

개체 분포의 다양성을 유지시키는 GA를 위한 스키마 추출 기법

The Schema Extraction Method for GA Preserving Diversity of the Distributions in Population

조용균, 장성환, 강 훈
중앙대학교 전자전기공학부

Yong-Gun Jo, Sung-Hwan Jang and Hoon Kang
School of Electrical & Electronics Engineering,
College of Engineering, Chung-Ang University,
221 HeukSuk-Dong, DongJok-Gu, Seoul 156-756, Korea
E-mail : infinity@sirius.cie.cau.ac.kr, bigloud@sirius.cie.cau.ac.kr & hkang@cau.ac.kr

Abstract

In this paper, we introduce a new genetic reordering operator based on the concept of schema to solve the Traveling Salesman Problem(TSP). Because TSP is a well-known combinatorial optimization problem and belongs to a NP-complete problem, there is a huge solution space to be searched. For robustness to local minima, the operator separates selected strings into two parts to reduce the destructive probability of good building blocks. And it applies inversion to the schema part to prevent the premature convergence. At the same time, it searches new spaces of solutions. In addition, we have the non-schema part to be applied to inversion as well as for robustness to local minima. By doing so, we can preserve diversity of the distributions in population and make GA be adaptive to the dynamic environment.

1. Introduction

Given N cities and distances among them, the goal of the Traveling Salesman Problem (TSP) is to find an ordering of cities that makes the shortest tour for a virtual salesman to travel each city only once and to return to the city from which he started. TSP is a representative combinatorial optimization problem and belongs to a class of problems known as NP-complete[1]. It minimizes the tour length a salesman travels but the number of possible tours existing in the search space of solutions increases exponentially as one city is added. The various researches have been carried out to find

near optimal solution. Simulated annealing[2] executes iterative global search to solve TSP according to the schedule of the temperature transition and cooling rates. The Hopfield network[3] defines neurons as cities and minimizes the energy function inducted from the constraints of TSP by recall process. Genetic Algorithms (GAs) search the space of solutions by producing more offsprings whose parents have higher fitness values, based on the mechanics of natural selection and natural genetics[4, 5].

GAs, first invented by John Holland, mimic the mechanism of biological evolution. They select good strings of chromosomes, i.e. ones with higher fitness to the environment or given problem, then produce better offsprings by genetic operators: reproduction, crossover, and

This paper is supported by Brain Science Research Center of Korean Ministry of Science and Technology under the grant no. 98-J04-01-01-A-07

mutation. Reproduction has good parent strings bear more children than others, crossover mixtures the good parts of parents to reproduce fitter offsprings than former generations. Mutation protects from losing some potentially useful genetic material, so that it provides possibility to find the optimal solution and the variety of population preserved and improved. Each strings in population is regarded as one search point in the search space. As generation goes, the initial population evolve in the way to contain solutions close to optimum. Although GAs have simple structures, they show the powerful performance over the other algorithms. Holland explained this through schema theorem: short, low- order, above-average schemata receive exponentially increasing trials in subsequent generations. Until recently this theoretical foundation was the basis of almost all subsequent theoretical work on genetic algorithms.

In general TSP with GAs, the order of visiting cities is represented as chromosome and an allele's meaning is position-independent. Therefore, crossover and mutation, the basic operators of the simple GA, cannot be used because they will destroy the important orderings with high probability. For this reason, a lot of new GA operators were introduced: inversion, partially matched crossover(PMX), order crossover(OX), cycle crossover(CX), etc. We call them reordering operators in general. Their performances vary along their own probabilities. Inversion shows the best performance among them but still can not guarantee the global minimum.

In this paper, we introduce a new genetic reordering operator, Schema Extraction(SE) method. It separates selected strings into two parts, schema and non-schema to reduce the destructive probability of good building blocks. And it applies inversion to the schema part to prevent the premature convergence. At the same time, it searches new spaces of solutions. In addition to, we have the non-schema part to be applied to inversion as well as for robustness to local minima. By doing so, we can preserve diversity of the distributions in population and make GA be adaptive to the dynamic environment. We simulated it to compare with inversion and PMX and show its performance.

II. Schema Theorem

GAs are simple for us to understand conceptually and can be implemented by computer programs for desired solutions easily. However, their behavior can be complicated and there are many considerations for applying GAs to real problems. They include coding methods from which the right strings will be represented, fitness evaluation methods, construction methods of initial population, and the proper probabilities of crossover and mutation. Nevertheless, GAs work great by discovering, emphasizing, and recombining good building blocks of solutions in a highly parallel fashion. This is known as the Building Block Hypothesis[5]. Holland introduced the notion of schemata to formalize building blocks. A schema is a set of bit strings that can be described by a template made up of alleles and asterisks. Suggesting the Schema

