New Usage of Sammon's Mapping for Genetic Visualization

Yong-Hyuk Kim and Byung-Ro Moon

School of Computer Science & Engineering, Seoul National University
Shillim-dong, Kwanak-gu, Seoul, 151-742 Korea
{yhdfly, moon}@soar.snu.ac.kr

Abstract. It is a hard problem to understand the fitness landscape of a problem as well as the evolution of genetic algorithms. For the purpose, we adopt Sammon's mapping for the investigation. We demonstrate its usefulness by applying it to the graph partitioning problem which is a well-known NP-hard problem. Also, through the investigation of schema traces, we explain the genetic process and the reordering effect in the genetic algorithm.

1 Introduction

An NP-hard problem such as graph partitioning problem or traveling salesman problem (TSP) has a finite solution set and each solution has a cost. Although finite, the problem space is intractably large even for a small but nontrivial problem. A number of studies about the ruggedness and the properties of problem search spaces were done. Weinberger [17] conjectured that, if all points on a fitness landscape are correlated relatively highly, the landscape is bowl shaped. Boese *et al.* [1] suggested that, through measuring cost-distance correlation for the TSP and the graph partitioning problem, the cost surfaces are globally convex. Jones and Forrest [11] introduced fitness-distance correlation as a measure of search difficulty. Good insight into the problem space can provide a motivation for a good search algorithm [1]. We examine the problem space and hope to get some insight into the problem space.

For NP-hard problems with intractably large problem space, it is almost impossible to find an optimal solution by exhaustive or simple search methods. Thus, in case of NP-hard problems, heuristic algorithms or meta-heuristics are used. The genetic algorithm (GA) is one of the most powerful search methods among them. A number of studies for understanding GA's working mechanism were done. These include schema theorem [10], Royal Road function [13], etc.

Visualization is one of the most basic tools for studies of search spaces. A notable method for fitness landscapes is the plotting of fitness-distance correlation [1]. For GA visualization, the most popular method is the fitness flow over time as in many GA papers. Another wholesale method is the population data matrix[1] for identifying population features. In this paper, we propose new visualization

[1] In the matrix, the entire population is displayed in textual form.

E. Cantú-Paz et al. (Eds.): GECCO 2003, LNCS 2723, pp. 1136–1147, 2003.

techniques primarily using Sammon's mapping. We analyze the problem space for graph partitioning more elaborately. We visualize the solutions associated with the genetic search. We also trace schemata and analyze them.

The remainder of this paper is organized as follows. In Section 2, we summarize graph partitioning problem and Sammon's mapping which is used as a major tool for visualization in this paper. We analyze fitness landscapes for graph partitioning in Section 3. In Section 4, we provide some visualization for genetic algorithms. In Section 5, schema traces are visualized and analyzed. Finally, we make our conclusions in Section 6.

2 Preliminaries

2.1 Graph Partitioning

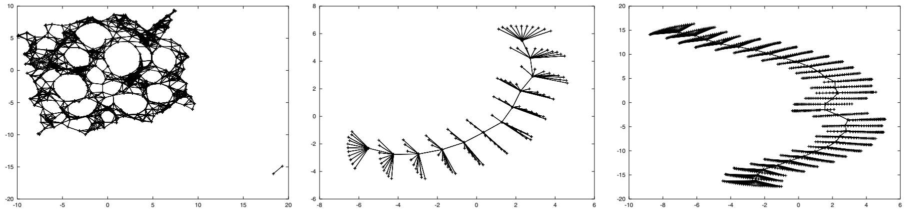
Let $G = (V, E)$ be an unweighted undirected graph, where V is the set of vertices and E is the set of edges. A bipartition (A, B) consists of two subsets A and B of V such that $A \cup B = V$ and $A \cap B = \phi$. The *cut size* of a bipartition is defined to be the number of edges whose endpoints are in different subsets of the bipartition. The bipartitioning problem is the problem of finding a bipartition with minimum cut size. If the difference of cardinalities between two subsets is at most one, the problem is called *graph bisection* problem. It is a representative NP-hard problem [8]. In this paper, we use three graphs from [3] (one geometric graph and two caterpillar graphs)[2] as test beds.

