Proceedings of the 2001 IEEE Congress on Evolutionary Computation Seoul, Korea • May 27-30, 2001

TGA: A New Integrated Approach to Evolutionary Algorithms

Chuan-Kang Ting
Department of Computer Science and Engineering
National Sun Yat-Sen University
Kaohsiung, Taiwan
d8834804@student.nsysu.edu.tw

Sheng-Tun Li*
Department of Information Management
National Kaohsiung First University of Science and Technology
Nantz, Kaohsiung, Taiwan
stli@ccms.nkfust.edu.tw

Chungnan Lee
Department of Computer Science and Engineering
National Sun Yat-Sen University
Kaohsiung, Taiwan
cnlee@mail.nsysu.edu.tw

Abstract- Genetic Algorithm (GA) is a well-known heuristic optimization algorithm. However, it suffers from the serious problem of premature convergence, which is caused mainly by the population diversity decreasing in evolution. In this paper, we propose a novel algorithm, called TGA, which integrates the memory structure and search strategy of Tabu Search (TS) with GA. As such, the selection efficiency is improved and the population diversity is maintained by incorporating the regeneration operator. The traveling salesman problem is used as a benchmark to evaluate TGA and compare it with GA and TS. Experimental results show that TGA gets the better performance than GA and TS in terms of both convergence speed and solution quality.

1. Introduction

Genetic Algorithm (GA) is a well-known heuristic algorithm based on the imitation of natural evolution [1, 2]. Many studies have shown GA's outstanding ability in search and optimization problems. A serious problem of GA is the premature convergence, which makes the GA fall into local optimum. The key point is the decreasing of population diversity during evolution [3, 4]. When the diversity is insufficient, the search will limit in a certain region and lack vitalities to escape, and then the premature convergence occurs.

Several studies proposed different methods to maintain the population diversity in evolution. The adjustments generally focus on the genetic operators: selection, crossover [5, 6, 7], and mutation [8]. As far as selection is concerned, H. Kita and M. Yamamura [6] indicated that the selection operation narrows the population diversity in the process of selecting and duplicating. Therefore, to keep off the premature convergence, the selection must not only take proper individuals to exploit the interesting region, but also prevent from producing too many similar offspring to lose population diversity.

In order to maintain population diversity with selection, Shimodaira [9] proposed the cross-generation probabilistic survival selection (CPSS) based on the hamming distance from the best individual. Using the CPSS, the individuals similar with the best will have less probability to be selected. Therefore, it makes the resemblance to the best individual decrease, and thus the population diversity is maintained. Matsui [10] proposed the correlative tournament selection (CTS), which picks distinct parents to avoid propagating similar offspring. Among two parents and two offspring, the correlative family-based selection (CFS) takes the best and the one most distinct to survive. Thus the population diversity of genotype can be maintained. Motivated by simulated annealing (SA), Mori [11] proposed the thermodynamical genetic algorithm (TDGA), in which the selection intends to minimize the free energy of population and then keeps the diversity.

Most of these selecting approaches are on the basis of computing the differences of individuals. Alternatively we adopt the memory structure to perform the selection operation. Tabu Search (TS) [12] is a meta-heuristic algorithm using explicit memory structures to record the trajectory of searching and contribute to guide the search in consideration of both intensification and diversification. Many studies revealed that TS outperforms GA in certain applications.

In this paper, we propose a novel algorithm, called TGA, which integrates TS with GA for improvements. There are various studies toward hybridizing GA and TS to enhance the heuristic algorithms. F. Glover and M. Laguna [13] first proposed the scatter search to provide possibilities for integrating GA and TS, and then several studies used the hybrid of GA and TS in different ways. K. Handa and S.

* This work was supported in part by NSC89-2213-E-327-007.

Kuga [14] considered the different convergence speed of GA and TS in the first and the second half of the search. Hence they proposed the concatenation of GA and TS, which switched the search of two methods to avoid premature convergence. Sue [15] exploited GA and TS to solve different levels of problems. A common way of hybridization is to view TS as the enhancement of local search for GA. The best solution, which obtained from the GA population, is performed with TS to search the neighborhood. Then the result of TS is returned to GA and acts as a member of population [16, 17]. K. Nara [18] additionally used the tabu list to forbid the mate of chromosomes, whose hamming distances are too close with each other. Ozdamar [19] combined TS and SA to enhance the local search. Chin [20] proposed that the intensity of TS is gradually increasing as generations.

In general, most hybrids of GA and TS are based on the philosophy of running these two algorithms separately and then passing the computation result from one to the other as an initial solution. The structure of the original algorithm is not altered in these studies. By contrast, in our proposed algorithm TGA, the characteristics of TS are taken into consideration for the selection operator of GA instead of running GA and TS by turns. The selection will be guided according to the search strategy of TS. By incorporating TS into GA, it can achieve a better performance than by applying GA or TS independently.

