# Evaluation of Edge Assembly Crossover for Hybrid GA

Ikuo YOSHIHARA[1) ] Masakazu SATO[2)] Hung Dinh NGUYEN[3)]
Kunihito YAMAMORI[4)]

Abstract

Traveling Salesman Problem (TSP) is one of the representative combinatorial optimization problems. The promising approach to solve TSP is Genetic Algorithm (GA). GA has global search ability, and heuristic is used in order to compensate local search ability, because GA is lack of local search ability. Hybrid GA (HGA) combined with heuristics can be expected to obtain high quality solutions.We employ Lin-Kernighan heuristics which is very effective in TSP. Because crossover is one of the important operators of GA, lots of crossovers are proposed for TSP. We employ Edge Assembly Crossover (EAX) which can obtain high quality solutions. We evaluate the proposed HGA combined EAX and heuristics and show that it finds the best solutions.

Key Words:
Traveling Salesman Problem, Genetic Algorithm, Edge Assembly Crossover

## 1 Introduction

Traveling Salesman Problem (TSP) is a problem which finds the shortest tour to visit all the cities only once and come back to the starting city, and it is one of the representative combinatorial optimization problems. It is applied in many fields such as vehicle course plan, scheduling, X-ray crystal structure analysis, and VLSI wiring etc[1)2)] .

TSP is known as an NP-complete problem. It is difficult to find the optimal solution, because the total number of combination $\frac{(N-1)!}{2}$ increases more than exponentially with increase of the number of cities N.

The promising approach to solve TSP is Genetic Algorithm (GA). In spite of having global search ability, GA lacks in local search ability. Heuristics is used in order to compensate local search ability. We employ Lin-Kernighan heuristics which is very effective in TSP. The combination of GA and heuristics is called Hybrid GA (HGA).

Because crossover is one of the important operators of GA, many crossovers have been proposed for TSP, for example, partially matched crossover (PMX), order crossover (OX), cycle crossover (CX), maximal preservative crossover (MPX) [7)], greedy subtour crossover (GSX) [5)], edge recombination (ER), and edge assembly crossover (EAX). We employ EAX which is excellent in characteristic preservation, and can generate a wide variety of offspring. Nagata and Kobayashi showed good search ability of EAX by comparison experiment with the traditional crossover (EX, EXX) and the traditional local search operators[3)]. In addition, they found high quality solutions at TSPs (101-3,038cities) on the TSPLIB95 using GA that employed EAX, and they did not use any other local search heuristics[4)].

We evaluate the proposed HGA combined EAX and heuristics. We perform a comparison experiment with MPX and GSX in order to evaluate the quality of a solution.

## 2 Genetic Algorithms

To use a GA, we must represent a solution of the problem as genotype. The GA randomly creates a population of solution(in form of chromosomes) and applies genetic operators(crossover, mutation, and selection) to generate new solutions.

[1)]Professor, Dept. of Computer Science and Systems Engineering
[2)]Undergraduate student, Dept. of Computer Science and Systems Engineering
[3)]Doctoral student, Graduate School of Engineering
[4)]Associate Professor, Dept. of Computer Science and Systems Engineering

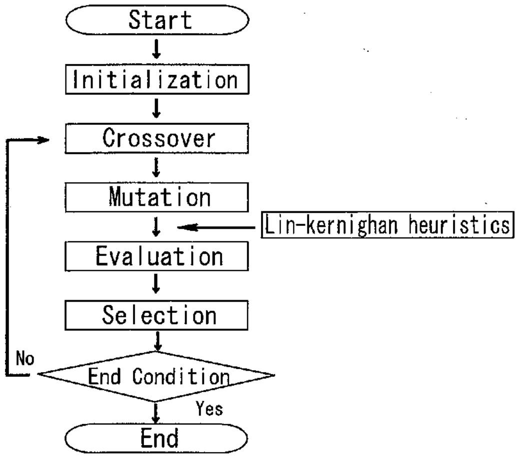


Fig. 1 The flow chart of HGA

## 2.1 Hybrid GA

In spite of having global search ability, GA lacks in local search ability. Employing local search heuristics (Fig.1), GA is vastly improved[5]. The combination of GA and heuristics algorithm is called Hybrid GA (HGA). HGA is adopted in many cases such as a combinatorial optimization problem and real world applications[6].

## 2.2 A multi-community model based on GENITOR-TYPE GA

A multi-community model based on GENITOR-TYPE GA is used in this paper.

**(1)** A multi-community GA (Island model)

The island model is called by the analogy from population genetics. The population is divided into some subpopulation, and a part of subpopulation is exchanged (migration) at intervals of a certain generation (migration interval). The island model can maintain diversity in population.

