is not the parallel effect but the distributed effect. The Distributed effect decreases the number of calculation.

In Figure 4, the calculation time that is derived by the PC cluster system is shown. This result is obtained in the situation where there are 16 groups of DuDGA. Therefore, when there are two processors, each processor has 8 groups. When there are 8 processors, each processor has 2 groups.

In Figure 4, it can also be said that the number of processor becomes bigger, the calculation time is smaller.

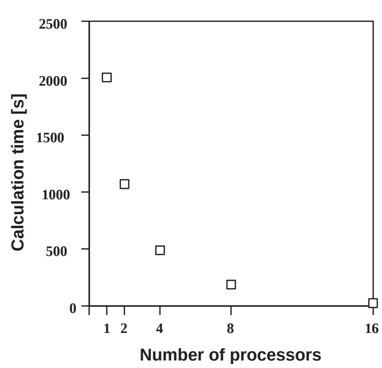

Figure 4: Calculation time (16 processes)

In this situation, there are several processes in one processor. This is unfavorable condition in Linux, because it cannot be obtained high efficiency by message passings on Linux. Therefore, when we need better solutions, we should use the threads in the processor or just use the same number of groups as the number of processors.

5 Conclusions

In this paper, a new model of distributed genetic algorithm is proposed. That is called Dual Individual Distributed Genetic Algorithm: DuDGA. In the DuDGA, there are only two individuals and this constitution reduces the number of necessary parameters of DGA. At the same time, the searching ability is also made progress compared to the DGA. The DuDGA is applied to finding the optimum of some typical test functions on the PC cluster systems. Through the numerical examples, the characteristics of DuDGA are confirmed. By dappling the DuDGA to the test functions, it is found that the DuDGA has high searching ability and its network cost is very small. Therefore, it can be concluded that DuDGA is a suitable model of GAs for PC cluster systems.

References

[1] D. E. Goldberg (1989) Genetic Algorithms in Search, *Optimization and Machine Learning*, Addison-Wesley, Reading, MA.

[2] J. Nang and K. Matsuo (1994). A Survey on the Parallel Genetic Algorithms, *J. SICE*, Vol. 33, No.6, 500-509.

[3] R. Tanese (1989). Distributed Genetic Algorithms, *Proc. of 3th Int. Conf. Genetic Algorithms*, 432-439.

[4] T. C. Belding (1995). The Distributed Genetic Algorithms Revised, Proc. of 6th Int. Conf. Genetic Algorithms,114-121.

[5] M. MIKI, T. HIROYASU and K. HATANAKA (1999). Parallel Genetic Algorithms with Distributed-Environment Multiple Population Scheme, *Proc. 3rd World Congress of Structural and Multidisciplinary Optimization (WCSMO)*, Vol. 1,186-191.

[6] V. S. Gordon and D. Whitley (1993). Serial and Parallel Genetic Algorithms as Function Optimizers, *Proc. of 5th Int. Conf. Genetic Algorithms* .

[7] D. Whitley, et. al. (1997). Island Model Genetic Algorithms and Linearly Separable Problems, *Proc. of AISB Workshop on Evolutionary Computation*

[8] E. Cantu-Paz (1999). Topologies, Migration Rates, and Multi-Population Parallel Genetic Algorithms, *Proceedings of GECCO 1999*,91-98.

