

# A Parallel Genetic Algorithm and Its Variance Analysis for A New Multiple Knapsack Problem

Hayato Miyagi<sup>1</sup> and Morikazu Nakamura<sup>2</sup>

<sup>1</sup>Department of Information Engineering, University of the Ryukyus  
Okinawa 903-0213, Japan

E-mail : <sup>1</sup>haya2@ads.ie.u-ryukyu.ac.jp, <sup>2</sup>morikazu@ie.u-ryukyu.ac.jp

**Abstract:** This paper presents some results of experimental evaluation on the fitness varieties in parallel genetic algorithms based on tree topology migrations. The evaluation focuses on the relation between the solution quality and the fitness varieties. We perform the evaluation for the typical tree topologies; the star, line, and balanced binary tree with varying migration interval since the solution quality of the parallel genetic algorithm depends on the kind of the topology and migration interval. The results show that the line topology can keep the chromosome variety of the GA and obtain better quality of solutions since the topology has the longest distance between the root and the leaves.

## 1. Introduction

Genetic algorithms (GAs) are one of the meta-heuristics and used for combinatorial optimization [1]. GAs are often used for lots of real applications in engineering, operations research and computer science[2]. However, GAs generally require expensive computation costs because they perform searching iteratively by multiple points.

In order to overcome the weak point, parallel processing is easily considered. Many researchers have proposed parallel models of GAs[3]. These are classified into two categories: the centralized and distributed models. The centralized model aims for the speed up by evaluating chromosomes in parallel. On the other hand, the distributed model, well-known as *island model*, divides the chromosome pool into several sub pools and performs genetic operations independently in parallel[4]. However GA processes may exchange their chromosomes, usually elite chromosomes, each other some times, that is *migration*. The distributed model achieves not only the speedup but also keeping the chromosome variety. Keeping chromosome variety is the most important to obtain good quality of solutions in evolutionary computation. In [5], we proposed a parallel GA of the island model in which a tree topology is established on the computing resources before execution and migration can be taken place only from a node to the parent. We also investigated influence of the tree topology to the solution quality [5][6].

Furutani [7] applied successfully Fisher's theory in biology[8] to GAs, that is, the evolution speed is proportional to the chromosome variance.

## 2. Preliminaries

### 2.1 Parallel Genetic Algorithms

GAs are a population-based searching algorithm in which the initial population is randomly generated and predefined

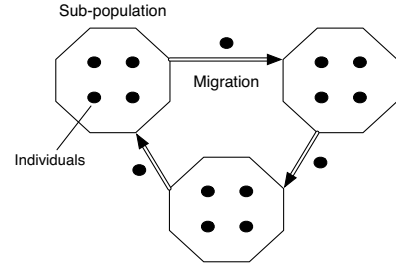


Figure 1. Distributed GAs

genetic operations are performed[1].

In the island model, a population is divided into sub-populations, islands. And then, the genetic operations are executed in each island independently. Some times an island may send chromosomes to its neighbor island, that is, migration. Figure 1 shows the image of the island model.

### 2.2 Multiple Knapsack Problem

Multiple knapsack problem (MKP) is one of the optimization combinatorial problem. The purpose of the MKP is to maximize the total profit calculated from the values of the selected objects for knapsacks.

$$\max \sum_{i=1}^m \sum_{j=1}^n p_j x_{ij} \quad (1)$$

$$\text{subject to : } \sum_{j=1}^n w_j x_{ij} \leq c_i \quad (2)$$

$$x_{ij} \in \mathcal{N}, i = 1, \dots, m, j = 1, \dots, n \quad (3)$$

In this research, we define a new MKP by introducing a constraint, called "concurrent selection". The constraint is such that the same number of objects is assigned to all the knapsacks, that is, all the knapsacks finally include the same number for each object. However, the profit and weight of an object are different if the knapsack is different.

Let us consider  $m$  knapsacks in which each knapsack  $i$  has capacity  $c_i$ , and  $n$  objects such that each object  $j$  has profit  $p_{ij}$  in knapsack  $i$ , and weight  $w_{ij}$ .  $x_j$  is the amount of object  $j$ .  $\mathcal{Z}'$  is a set of natural numbers.

$$\max \sum_{j=1}^n p_{ij}x_j, i = 1, \dots, m \quad (4)$$

$$\text{subject to: } \sum_{j=1}^n w_{ij}x_j \leq c_i, i = 1, \dots, m \quad (5)$$

$$x_j \in \mathcal{Z}', j = 1, \dots, n \quad (6)$$

### 3. Design of GAs and Parallelization

In order to obtain the solution, we convert the multi-objective optimization problem into the single-objective case by just adding the weighted sums of profits, where  $f_i(x)$  represents the  $i$ -th objective function that corresponds to the  $i$ -th knapsack and  $I_j$  is the weight representing its importance among the objective functions. The single objective form can be denoted by the following:

$$f(x) = I_1 \cdot f_1(x) + I_2 \cdot f_2(x) + \dots + I_n \cdot f_n(x) \quad (7)$$

A solution is arranged as a sequence of the amounts of included objects and an amount is described as a natural number. For genetic operations, we employ the elite and roulette selection, serial point crossover and 1-bit reversed mutation.