The rest of this paper is organized as follows. In Section 2, we briefly describe the algorithms of GA and TS. In Section 3, the detailed description of TGA is presented. Performance evaluation on the Traveling Salesman Problem (TSP) of the proposed algorithm is presented in Section 4. Finally, conclusions are given in Section 5.

2. GA and TS

GA and TS are well-known heuristic algorithms. Many studies have shown that they can achieve good results in hard combinatorial optimization problems. Before presenting our proposed algorithm, we briefly review the concepts of GA and TS as follows.

2.1. Genetic Algorithm

Genetic Algorithm (GA) was first proposed for optimization by Holland [1]. The algorithm is based on Darwin's "The Fittest Survives" and mimics the evolution in nature to obtain the optimization. The possible solution is usually encoded as binary string, which is similar to chromosome (individual). Each chromosome consists of a number of genes. An accompanied fitness value is used to evaluate the quality of each chromosome. The better the quality, the higher the fitness is. In contrast to other optimization methods, GA uses multiple agents to search. A population contains certain number of solutions, and always keeps the fittest ones during evolution.

GA begins with an initial random population. Then genetic operators are applied iteratively to chromosomes in order to get better solutions (fitter chromosomes). The operators mimic the evolutionary mechanism in nature. First, the selection operator chooses a pair of chromosomes as parents. The probability of selection is in proportion to chromosome's fitness value. The chromosomes with higher fitness have greater opportunities to be chosen, which fits the Darwin's assumption: the fitter ones have more opportunities to generate more offspring. After parents are selected, one applies the crossover operator to exchange and recombine parts of the genes from parents to produce offspring. Finally, the mutation operator is applied to mutate some genes according to the mutation probability. In general, the probability of crossover ranges from 0.8 to 1.0, and that of mutation ranges from 0.01 to 0.2.

GA applies these operators iteratively until a predetermined number of iterations (generations) reaches or the fitness value converges.

2.2. Tabu Search

Tabu Search (TS) was a meta-heuristic approach proposed by Glover [12]. Explicit memory structure is introduced to guide the searching process. Many researches have shown that Tabu Search has better performance in various combinatorial optimization problems [12].

The basic components of Tabu Search are described as follows.

- Move: the process from one solution state to another, i.e. the process from current solution to neighboring one.
- Neighborhood: the set of trial (candidate) solutions. The trial solution is defined as the solution that is related to current solution by a little permutation. For example, a set of strings that differ from the current string by one character.
- Tabu List: it records the moves that are forbidden. This is the most unique feature of Tabu Search, which prevents the search from tapping or cycling in the local minimum. The size of tabu list affects the search strategy. The larger size of tabu list makes the search focus on exploration or diversification, whereas the smaller size makes the search focus on intensification [21].
- Aspiration criteria: it makes the superior solutions have opportunity to override the tabu restriction. If a trial solution is better than the best solution, the move is allowed even that it is in the tabu list.

Tabu Search begins with an initial solution that is generated randomly (labeled as the current solution). Then, its neighborhood is generated from the current solution and sorted in descending order. If the best trial solution in the neighborhood is not in tabu list, or if it is in tabu list but satisfied with the aspiration criteria, then it is chosen to be the new current solution. Otherwise, the next trial solution is chosen to examine. This process is repeated until the search converges or the iteration is terminated, the best solution obtained so far is the result of optimization.

3. The Proposed Algorithm - TGA

In this section, we propose the novel optimization algorithm, TGA, which is based on the evolutionary structure of GA and is augmented by the features of TS. Instead of running GA and TS by turns, we propose a hybrid approach for allowing more fusion. The new approach relies on GA for the adaptation and robustness of genetic operators, and integrates with the memory structure and search strategy of TS.

The pseudocode of proposed TGA is presented in Figure 3.1. Most procedures of TGA are the same with original GA except the processes of selection and regeneration. In TGA, the selection operator will repeat until an acceptable couple is picked. The strategy of selection incorporates the characteristics of TS more than fitness merely. First, the tabu list forbids certain mates in order to prevent duplicate selection from retarding the diversification. Second, the aspiration criterion makes the superior offspring have opportunity to override the tabu restriction.

When the population diversity is too low to select an acceptable couple, the selection will repeat infinitely and the deadlock occurs. To avoid this, a predefined criterion of repeated selection is used to escape the infinite loop. Moreover, we introduce the regeneration operator to violently disturb a randomly selected individual as newborn, which is expected to activate the population diversity and improve the probability of successful selection.