2.2 Sammon's Mapping

Sammon's mapping [16] is a mapping technique for transforming a dataset from a high-dimensional (say, m-dimensional) input space onto a low-dimensional (say, d-dimensional) output space (with $d < m$). The basic idea is to arrange all the data points on a d-dimensional output space in such a way that minimizes the distortion of the relationship among data points.

Sammon's mapping tries to preserve distances. This is achieved by minimizing an error criterion which penalizes the differences of distances between points in the input space and the output space. Consider a dataset of n objects. If we denote the distance between two points x_i and x_j in the input space by δ_{ij}

[2] The classes are briefly described below.
i) U$n.d$: A random geometric graph on n vertices that lie in the unit square and whose coordinates are chosen uniformly from the unit interval. There is an edge between two vertices if their Euclidean distance is t or less, where $d = n\pi t^2$ is the expected vertex degree.
ii) rcat.n: A caterpillar graph on n vertices. It is constructed out of a straight line (called the spine), where all the vertices on this line have degree two except the outermost two vertices. Each vertex on the spine is then connected to $\sqrt{n}$ new vertices, the legs of the caterpillar.



(a) U500.10 (b) rcat.134 (c) rcat.994

Fig. 1. Examples of Sammon's mapping

and the distances between x'_i and x'_j in the output space by d_{ij}, then Sammon's stress measure E is defined as follows:

$$E = \frac{1}{\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\delta_{ij}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \frac{(\delta_{ij} - d_{ij})^2}{\delta_{ij}} \quad .$$

The stress range is [0,1] with 0 indicating a lossless mapping. This stress measure can be minimized using any minimization technique. Sammon [16] proposed a technique called pseudo-Newton minimization, a steepest-descent method. The complexity of Sammon's mapping is $O(n^2m)$. There were several studies about Sammon's mapping [7] [5] [14].

The resulting output space depicts clusters of the input space as groups of data points mapped close to each other in the output space. Figure 1 shows Sammon's mapping of three graphs into 2-dimensional space. In the graphs, we defined the distance between two vertices to be the shortest path length between each other. We can observe that the mapping well accords with the characteristics of the graphs.

3 Fitness Landscapes

In this section, we first extend the experimentation of Boese *et al.* [1] to examine the local-optimum space. We mean by local-optimum space the space consisting of all local optima with respect to a local optimization algorithm. We then examine the distribution of local optima. In our experiments, we used a sufficiently large number of local optima. We do not care about solutions other than local optima. The Kernighan-Lin algorithm (KL) [12] was used for local optimization.

In the graph bisection problem for a graph $G = (V, E)$, each solution (A, B) is represented by a $|V|$-bits code. Each bit corresponds to a vertex in the graph. A bit has value 0 if the vertex is in the set A, and has value 1 otherwise. In this encoding, a vertex move in the solution changes the solution by one bit. Thus, it is natural to define the distance between two solutions by the Hamming distance. Formally, we define the *genotype distance* between two solutions as follows.

Definition 1 *Let the universal set U be $\{0,1\}^{|V|}$. For $\mathfrak{a}, \mathfrak{b} \in U$, we define the* genotype distance *between $\mathfrak{a}$ and $\mathfrak{b}$ as follows:*

$$d_g(\mathfrak{a}, \mathfrak{b}) = \mathfrak{H}(\mathfrak{a}, \mathfrak{b})$$

where $\mathfrak{H}$ is the Hamming distance.

However, if the genotype distance between two solutions is $|V|$, they are equal. We hence define the *phenotype distance* between two solutions as follows.