#### 3.1 Parallelization

In order to obtain good solution quality in reasonable time, we parallelize the GA based on the island model with tree topology migration. First of all, we establish a tree topology on the computing resources. In the topology, the nodes except for the leaves have children and the nodes except for the root has one parent. Each node in the topology corresponds to an island, that is, one GA process is independently run on an island. And then at some interval, a node sends chromosomes, updated elite chromosomes, to its parent, that is, *migration*. In our model, we call the interval *migration interval*. At every migration interval, a GA process checks whether or not the elite chromosome has been updated since the last migration and performs the migration if it is true. The procedure for the GA process is shown as follows:

```

0: procedure PGA;
1: begin
2:   Generation_no := 0;
3:   initialize a sub-population P;
4:   evaluation of the chromosome in P;
5:   while(termination condition is false)
6:     begin
7:       select randomly two parents and apply
         the one-point crossover;
8:       evaluation of the chromosomes in P;
9:       perform the selection;
10:      check the communication buffer;
11:      if(received chromosomes) begin
12:        replace the worst chromosome in P
          by the received one;
13:      endif;

```

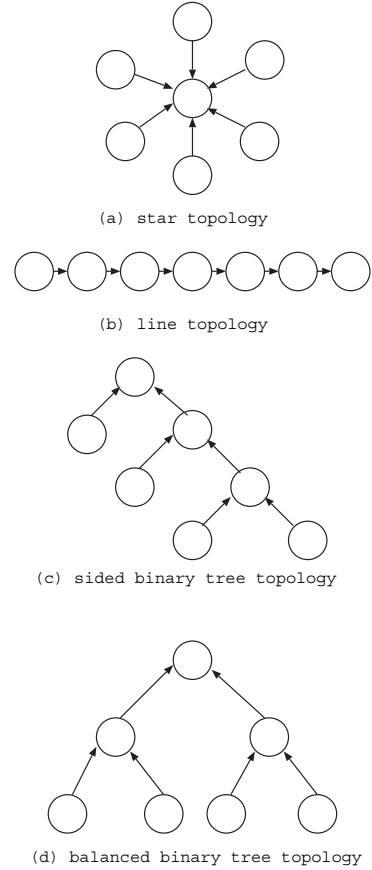


Figure 2. Tree Topology

```

14:   if(migration condition is satisfied)
15:     send the best chromosome to
       the parent;
16:   endif;
17: end while;
18: end

```

Note that effects of the cooperative evolution among islands depends on the topology where the migration is performed. Although there may exist many kinds of tree topology, we focus on the typical four topologies in this paper. Figure2 shows the four tree topologies with 7 node: the star, line, sided binary tree, balanced binary tree. Table 1 shows the comparison on the three characteristics. *Height* represents the height of the tree, *Leaves* the number of leaves, and *Descendant* the average number of descendants, respectively. From the table, we can observe that the star and line have the opposite characteristics and the others intermediate between the extremes.

## 4. Experimental Evaluation

### 4.1 Environment

We implemented the parallel GAs by using C language and a message passing library; LAM/MPI[9]. In the experiment, we use the parameters shown in Table 2 unless otherwise stated. The purpose is to investigate the effect of migra-

Table 1. Properties of topologies

Topology	Height	Leaves	Descendant
<i>star</i>	1	$n - 1$	$(n - 1)/n$
<i>balanced</i>	$\log \frac{(n+1)}{2}$	$(n + 1)/2$	$\frac{(n+1) \log(n+1)/4+2}{n}$
<i>sided</i>	$(n - 1)/2$	$(n + 1)/2$	$(n - 1)(n + 1)/4n$
<i>line</i>	$n - 1$	1	$(n - 1)/2$

Table 2. Parameter of Parallel GAs

Total number of individuals	300
Chromosome length	4
Number of items	187
Number of islands	25
Selection	Elite and Roulette selection
Crossover	One point crossover
Crossover rate	0.99
Mutation rate	0.00
Migration gap	100 generation
Migration rate	one individual
Topology	Line

tion by changing parameter on parallel GAs. Therefore, we do not use the mutation as a genetic operator.

We show the experimental results by calculating the average of twenty times runs.