To accommodate to TGA, the components of TS should be modified accordingly. The detailed description is presented in the following subsections.

```
TGA()
{
    t = 0
    initialize population P(t)
    evaluate P(t)
    while not terminated do
        t = t + 1
        while (population size P(t) not filled) do
            n = 0
            repeat
                n = n + 1
                select parents from P(t-1)
                crossover
            until (not tabu) or (aspiration) or (n > deadlock)
            if (n > deadlock)
                regeneration
            else
                mutation
        od
        survive P(t-1), P(t)
    od
}
```

Figure 3.1. Pseudocodes for TGA

3.1. Representation

In addition to original genes of solution information, a memory structure, consists of clan number and tabu list, is introduced to record the trajectory of evolution for the strategy of TS. The clan number, a unique number for clan identification, is assigned at the stage of initialization with each chromosome. The offspring will inherit the clan number from parents during evolution. As the functionality of surnames in human society, the clan offers protection for certain reasons such as eugenics. We join the tabu list with this clan number to indicate forbidden moves rather than record all the chromosome's information. Besides, in order to accommodate to the multiple agents of GA, the tabu list is appended with clan to each chromosome instead of a fixed table of memory. An example of representation for the TSP is shown in Figure 3.2.

8	26	7	16	……	22	7	2	8	
genes (cities for TSP)						#clan	Tabu List		

Figure 3.2. Chromosome structure

3.2. Tabu List

The effect of the tabu list in TGA focuses on the diversification in population without sacrificing the intensification, however. In TGA, the memory structure acts like the pedigree of clan, which records the trajectory of evolution. As the tabu list prevents some moves from trapping in local optimum in TS, it forbids the individuals with the same clan to mate in TGA. Thus the assimilation caused by inbreeding can be avoided, and the loss of population diversity from similar selection can be restrained as well.

In practice, if we ignore the internal processes of recombination such as crossover and mutation, the external behaviors of selection and reproduction can be viewed as a move. Similar to the move in TS, the selection will be restricted by tabu; namely, mating with a chromosome labeled tabu is forbidden. In this way, the TGA will follow the fittest survivorship of GA in consideration of strategy of TS.

The aspiration criterion provides opportunity for the superior solution to override the tabu restriction. If the trial mating could produce offspring superior to the best solution so far, then the mating is allowed in spite of tabu. It will encourage the intensification under the consideration of diversification.

Figure 3.3 illustrates how the operations work. The crossover illustrated here is the so-called partially matched crossover (PMX) [2]. There are two stages in which adding tabu is necessary. First, when two chromosomes are selected as parents, one will add the clan number of its mate to the tabu list. This kind of tabu list prevents the duplicate mating, which leads to the decreasing of diversity in the offspring population. Second, the offspring will copy the update of clan number and tabu list from one of the parents. It inherits the information about antecedents to prevent inbreeding, which causes the assimilation of chromosomes to decrease the diversity.

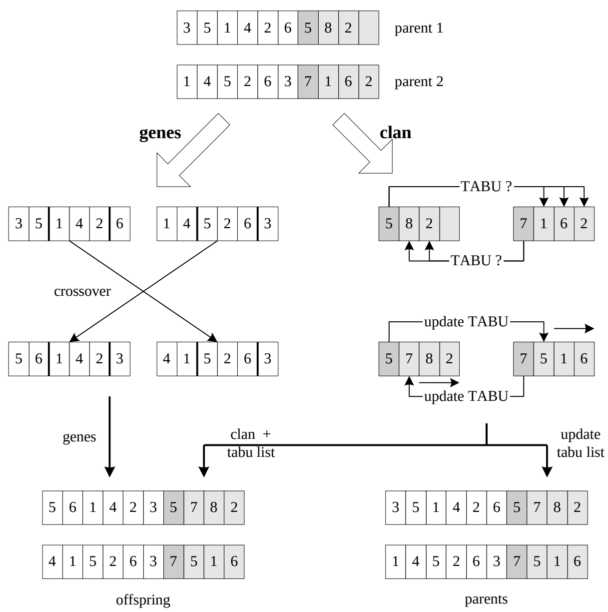


Figure 3.3. The illustration of the operations

4. Performance Evaluation

Traveling Salesman Problem (TSP) is a classical combinatorial optimization problem. This problem consists of several random cities. Its objective is to find the shortest path to travel all the cities. In this paper, we conduct several experiments on TSP to evaluate the performance of TGA in terms of both solution quality and convergence speed. Furthermore, to verify the diversification of TGA, the variation of population diversity is examined by computing the cyclic hamming distance between individuals. Finally we compare TGA with GA and TS.

The crossover operator adopted in our experiments follows the partially matched crossover (PMX), which is usually proposed to tackle TSP for GA [2]. The mutation operator is to pick two random genes and swap. To consider the duplicate individuals will decrease the diversity of population continuously and lead to premature convergence [9], we remove the duplicates from survivals in GA and TGA.

In TGA, an additional operator called regeneration is necessary. Here we adopt the shift-operator as regeneration for TSP. As shown in Figure 4.1, the shift-operator picks an arbitrary position to shift the genes a random distance. Therefore, the positions of genes are disturbed completely, but the sequence is unchanged. In this way, the effort of TSP optimized so far is kept, and the further variation for individual is provided.