Definition 2 *Let the universal set U be $\{0,1\}^{|V|}$. For $\mathfrak{a}, \mathfrak{b} \in U$, we define the* phenotype distance *between $\mathfrak{a}$ and $\mathfrak{b}$ as follows:*[3]

$$d_p(\mathfrak{a}, \mathfrak{b}) = min(d_g(\mathfrak{a}, \mathfrak{b}), |V| - d_g(\mathfrak{a}, \mathfrak{b}))$$

where d_g is the genotype distance.

By the definition, $0 \leq d_p(\mathfrak{a}, \mathfrak{b}) \leq \lfloor |V|/2 \rfloor$ while $0 \leq d_g(\mathfrak{a}, \mathfrak{b}) \leq |V|$. In this paper, we use the phenotype distance d_p for the distance between two solutions unless otherwise noted.

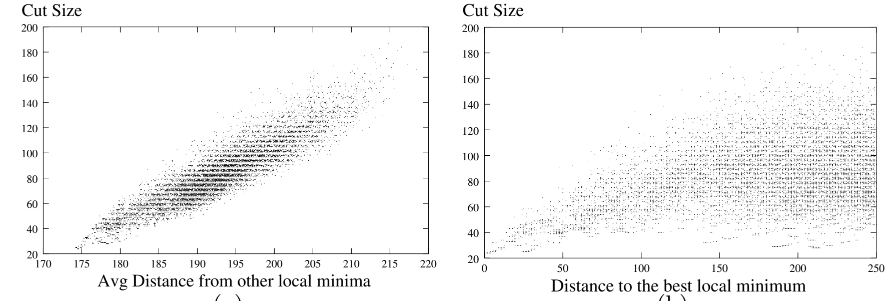
3.1 Cost-Distance Correlation

Given a set of local minima, Boese *et al.* [1] plotted, for each local minimum, i) the relationship between the cost and the average distance to all the other local minima, and ii) the relationship between the cost and the distance to the best local minimum. They performed experiments for the graph bisection and the traveling salesman problem, and showed that both problems have strong positive correlations for both i) and ii) in the above. This fact hints that the best local optimum is located near the center of local-optimum space. From their experiments, they conjectured that the cost surfaces of both problems are globally convex. In this subsection, we repeat their experiments for another graph.

The solution space for the experiment was selected as follows. First, we chose a large number of random solutions and obtained the corresponding set of local optima by locally optimizing them. Then, we removed the duplicated solutions in the set if any. Figures 2(a) and (b) show the plotting results with 9,302 local minima[4] for the graph U500.10. The correlation coefficient for the experiment i) was 0.91. It is consistent with Boese *et al.*'s results with strong cost-distance correlation.

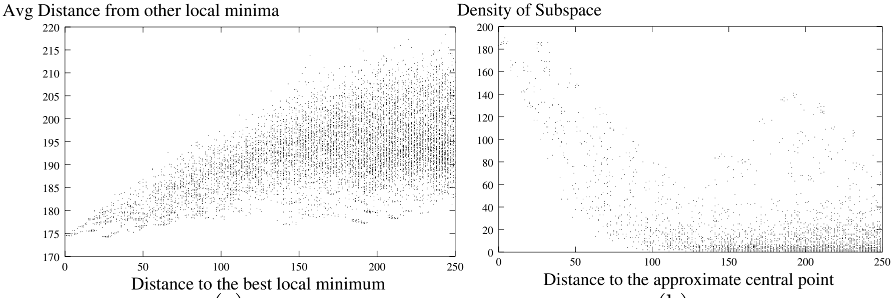
[3] Given an element $\mathfrak{a} \in U$, there is only one element such that it is different from $\mathfrak{a}$ and the distance d_p to $\mathfrak{a}$ is zero. If the distance between two elements is equal to zero, we define them to be in relation R. Then, the relation R is an *equivalence relation.* Suppose Q is the quotient set of U by relation R ($Q = U/R$), it is easily verified that (Q, d_p) is a *metric space.*

[4] There were 698 duplications among 10,000 local minima.