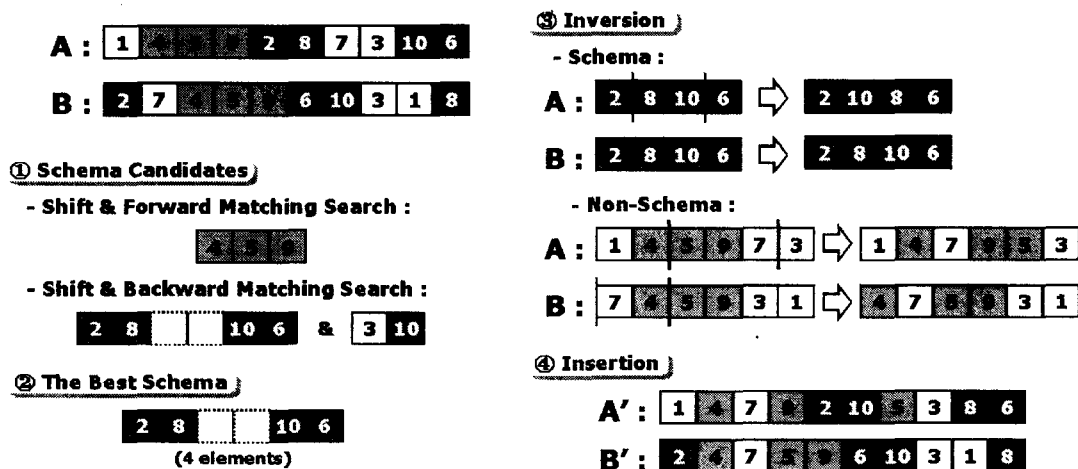


Fig. 1 A schematic diagram of Schema Extraction

Theorem, he showed that short, low-order schemata whose average fitness remains above the mean of population will receive exponentially increasing numbers of their instances over generation, since the number of instances of those schemata that are not disrupted and remain above average in fitness increases by a factor of $f(H)/\bar{f}$ (i.e., the ratio of the average fitness of the schema to the average fitness of the population) at each generation.

III. Schema Extraction

In GAs, We should use special genetic operators so called reordering operators to solve TSP since an allele's meaning is position-independent. The representative reordering operators are inversion, PMX, OX, CX. Among these operators, inversion has the best performance in time and the optimal solution. PMX and CX appear to have premature convergence problem. OX shows impressive results at some probabilities but its errors are greater than inversion.

We suggest Schema Extraction operator which extracts a set of matching genes from two selected combinatorial strings. The selected strings are possible to have good results so that we can consider the extracted set as schema candidates. Among these ones, the operator chooses the best one and separates it from strings. And inversion operator is applied to it and the rest ones, then the outputs are reinserted into the original positions. Fig. 1 depicts the schematic diagram of SE and The procedure of SE is shown as follows.

Since each element is independent of its position, it is hard to obtain the best schema just by comparing two strings. All the possible cases must be considered. So we put shift, forward- matching and backward-matching process into the algorithm. After this stage, we can collect schema candidates of two strings and elect the best schema among them according to the conditions as follows.

- ① the number of defined bits is large.
- ② the fitness of schema is high.

The condition 1 has priority over the condition 2. As shown in Fig.1, inversion is applied to both schema and non-schema as the secondary operator. Inversion for schema part provides the opportunity to jump out of local minimum. On

```

procedure SchemaExtraction()
begin
  do
    shift(p1 & p2)
    forward_matching(p1, p2)
    backward_matching(p1, p2)
    update_number(schema)
    update_distance(schema)
  while (not end of chromosomes)
    compare number of schema with
      the max. number
    update_max_number()
    compare distance of schema with
      the min. distance
    update_min_distance()
    reinsert(inversion(schema) &
      inversion(non-schema))
end

```

the other hand, inversion for non-schema gives chances to obtain good allele orderings that will subsequently permit more efficient propagation of building blocks. By doing so, SE acquires robustness for optimal solution.

IV. Simulation Results

Three probabilities are required for SE: probability of applying SE to the selected strings, of inversion for schema, and of inversion for non-schema. For the 1st probability, we restricted its value under heuristic number of 30% because of preventing the average fitness of population from remaining steady-state or decreasing. We simulated TSP solving by SE and inversion with 1000 population over 1000 generation.