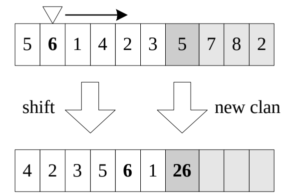


Figure 4.1. The shift-operator

Each experiment will take 20 trials for generalization. Two sets of data, 50 and 100 cities, are generated to evaluate these algorithms. For generalization and comparability the generator of city data follows the 8th DIMACS Implementation Challenge [22]: symmetric TSP, two dimension Euclidean metric distance (EUC_2D). The population size of GA varies with 20, 50, and 100 chromosomes that are generated randomly every time. The other parameters for GA are set as follows. Population size = 20, crossover rate (pC) = 1.0, and mutation rate (pM) = 0.05. The parameters for TGA follow the values of GA except the clan. The size of clan structure, which contains one clan number and tabu list, is defined in proportion to the size of population. An excessive large size of clan structure will make the selection commonly fail to mate, but an excessive small size will lose its effect. In our experiments, the acceptable ratio of clan structure in TGA is set to 0.2 as default:

$$clan\_size = 0.2 \times pop\_size$$

The population diversity is defined based on hamming distance between chromosomes. Since the order of cities for TSP is cyclic, two chromosomes with large hamming distance may be the same in sequence. For this reason, we must align the chromosomes to calculate the real difference between chromosomes. To consider that the difference between aligned strings is least, the hamming distance for TSP can be defined as:

$$H(x, y) = \min\left(h(x_k, y)\right) \qquad k = 0,1\cdots,c-1$$

where $h(x, y)$ is the hamming distance between chromosome x and y, x_k denotes the chromosome shifted k genes, c is the number of cities, and p is the population size. By the cyclic hamming function H, the degree of population diversity can be evaluated more exactly. In our experiments, two kinds of measurement are used to evaluate the population diversity. The first one, D_{best}, is the measurement of difference between chromosomes and the best solution. Second, the difference in full population between each other, D_{full}, is calculated as well.

$$D_{best} = \frac{1}{p}\sum_{k=1}^{p} H(x_k, x_{best})$$

$$D_{full} = \frac{2}{p(p-1)}\sum_{i=1}^{p}\sum_{j=i+1}^{p} H(x_i, x_j)$$

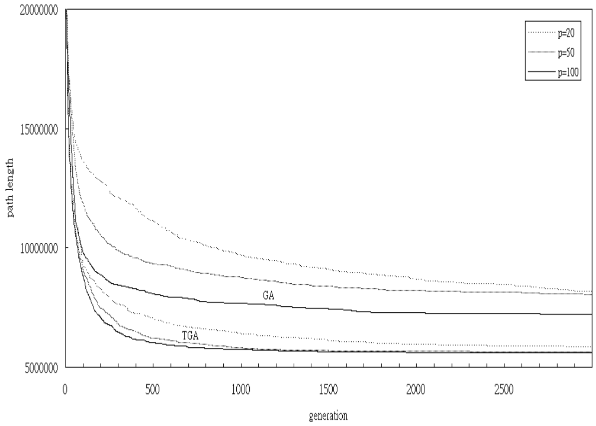


Figure 4.2(a) The plot of convergence at c=50

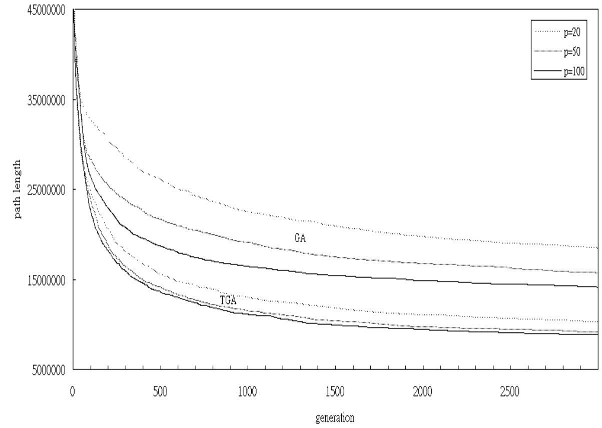


Figure 4.2(b) The plot of convergence at c=100

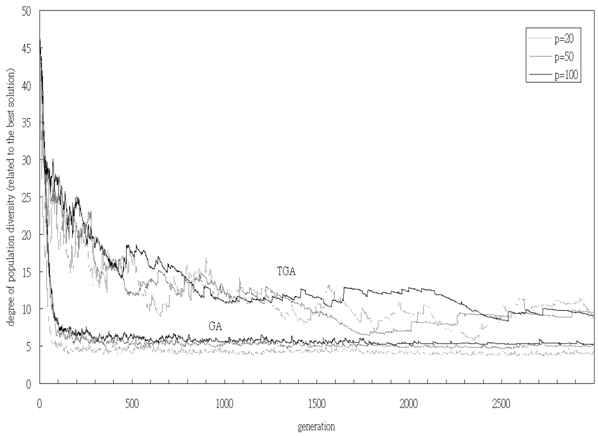


Figure 4.3(a) The plot of population diversity related to the best solution at c=50