(a) (b)

Fig. 2. Fitness-distance correlation (U500.10)



(a) (b)

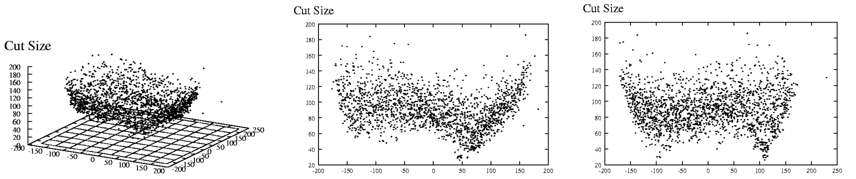
Fig. 3. Distribution of local minima (U500.10)

3.2 Distribution of Local Optima

As a result of the experiments of Boese *et al.* [1], we agree with the conjecture about the global convexity of local-optimum space but it is difficult to obtain further deduction. Figure 3(a) shows the relationship between the distance to the best local minimum and the average distance to the other local minima for each local minimum in the local-optimum space. In the figure, there are considerably many solutions such that they are far from the best solution but their average distances are small. This fact suggests that solutions may be clustered in more than one place.

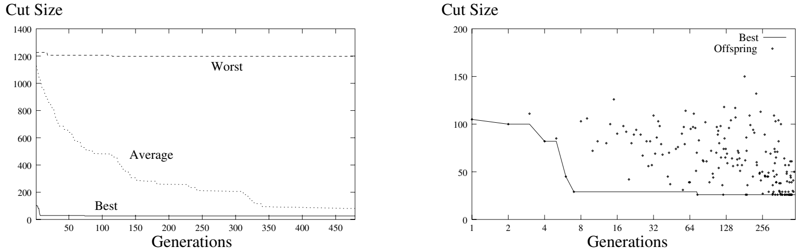
We devised another way to examine the distribution of local optima. For each solution $\mathfrak{s}$ in the problem space, we chose a ball centered at $\mathfrak{s}$ with radius r (here we set r to be $|V|/8$) and counted the number of local optima inside the ball. Figure 3(b) plots the densities of the balls. It shows that the density of local optima near the center of the problem space is remarkably high. Interestingly enough, one can also observe fairly high-density areas far from the center. It suggests the existence of "medium valleys"[5] or "small valleys." It can not be explained by the experimental methods such as [1].

[5] The relative notion to big valleys mentioned by Boese *et al.* [1].



(a) 3D (b) 2D projection (XZ plane) (c) 2D projection (YZ plane)

Fig. 4. Fitness landscape with Sammon's mapping (U500.10)



(a) Best, average, and worst (b) Best and offspring

Fig. 5. Traditional plotting (a hybrid GA on U500.10)

3.3 Visualization by Sammon's Mapping

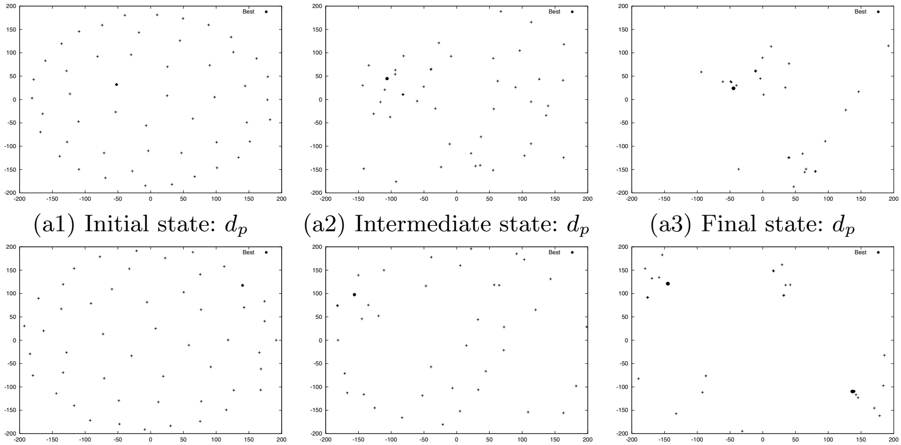
Sammon's mapping is a good visualization tool for multi-dimensional datasets. Local-optimum spaces are also good candidates for Sammon's mapping. Sammon's mapping of the local-optimum space helps visually understanding the problem space. Figure 4(a) shows Sammon's mapping of the local-optimum space for the graph U500.10. The local optima were Sammon-mapped on the XY plane. The Z-axis means the cut size. Figures 4(b) and (c) indicate the projected spaces of Figure 4(a) into XZ plane and YZ plane, respectively. We can observe the fitness landscape with respect to Sammon's mapping. In this case, the result suggests the existence of valleys in more than one place.