#### 4.2 Topologies

First of all, we observe the solution quality and solution variance for the four topologies. Here the solution variance is calculated based on the Hamming distance from the solution 000...000.

Figure 3, 4 show the curves of the best solution and the variance of the root node for each topology. In Fig. 3, the line performed the best and then the sided binary, balanced binary, star did worst.

In Fig. 4, the solution variance of the line topology is small, but it is kept for longer time than the other tree topologies. On the other hand, the variance of the star is large at early stages, but it is drastically reduced within shorter time than the others.

In the line topology, one migration should not improve drastically solution quality but the variance can be kept for long time. On the other hand, in the star, one migration can improve solution quality well, but the variance should be reduced by one migration.

The balanced and sided binary trees show intermediate features between the line and star.

#### 4.3 Migration Interval

Figure 5, 6 show the solution curves of the best chromosome and the variance of the root node with changing the migration interval. In Fig. 5, the migration with interval being equal to 1 performed very fast evolution, but led to immature

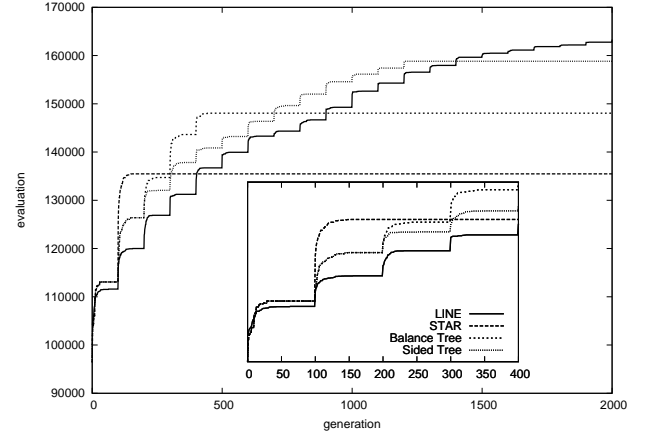


Figure 3. Solution Curves with Varying Topologies

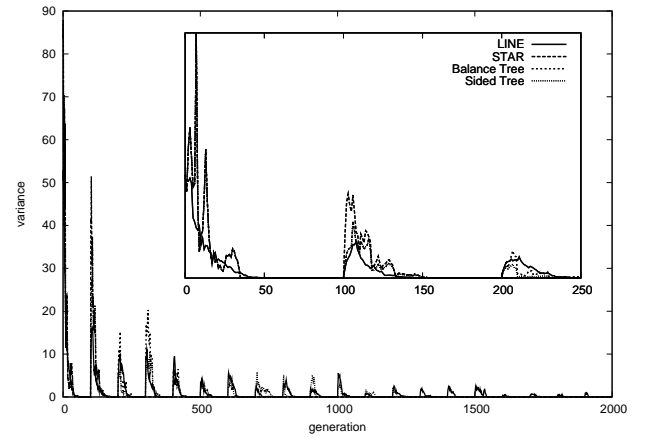


Figure 4. Variance Curves with Varying Topologies

convergence. We can observe that the variance is also drastically reduced by the often migration. Therefore, we need to set long enough interval for the proper migration.

#### 4.4 Number of Processes

Figure 7, 8 show the solution quality and the variance with changing the number of processes. In Fig. 7, the case with 25 processes can obtain better solution quality in the long run. Figure 8 shows that variety can be kept more if the number of processes is more.

### 5. Conclusion

In this paper we presents some remarkable results of the experimental evaluation for a parallel genetic algorithm based on tree topology migrations. The evaluation focused on the relation between the solution quality and the chromosome variance. We performed the evaluation for the typical tree topologies; the star, line, and balanced binary tree with varying migration interval since the solution quality of the parallel genetic algorithm depended on the kind of the topology and migration interval. The results showed that the line topology could keep the chromosome variety of the parallel GA and