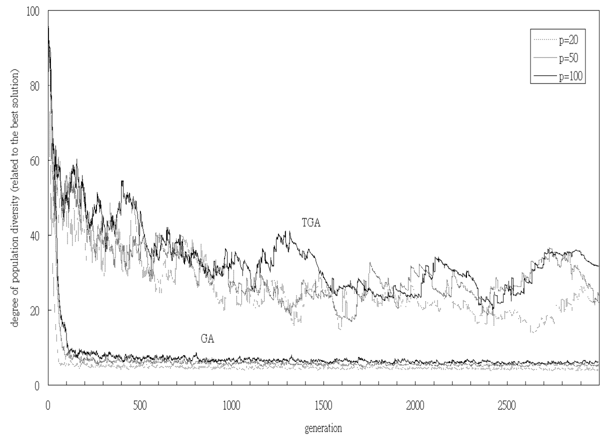


Figure 4.3(b) The plot of population diversity related to the best solution at c=100

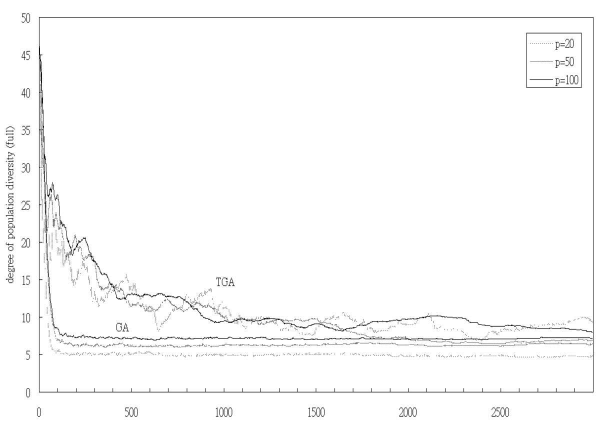


Figure 4.4(a) The plot of population diversity at c=50

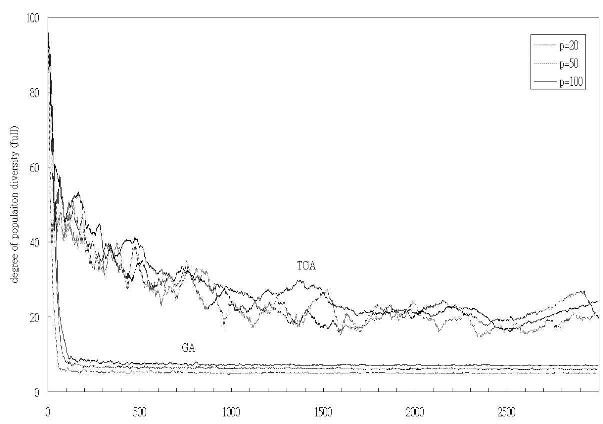


Figure 4.4(b) The plot of population diversity at c=100

Figures 4.2(a)~4.4(b) present the convergence and variation of population diversity on different population size for GA and TGA. Figure 4.2 show that both the convergence speed and solution quality of TGA are better than those of GA on all population size. The TGA with smaller population size can obtain satisfied result as well. With the memory structure of the clan and tabu list, TGA selects parents by strategy and exploits the individuals more efficiently. Figure 4.3 shows the variation of population diversity related to the best solution. Figure 4.4 shows the degree of diversity in full population. These results prove that TGA is indeed able to maintain the population diversity based on following reasons. First, The decrease of population diversity for TGA is much slower than GA. It supports that the selection with tabu list restrains the inbreeding leading to assimilation, which makes the population monotonous. In addition, the moderation of losing diversity will prevent the search from over-intensified and help the exploration more thorough. Second, the degree of diversity for TGA is maintained above 20%, and that for GA remains only 5% to 10%. In TGA, when the population diversity is too low to achieve a successful selection, the regeneration operator will activate the population to increase the degree of diversity. Thus the population diversity will remain enough vitality to escape local optimum and prevent premature convergence.

To further evaluate the performance of TGA, we compare it with GA and TS separately. The move operator of TS is defined to exchange two genes. TS performs the moves sequentially on all adjacent solutions to generate neighborhood, and the best of them are selected as current solution. In our experiments, the tabu list of size 12 is acceptable empirically. The algorithms are implemented in C language and run on Intel PentiumIII-600 Windows system. The experiments will run 3000 generation and 20 trials for each set of parameters.

Tables 4.1 and 4.2 illustrate the best solution obtained from each algorithm. The results show that TGA is significantly better than GA and TS in terms of solution quality. The superiority of TGA ranges from 26% to 44% over GA and 16% to nearly 30% over TS. The deviation gets greater in larger number of cities. The computation time spent for TGA is longer than GA, but is much shorter than TS.

Figure 4.5 depicts the convergence of TGA compared with GA and TS. From these plots, we can see that TGA gets better solution than GA and TS. The convergence speed of TGA is faster than GA, but is slower than TS. Although the convergence of TS is the fastest, it spends much more computation than TGA and GA. To consider the computation time, we compare these algorithms by period of time instead of iteration in the following experiments.