Fig. 2. and Fig. 3. shows the results for inversion and SE, respectively. The enormous experiments revealed that the proper probabilities of SE for 25-city TSP was (30%,50%,50%). The simulation results for 30 cities are shown in Fig. 4 and Fig. 5 as well. In the cases of inversion, 20% made best results. Fig. 3 shows SE escaped from local minimum around generation 850. Taking a careful look at the histograms(the 3rd pictures) from Fig. 2 to Fig.5 lead us to find out that the distribution of population solved by inversion was more concentrated than that solved by SE. That means SE preserves diversity of distributions of population so that it is more adaptive to the

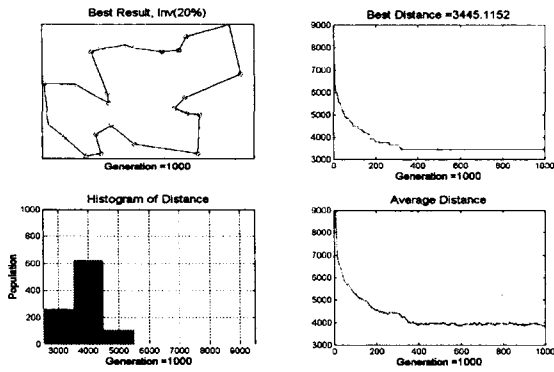


Fig. 2. 25-city TSP solved by inversion

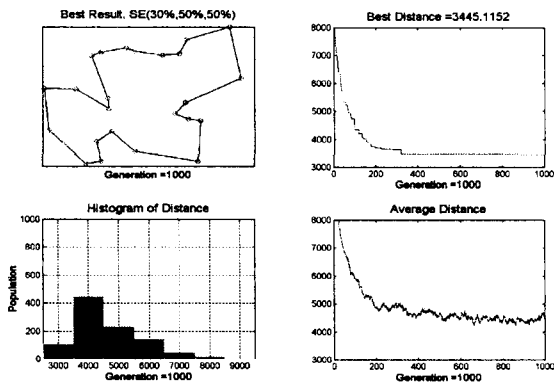


Fig. 3. 25-city TSP solved by SE

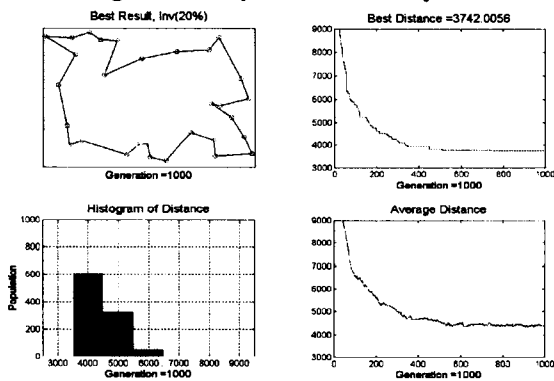


Fig. 4. 30-city TSP solved by inversion

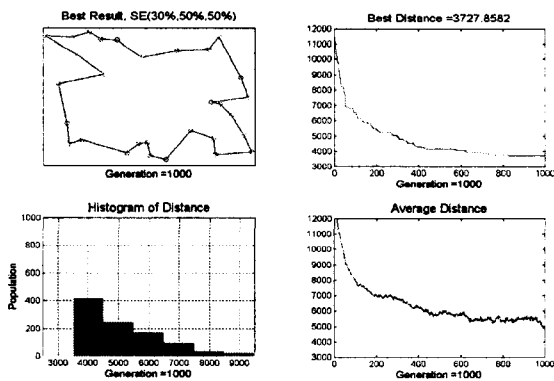


Fig. 5. 30-city TSP solved by SE

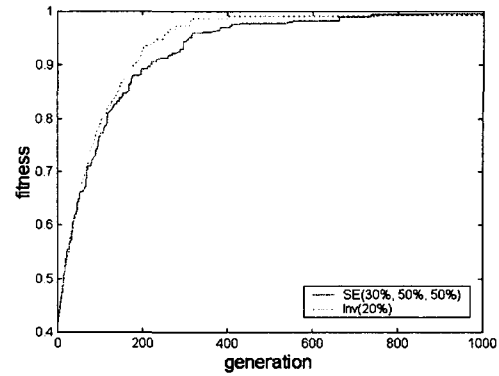


Fig. 6. The fitness comparison between SE and inversion

dynamic environment and the robustness to the local minimum is obtained. Fig. 6. represents the fitness comparison between SE(30%,50%,50%) and inversion(20%). This results obtained by averaging the 5 trials. It shows that SE finds more optimal solution than inversion and possibility of premature convergence of inversion.

V. Conclusion

In this paper, we suggest a new genetic operator, Schema Extraction method, based on the concept of schema. By separating schema and non-schema, more instances of possible building blocks are preserved and reproduced. The simulation results shows SE preserves diversity of distributions of populations and provides fast initial convergence and robustness to local minima.

References

- [1] Garey MR, Johnson DS(1979), Computers and Intractability: a guide to the theory of NP-completeness. Freeman, New York
- [2] Kirkpatrick S, Gelatt CD, Vecchi MP(1983), Optimization by simulated annealing, Science 220: 671-680
- [3] Hopfield JJ, Tank DW(1985), 'Neural' computation of decisions in optimization problems. Biological Cybernetics, 52:141-152
- [4] Holland JH(1975), Adaptation In Natural and Artificial Systems, The University of Michigan Press, Ann Arbor
- [5] Goldberg DE(1989), Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, MA