**(2)** GENITOR-TYPE GA

GENITOR[9] has some characteristics.

- Using linear ranking for selecting parents.
- In each generation, crossover is performed only for one pair and mutation is performed only for one individual.

GENITOR is acknowledged hard to converge prematurely and stagnate.

## 2.3 Genetic Coding

Path representation that city names are enumerated in turn from the starting city is used for coding.

For example, tour (1 - 4 - 2 - 7 - 3 - 6 - 5) is expressed like Fig.2.

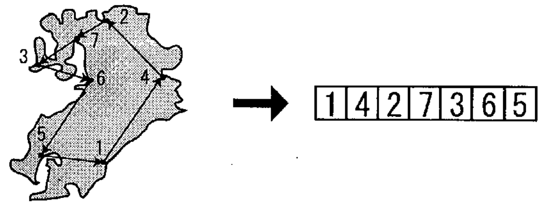


Fig. 2 Genetic Coding

## 2.4 Genetic Operations

**(1)** Mutation

To clarify the effect of crossover, mutation is not employed.

**(2)** Fitness Evaluation

Individual which has the smaller tour length in population is given higher fitness. Therefore, the fitness of an individual is defined as the following formula,

$$fitness = \frac{1}{D}.$$

Here, D is tour length defined as follows,

$$D = \sum_{i=1}^{N} d_{L(i),L(i+1)}.$$

$N$: the number of cities

$d_{ij}$: distance between cities $i$ and $j$

$L(i)$: the city which it visits to the i-th

**(3)** Selection

In GA, it is determined whether an individual can survive based on the fitness. If the offspring is better than the worst parent, it is immediately inserted into population to replace the worst parent. As for a child, only one individual is generated by each generation.

**(4)** Crossover (EAX)

Parent's characteristics are inherited through crossover[8]. In TSP, it can be said that the good character which should be inherited to offspring is the subtour which was partially successful. The EAX is excellent in characteristic preservation, because intermediate individual are made of only edges of parents. In addition, the EAX can generate a wide variety of offspring.

The intermediate individual is modified into a valid tour, because it consists of some subtour.

**Step1** An intermediate individual $T$ which consists of some subtour is set.

**Step2** Subtour $U_r$ which consists of number of the fewest edge subtour $U_i(i = 1, 2, \cdots)$ in T is chosen.

**Step3** The pair$(v_1, v_2)$, $(v_3, v_4)$ from which the value of a formùla becomes the maximum is found.

$$w(v_1, v_2) + w(v_3, v_4) - w(v_1, v_4) - w(v_3, v_2) \quad (1)$$

$$((v_1, v_2) \in U_r, (v_3, v_4) \in U_i(i \neq r))$$

**Step4** The subtour containing edge$(v_3, v_4)$ is set as $U_s$. Edge $(v_1, v_2)$ and $(v_3, v_4)$ are removed, and edge $(v_1, v_4)$ and $(v_3, v_2)$are added. As a result, $U_r$ and $U_s$ are connected.

**Step5** Modification operation is ended when intermediate individual $T$ becomes one tour. It returns to $Step1$ except it.

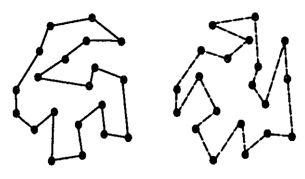


Fig. 3 An example parents(tourA,tourB)

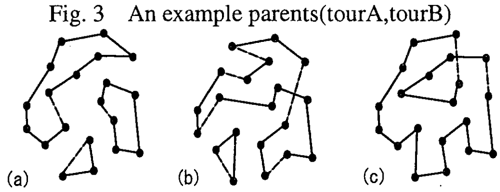


Fig. 4 Examples of intermediate individual obtained from the parents

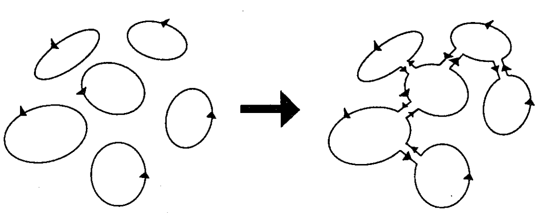

Fig. 5 An example of intermediate individuals (left) and a valid individual (right) obtained by modification

## 3 Comparison experiments of three crossovers

The comparison experiments with MPX and GSX are performed for evaluation of EAX.

### 3.1 Benchmarks and Experimental conditions

**(1)** Benchmarks

Ten TSPs(318-3795cities) are used on the TSPLIB95[1] which is the representative benchmark of TSP.