The comparison of convergence in other hybrid approaches is mostly measured by iterations. However, an algorithm may spend more time on computation while it converges faster in terms of iteration. Because each algorithm has different computation in one iteration, it is unfair to compare the speed of convergence by iteration only. For this reason, we record the state of the best solution by a predetermined time period instead of the number of iterations. Each algorithm also takes 20 trials to obtain the average trajectory converged.

The result of convergence by time is shown in Figure 4.6. In lower population size the convergence of TS turns into the slowest in time due to more computation costs. Although TGA spent more computation, it converges faster than GA. It indicates that the strategy of TGA contributes the evolution to operate more efficiently, in other words, enhancing the ability of intensification. This result confirms that the extra computation of TGA is worthy. As the population size gets larger, GA and TGA converges slower than TS because they must consume more time in one generation as the number of individual increases.

From Figures 4.5 and 4.6, it is worth noting that the TGA in small population size performs as well as in larger population size. The regeneration operator activates the population adaptively and compensates the lack of diversity in smaller population size. Moreover, the selection with the strategy of clan does help in mating for more efficient intensification and dynamically triggering the regeneration for diversification. It further suggests that we may apply TGA with smaller population to reduce computation and can get an acceptable solution as well.

From the comparison with GA and TS, TGA shows the better performance in convergence. Furthermore, TGA demonstrates its superiority in solution quality, especially when the complexity of problem increases.

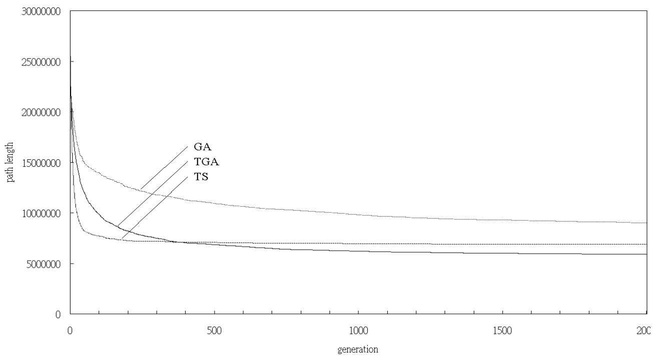
Table 4.1: Comparisons of the best solutions at c=50

pop size	algorithm	Avg.	Std	*Dev (%)	Time (sec)
20	TGA	5784933	207717	-	5.91
	GA	8577424	602286	32.56	2.25
	TS	6916467	516874	16.36	127.25
50	TGA	5657278	142658	-	35.73
	GA	7677595	687542	26.31	10.66
	TS	6916467	516874	18.20	127.25
100	TGA	5606238	156490	-	149.48
	GA	7630451	591935	26.53	37.78
	TS	6916467	516874	18.94	127.25

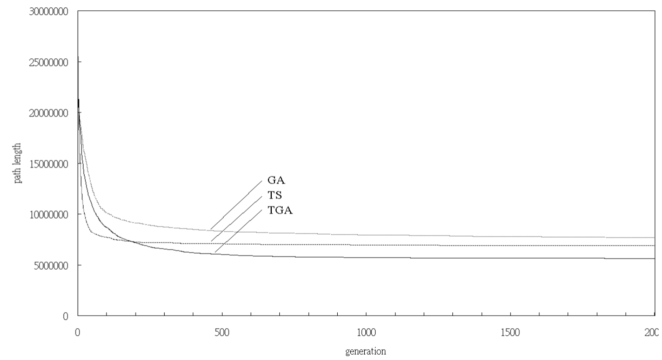
Table 4.2: Comparisons of the best solutions at c=100

pop size	algorithm	Avg.	Std	*Dev (%)	Time (sec)
20	TGA	10165336	411612	-	7.90
	GA	18667690	969080	45.55	3.22
	TS	12812263	997175	20.66	1994.47
50	TGA	9200889	304024	-	46.81
	GA	15997238	839670	42.48	16.58
	TS	12812263	997175	28.19	1994.47
100	TGA	8969337	346174	-	183.35
	GA	14268095	931228	37.14	57.78
	TS	12812263	997175	29.99	1994.47

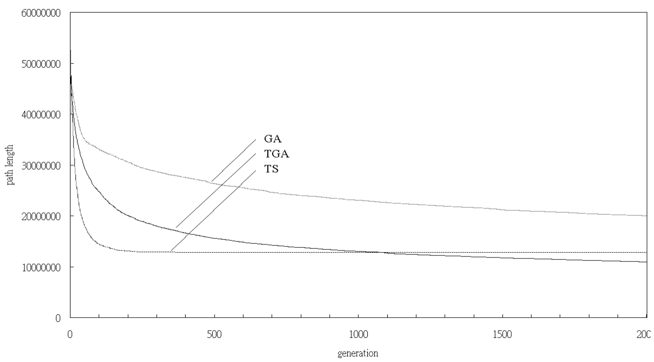
*Dev: relative deviation of best solution obtained by TGA from that obtained by GA or TS