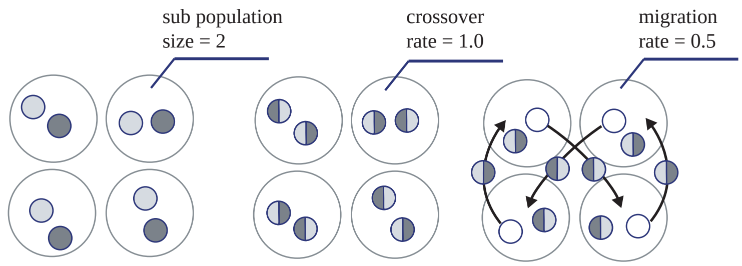

Figure 5: Dual Individual Distributed Genetic Algorithms

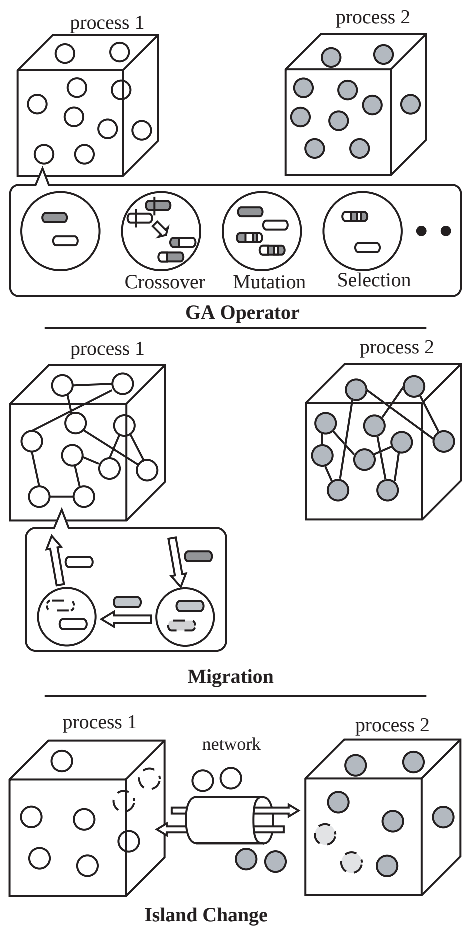

Figure 6: Parallel implementation of DuDGA

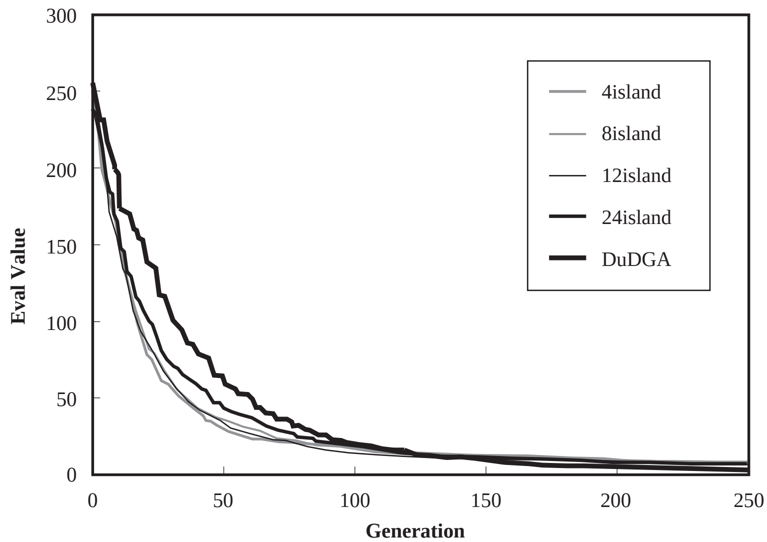

Figure 7: History of Evaluation values

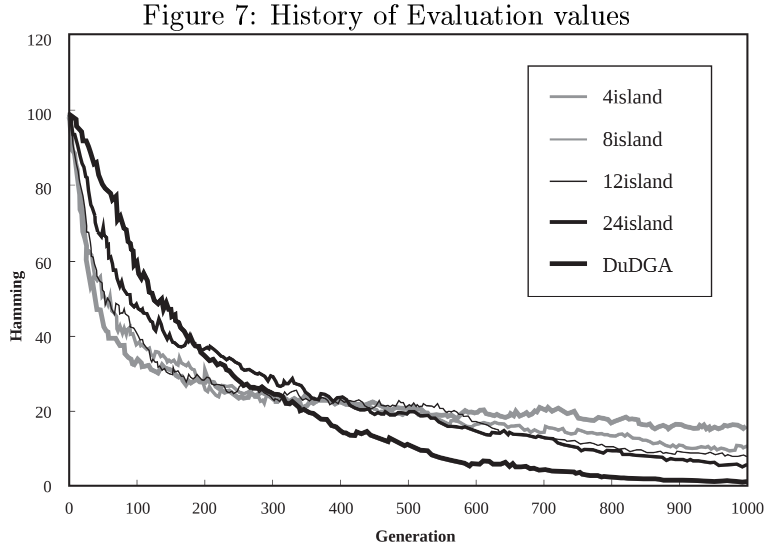

Figure 8: History of Hamming Distance to the Optimum Solution