**(2)** GA parameters

- Population size $\cdots$ 32
- Number of islands $\cdots$ 2
- Selection bias $\cdots$ 1.25
- Mutation rate $\cdots$ 0.0
- Migration interval $\cdots$ 500
- Number of migrants $\cdots$ 3

**(3)** Terminal conditions

- In case an elite is not replaced for 2000 successive generations, or
- In case the generation gets at 10000.

**(4)** Computational environment

- CPU:Intel Pentium Ⅲ (935MHz)
- Main memory:256MB
- OS:FreeBSD 4.8
- Compiler:gcc (ver.2.95.4)

### 3.2 Quality of the solution

Quality of the solution is evaluated by the following formula.

$$quality(\%) = \frac{tour\ \ length - optimum}{optimum} \times 100 \quad (2)$$

### 3.3 Results and Discussions

Table1 is result of a comparison experiments of three crossovers. The experiments were performed 20 times to each problem. Fig.6 is change of tour length of the problem "pcb3038".

In case the number of cities is small i.e. 198-318, three crossovers (EAX,MPX,GSX) have scarcely differences. However, As the number of cities increases, compared with GSX, the high quality solutions found by EAX and MPX. In case the number of cities is big i.e. 3038-3795,

[1]URL http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/

Table 1 Result of comparison experiments of three crossovers

| EAX | | | | | |
|---|---|---|---|---|---|
| Problem | Optimum | Ratio of opt. | Quality(%) | Ave. gen. | CPU time(s) |
| d198 | 15,780 | 20/20 | 0.000 | 2045 | 500.34 |
| lin318 | 42,029 | 20/20 | 0.000 | 2100 | 276.61 |
| pcb442 | 50,778 | 20/20 | 0.000 | 2100 | 525.49 |
| att532 | 27,686 | 16/20 | 0.012 | 2180 | 593.90 |
| rl1323 | 270,199 | 8/20 | 0.005 | 2295 | 674.72 |
| fl1577 | 22,249 | 3/20 | 0.025 | 2780 | 3401.34 |
| d2103 | 80,450 | 0/20 | 0.049 | 2485 | 1580.26 |
| pr2392 | 378,032 | 8/20 | 0.003 | 2600 | 594.70 |
| pcb3038 | 137,694 | 2/20 | 0.011 | 2770 | 1140.86 |
| fl3795 | 28,772 | 11/20 | 0.051 | 2990 | 10344.28 |

| GSX | | | | | |
|---|---|---|---|---|---|
| Problem | Optimum | Ratio of opt. | Quality(%) | Ave. gen. | CPU time(s) |
| d198 | 15,780 | 20/20 | 0.000 | 2040 | 22.29 |
| lin318 | 42,029 | 20/20 | 0.000 | 2150 | 19.07 |
| pcb442 | 50,778 | 17/20 | 0.007 | 2300 | 7.14 |
| att532 | 27,686 | 14/20 | 0.018 | 2575 | 27.42 |
| rl1323 | 270,199 | 6/20 | 0.017 | 2570 | 51.10 |
| fl1577 | 22,249 | 0/20 | 0.103 | 3485 | 105.67 |
| d2103 | 80,450 | 0/20 | 0.110 | 2770 | 50.54 |
| pr2392 | 378,032 | 0/20 | 0.084 | 2875 | 25.62 |
| pcb3038 | 137,694 | 0/20 | 0.098 | 3530 | 40.89 |
| fl3795 | 28,772 | 0/20 | 0.306 | 3955 | 136.67 |

| MPX | | | | | |
|---|---|---|---|---|---|
| Problem | Optimum | Ratio of opt. | Quality(%) | Ave. gen. | CPU time(s) |
| d198 | 15,780 | 20/20 | 0.000 | 2045 | 93.73 |
| lin318 | 42,029 | 20/20 | 0.000 | 2110 | 31.14 |
| pcb442 | 50,778 | 20/20 | 0.000 | 2105 | 31.10 |
| att532 | 27,686 | 17/20 | 0.009 | 2505 | 60.77 |
| rl1323 | 270,199 | 6/20 | 0.007 | 2390 | 87.15 |
| fl1577 | 22,249 | 2/20 | 0.025 | 3055 | 482.97 |
| d2103 | 80,450 | 1/20 | 0.026 | 2785 | 310.39 |
| pr2392 | 378,032 | 1/20 | 0.008 | 2795 | 55.86 |
| pcb3038 | 137,694 | 0/20 | 0.017 | 3280 | 133.79 |
| fl3795 | 28,772 | 2/20 | 0.104 | 3755 | 690.39 |