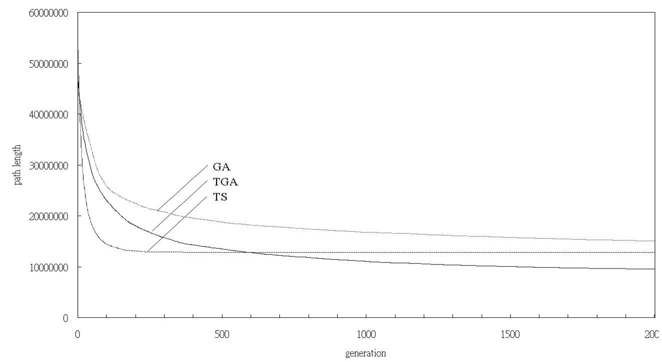
(a) c=50, p=20



(b) c=50, p=100

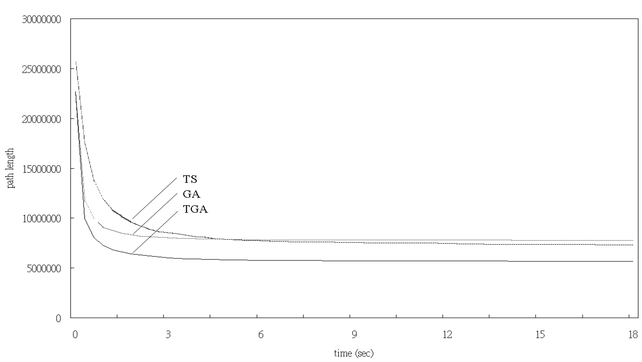


c) c=100, p=20

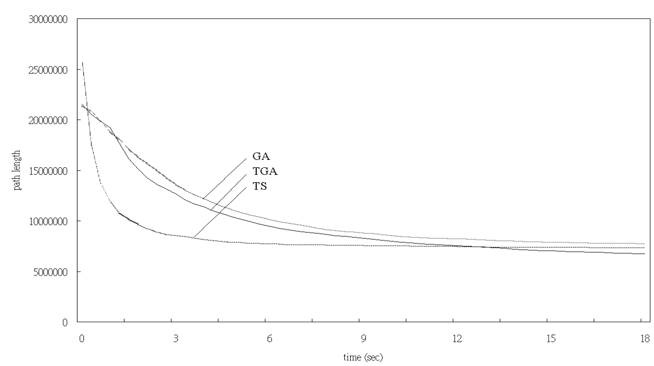


(d) c=100, p=100

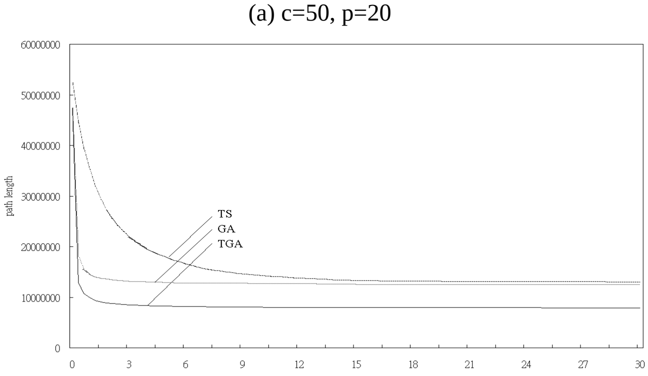
Figure 4.5. The comparison of GA, TS, and TGA by generation
for different number of cities *c* and different population size *p*



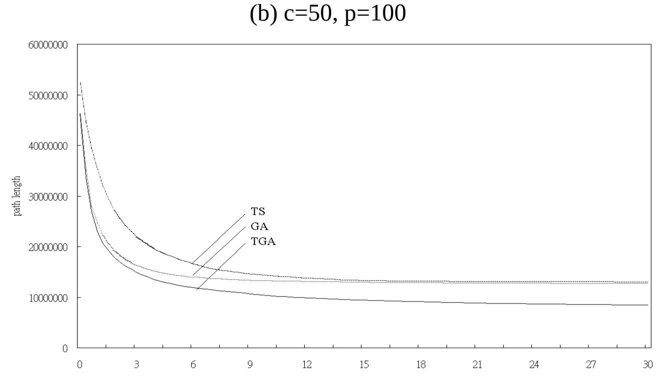
(a) c=50, p=20



(b) c=50, p=100



(c) c=100, p=20



(d) c=100, p=100

Figure 4.6. The comparison of GA, TS, and TGA by time
for different number of cities *c* and different population size *p*

5. Conclusions

The premature convergence is a serious problem in GA, and the key point is the lost of population diversity during evolution. Lots of studies have tried to adjust the operators to maintain the population diversity. The calculating hamming distance mostly bases these modifications. In this paper, we have proposed a memory-based approach incorporated with the regeneration operator to maintain the population diversity instead of calculation.