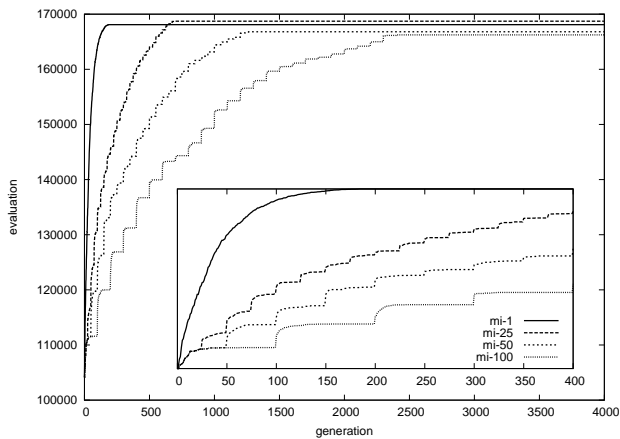


Figure 5. Solution Curves with Varying Migration Intervals

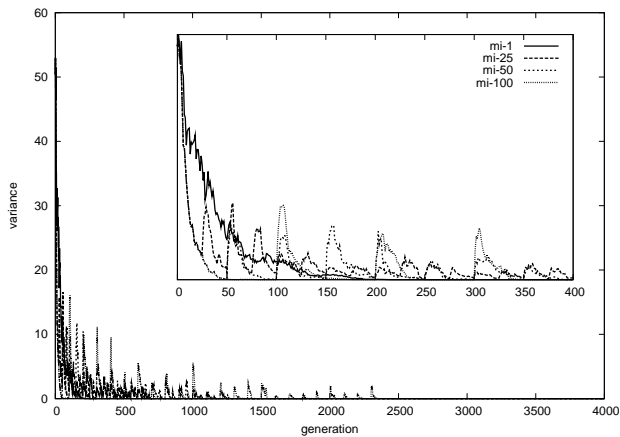


Figure 6. Variance Curves with Varying Migration Intervals

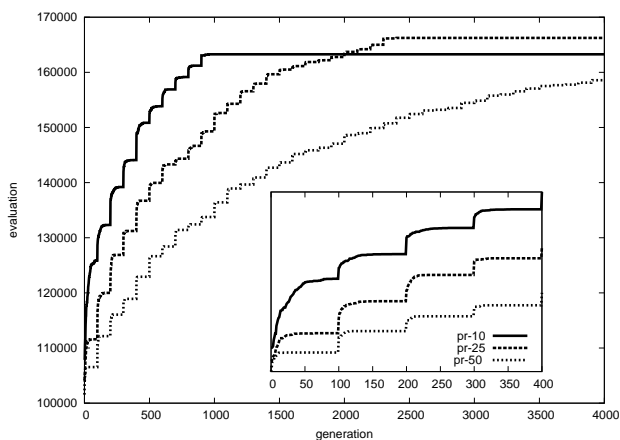


Figure 7. Solution Curves with Varying Number of Processes

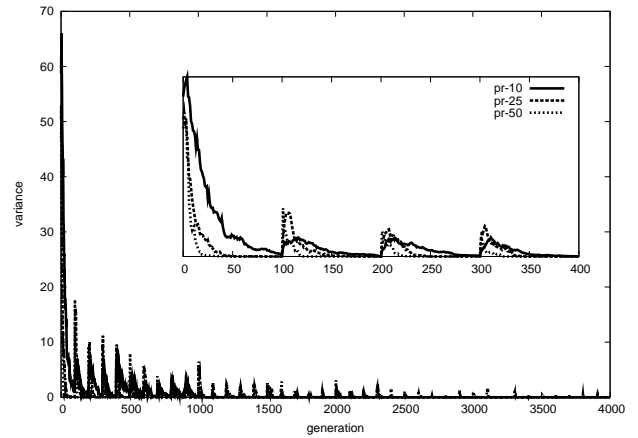


Figure 8. Variance Curves with Varying Number of Processes

obtain better quality of solutions since the topology has the longest distance between the root and the leaves.

## References

- [1] D.E.Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley, 1989.
- [2] Wadiq M.Sait, Habib Youssef, *Iterative Computer Algorithms with Application of Engineering*, Ieee Computer Society, 2000.
- [3] Enrique Alba, *Parallel Metaheuristics*, Wiley-InterScience, pp.107-125, 2005.
- [4] Reiko Tanese, *Distributed Genetic Algorithms*, Proceedings of the Third International Conference on Genetic Algorithms, pp434-439, 1989.
- [5] Yiyuan Gong, Morikazu Nakamura, Takashi Matsumura, and Kenji Onaga, *A Distributed Parallel Genetic Local Search with Tree-based Migration on Irregular Network Topologies*, IECIE Trans. on Fundamentals, Vol. E87-A, No 6, pp.1377-1385, 2004.
- [6] Yiyuan Gong, Senlin Guan, and Moikazu Nakamura, *Migration Effects of Parallel Genetic Algorithms on Line Topologies of Heterogeneous Computing Resources*, IECIE Trans. on Fundamentals, Vol. E91-A, No. 4, pp.1121-1128, 2008.
- [7] Hiroshi Furutani, *Analysis of Genetic Algorithms by Schema Theorem :Roles of Mutation and Crossover*, Transactions of InformationProcessing Society of Japan, Vol. 43 No. SIG 10(TOM7), pp35-45,2002.
- [8] Fisher,R.A. *The Genetical Theory of Natural selection*, 2nd edition, Dover, New York(1985).
- [9] Message Passing Interface Forum, "http://www.mpi-forum.org".