4 Visualization of a Steady-State Genetic Search

4.1 Previous Studies

Traditionally, the fitness-generation plotting has been popular for the visualization of genetic process. A great number of papers include these plottings to visualize their genetic search process. Figure 5 shows examples of the traditional plotting.

Recently, Dybowski *et al.* [6] proposed a GA visualization method using Sammon's mapping. There have been a number of studies about GA visualization using Sammon's mapping [4] [15]. They presented initial studies about small problems. An extensive survey of GA visualization techniques appeared in [9]. In this paper, we focus only on the visualization by Sammon's mapping.



(a1) Initial state: d_p (a2) Intermediate state: d_p (a3) Final state: d_p

(b1) Initial state: d_g (b2) Intermediate state: d_g (b3) Final state: d_g

Fig. 6. 2D mapping with different distances (hybrid GBA on U500.10)

4.2 Extended Experiments

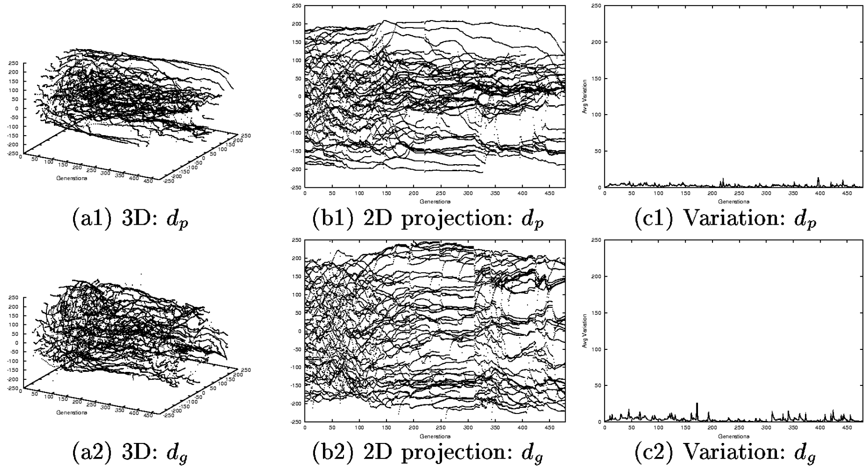
We extend the works of Dybowski *et al.* [6]. Using Sammon's mapping, they showed that population converges into one or more clusters. They used the Hamming distance (called the genotype distance in this paper) for the distance in the input space of binary chromosomes.

In this subsection, we provide two experiments for visualization with a GA. First, we make experiments with different distance measures. Then, we provide a new technique for a steady-state GA to visualize the change of population in the genetic search process.

We used the Genetic Bisection Algorithm (GBA) [3] for graph bisection problem. It is a steady-state GA with population size 50, 5-point crossover, adjacent repair[6], and GENITOR-style replacement [18]. If GBA is hybridized with KL local optimization, it is denoted by KL-GBA. We use KL-GBA on the graph U500.10 for the experiments in this subsection.