The proposed algorithm, TGA, is based on the structure of GA by adopting the features of adaptation and parallelism, and integrates with the memory structure and search strategy of TS. A clan of chromosome is introduced as the identification of trajectory in evolution. The selection operator is then guided by the clan and the tabu list for the consideration of efficient mating for intensification and prohibition against inbreeding for diversification. In addition, the fact that the regeneration timely activates the population results in sufficient diversity for exploration. The traveling salesman problem was used as a benchmark to evaluate the TGA and compare it with GA and TS. The experimental results show that TGA can outperform GA and TS in terms of both convergence speed and solution quality. Furthermore, the satisfying results in smaller population size suggest that we may apply TGA with smaller population to reduce computation and can get an acceptable solution as well. Further studies in discovering the respective effects of tabu list in TGA and applying it to other complicated problems are underway.

References

[1] Holland, J. H., “Adaptation in natural and artificial systems,” University of Michigan Press, 1975.

[2] D. E. Goldberg, “Genetic algorithm in search, optimization, and machine learning,” Addison-Wesley publishing company Inc., 1989.

[3] Y. Leung, Y. Gao, and Z. B. Xu, “Degree of population diversity – a perspective on premature convergence in genetic algorithms and its markov chain analysis,” IEEE Transactions on Neural, Networks, vol. 8, no. 5, pp. 1165-1176, 1997.

[4] L. Qu and R. Sun, “A synergetic approach to genetic algorithms for solving traveling salesman problem,” Information Science, vol. 117, pp. 267-283, 1999.

[5] F. Herrera, M. Lozano, and J. L. Verdegay, “Fuzzy connectives based crossover operators to model genetic algorithms population diversity,” Fuzzy Sets and Systems, vol. 92, pp. 21-30, 1997.

[6] H. Kita and M. Yamamura, “A functional specialization hypothesis for designing genetic algorithms,” IEEE Int. Conf. on Systems, Man, and Cybernetics, vol. 3, pp. 579-584, 1999.

[7] R. K. Ahuja, J. B. Orlin, and A. Tiwari, “A greedy genetic algorithm for the quadratic assignment problem,” Computers and Operations Research, vol. 27, pp. 917-934, 2000.

[8] C. C. Hsu, H. Takahashi, K. Shida, H. Fujikawa, and S.I. Yamada, “An eugenic mutations for optimum problems,” IEEE Int. Conf. on Industrial Electronics, Control, and Instrumentation, vol. 2, pp. 1537-1542, 1995.

[9] H. Shimodaira, “DCGA: a diversity control oriented genetic algorithm,” IEEE Int. Conf. on Tools with Artificial Intelligence, pp. 367-374, 1997.

[10] K. Matsui, “New selection method to improve the population diversity in genetic algorithms,” IEEE Int. Conf. on Systems, Man, Cybernetics, vol. 1, pp. 625-630, 1999.

[11] N. Mori, J. Yoshida, H. Tamaki, H. Kita, and Y. Nishikawa, “A thermodynamical selection rule for the genetic algorithm,” IEEE Int. Conf. on Evolutionary Computation, vol. 1, pp. 188-192, 1995.

[12] F. Glover and M. Languna, “Tabu Search,” Kluwer academic publishers, 1997.

[13] F. Glover, J. P. Kelly, and M. Laguna, “Genetic algorithms and tabu search: hybrid for optimization,” Computers Ops Res. vol. 22, no. 1, pp. 111-134, 1995

[14] K. Handa and S. Kuga, “Pollycell placement for analog LSI chip designs by genetic algorithms and tabu search,”

[15] A. H. Sue, “A hybrid heuristic for the uncapacitated hub location problem,” European Journal of Operational Research, no. 106, pp.489-499, 1998.

[16] A. H. Mantawy, Y. L. Abdel-Magid, and S. Z. Selim, “Integrating genetic algorithms, tabu search, and simulated annealing for the unit commitment problem,” IEEE transactions on power system, vol. 4, no. 3, pp. 829-836, 1999.

[17] C. F. Liaw, “A hybrid genetic algorithm for the open shop scheduling problem,” European Journal of Operational Research, no. 124, pp. 28-42, 2000.

[18] K. Nara, “Genetic algorithm for power systems planning,” Proc. of the 4th int. conf. on advances in power system control, operation and management, 1997.

[19] L. Ozdamar and S. I. Birbil, “Hybrid heuristics for the capacitated lot sizing and loading problem with setup times and overtime decisions,” European Journal of Operational Research, no. 110, pp. 525-547, 1998.

[20] A. J. Chin, H. W. Kit, and A. Lim, “A new GA approach for the vehicle routing problem,” IEEE, 1999.

[21] K. S. Al-Sultan and C. A. Fedjki, “A tabu search-based algorithm for the fuzzy clustering problem,” Pattern Recognition, vol. 30, no. 12, pp. 2023-2030, 1997.

[22] “8th DIMACS Implementation Challenge: The Traveling Salesman Problem”, http://www.research.att.com/~dsj/chtsp/index.html.