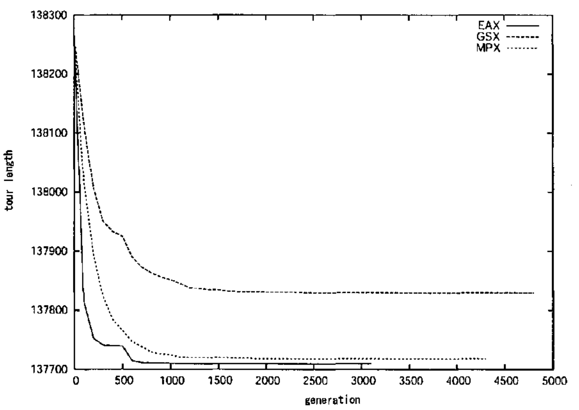

Fig. 6 Change of tour length(pcb3038)

the convergence of EAX is earlier and comparatively high quality solutions is found. It is thought that the heredity of the characteristics of EAX is excellent. EAX is understood that a calculation burden is large compared with MPX and GSX. The reason is probable that $2^k$($k$ is number of AB-cycle) candidates of offspring are generated from two parents with EAX. Improvement to the selection method and the division method of AB-cycle is important in order to reduce execution time.

## 4 Conclusions

We evaluate the proposed HGA combined EAX and heuristics. We perform a comparison experiment with three crossovers (EAX, MPX, and GSX) in order to evaluate the quality of a solution. In case the number of cities is large i.e. 3038-3795, the convergence of EAX is earlier and comparatively high quality solutions is found. The excellent search ability of EAX was confirmed by a comparison experiment. But, EAX is understood that a calculation burden is large compared with MPX and GSX. We want to improve the selection method and the division method of AB-cycle in order to reduce execution time.

## References

[1] Yoshitugu Yamamoto and Mikio Kubo, *Invitation to Traveling Salesman Problem*, Asakura Tosho Publishing, 1997 (in Japanese).

[2] Mikio Kubo, "Invitation to Traveling Salesman ProblemI,II,III," *Communications of the Operations Research Society of Japan*, Vol. 39, No. 1,2,3, pp. No1:25–31, No2:91–96,No3:156–162, 1994 (in Japanese).

[3] Yuichi Nagata and Shigenobu Kobayashi, "The Proposal and Evaluation of a Crossover for Traveling Salesman

Problems : Edge Assembly Crossover," *Journal of JSAI*, Vol. 14, No. 5, pp. 88–99, 1999.

[4] Yuichi Nagata and Shigenobu Kobayashi, "Edge Assembly Crossover:A High-power Genetic Algorithm for Traveling Salesman Problem," *Procceeding of the 7th International Conference on Genetic Algorithms*, pp. 450–457, 1997.

[5] Hung Dinh Nguyen and Ikuo Yoshihara and Kunihito Yamamori and Moritoshi Yasunaga, "Greedy Genetic Algorithms for Symmetric and Asymmetric TSPs," *IPSJ Transactions on Mathematical Modeling and Its Applications*, Vol. 43, No. SIG10(TOM7), pp. 165–175, 2002.

[6] Hiroaki Kitano, *Genetic Algorithm*, Sangyo Tosho Publishing, 1993 (in Japanese).

[7] Heinz Mühlenbein, "Evolution in Time and Space - The Parallel Genetic Algorithm," *Foundations of Genetic Algorithms*, pp.316-337, 1991.

[8] Masayuki Yamamura and Takahisa Ono and Shigenobu Kobayashi, "Character-Preserving Genetic Algorithms for Traveling Salesman Problem," *Journal of JSAI*, Vol. 7, No. 6, pp. 1049–1059, 1992 (in Japanese).

[9] Darrell Whitley, "The GENITOR Algorithm and Selection Pressure: Why Rank-Based Allocation of Reproductive Trials is Best," *Proc. 3rd Int'l Conf. on Genetic Algorithms*, pp.116-121, 1989.

## Appendices

The following operations generate an intermediate individual.

### A.1 Definitions

**(Def. 1)** Graph $G_{AB}$

Graph $G_{AB}$ is defined as follows.

$$G_{AB}=(V,E_A \cup E_B)$$

$E_A$:an set of edges that constructs tourA

$E_B$:an set of edges that constructs tourB

**(Def. 2)** AB-cycle

AB-cycle is a closed path which consists of even edge which follows $e_A \in E_A$ and $e_B \in E_B$ by turns on $G_{AB}$.

**(Def. 3)** Exchangeable Edge Set (E-set)

E-set has the equal number of the edge belonging to $E_A \cap D$ and the equal number of the edge belonging to $E_B \cap D$ on each point on subset $D$ of $E_A \cup E_B$.

### A.2 Outline of EAX

**Step1** Generate AB-cycle from $G_{AB}$.(Fig.8)

**Step2** Generate E-set from combine of arbitrary AB-cycle.

**Step3** Generate an intermediate individual by applying the E-set to tourA.(Fig.9)

**Step4** Modify the intermediate individual.

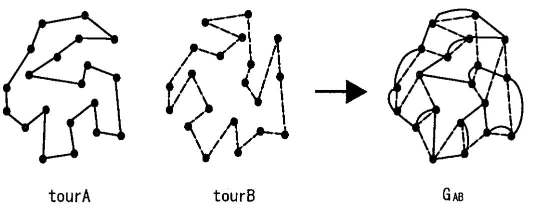


Fig. 7 Parents(tourA,tourB) and $G_{AB}$

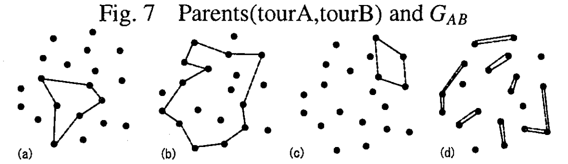


Fig. 8 An example of AB-cycles obtained from $G_{AB}$

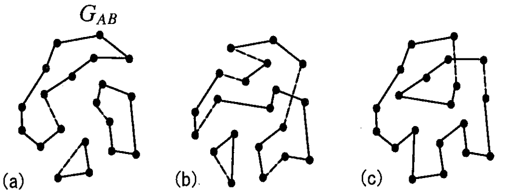


Fig. 9 Examples of intermediate individuals obtained from E-set

### A.3 AB-cycle

**(1)** Generation of AB-cycle

A path is formed by choosing $e_A \in E_A$ and $e_B \in E_B$ at random by turns on $G_{AB}$. If the closed path which consists of even edge is formed, it is extracted as AB-cycle. This operation is continued until all the edge on $G_{AB}$ is lost.

**(2)** Selection of AB-cycle

Total of intermediate individual which can be constituted by combining $e_A \in E_A$ with $e_B \in E_B$ correspond one-to-one to combination of AB-cycle. In other words, If $k$ AB-cycle is generated, $2^k$ intermediate individual can be generated from tourA and tourB. Therefore, it is necessary to choose AB-cycle. Then, the following two selection methods are proposed.

**(a)** Random Selection

AB-cycle is chosen at random with probability 0.5.

**(b)** Heuristic Selection

Heuristic method that considers a balance between exploitation and exploration is proposed. We use $\alpha = 1$.

$$gain_i = \sum_{e \in AB-cycle_i \cap E_A} w(e) - \sum_{e \in AB-cycle_i \cap E_B} w(e)$$

$$f_{A_i} = \sum_{e \in AB-cycle_i \cap E_A} f(e), \quad f_{B_i} = \sum_{e \in AB-cycle_i \cap E_B} f(e)$$

$$\bar{f_A} = \frac{\sum_i \sum_{e \in AB-cycle_i \cap E_A} f(e)}{\sum_i n_i/2}$$

$$\bar{f_B} = \frac{\sum_i \sum_{e \in AB-cycle_i \cap E_B} f(e)}{\sum_i n_i/2}$$

$$div_i = (f_{A_i} - \bar{f_A} \times n_i/2) - (f_{B_i} - \bar{f_B} \times n_i/2)$$

$$sum\_gain\_plus = \sum_{i \ s.t. \ gain_i \geq 0} gain_i$$

$$sum\_div\_plus = \sum_{i \ s.t. \ div_i \geq 0} div_i$$

$$GAIN_i = \frac{gain_i}{sum\_gain\_plus}$$

$$DIV_i = \frac{div_i}{sum\_div\_plus}$$

$$F_i = GAIN_i + \alpha \times DIV_i$$

$AB-cycle_i$ : $i$-th $AB-cycle(i = 1, 2, \cdots, k)$

$e$ : an edge

$w(e)$ : weight of $e$

$f(e)$ : ratio of $e$ in population

**A.4** Exchangeable Edge Set (E-set)

The combination of arbitrary AB-cycle can constitute E-set. An intermediate individual is generated by applying E-set to tourA as follows.

$$E_A \rightarrow (E_A \cap \bar{D}) \cup (E_B \cap D)$$