Given a space, various distance measures can be defined. The properties of the space are largely dependent on the distance measure. Particularly, in Sammon's mapping, if the output space is a metric space, the input space need to be a metric space to minimize the stress measure. We used two distance measures: the genotype distance d_g and the phenotype distance d_p defined in Section 3. With d_p as the distance measure, Figures 6(a1), (a2), and (a3) show 2-dimensional mapped population spaces at initial, intermediate, and final generation, respectively. Figures 6(b1), (b2), and (b3) show the results with d_g as the distance

[6] After the crossover, an offspring may not satisfy the balance requirement. It then selects a random point on the chromosome and changes the required number of 1's to 0's (or 0's to 1's) starting at that point on to the right. This adjustment produces some mutation effect.



(a1) 3D: d_p (b1) 2D projection: d_p (c1) Variation: d_p

(a2) 3D: d_g (b2) 2D projection: d_g (c2) Variation: d_g

Fig. 7. Visualization of genetic search process with different distances (hybrid GBA on U500.10)

measure. When we used the genotype distance d_g, the population converged into four clusters. However, with the phenotype distance d_p, the population converged into roughly two clusters. From the fact that one phenotype matches two genotypes in graph partitioning, it seems to be reasonable. We observed two notable valleys in this problem space in Section 3.2 and Section 3.3. It is interesting that the population of GA converged into two clusters. The plotting by Sammon's mapping can be extended to three dimensions. We omit the results here.

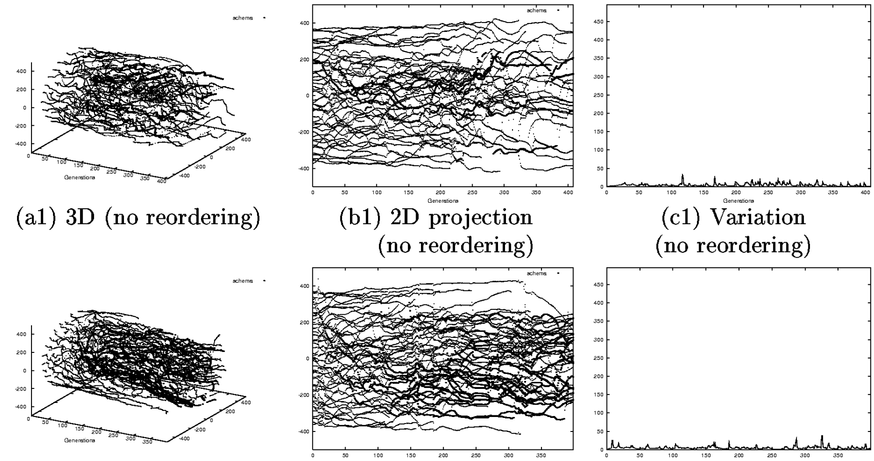
In the next experiment, we visualize the change of population in the process of a steady-state GA. Generally, Sammon's mapping starts with random initial positions of n objects. Iteratively, it optimizes the stress measure E. A steady-state GA typically generates only one offspring per iteration. It does not make a rapid change per population. Hence, if the positions of the previous generation are used for the initial positions of the next-generation Sammon's mapping, the positions would change steadily over the generations. This makes it possible to visualize solutions over the genetic search process. Figure 7 shows the visualization of a genetic process. Figure 7(a) visualizes the time-varying dataset such that X-axis is the time and YZ plane represents the 2-dimensional Sammon's mapping. Figure 7(b) shows its projection into XZ plane. Figure 7(c) gives the average variation between the previous population positions and the current population positions. More formally, in the i^{th} generation, average variation V_i is defined to be $V_i = \frac{1}{K}\sum_k ||s_k(i) - s_k(i-1)||$, where $s_k(i)$ is the mapped vector of the k^{th} chromosome in the i^{th} generation and K is the population size. At a generation with large variation, which probably suggests the occurrence of an important solution, the continuity gets broken. From this visualization, we can also observe the process of population convergence. It helps us to understand the genetic search process more elaborately. It is notable that this visualization is related to average cost plotting of Figure 5(a). In the range with stable average costs, the mapped data also shows minor changes (e.g., see the range [150, 310] in Figure 5(a) and Figure 7). The phenomenon of punctuated equilibria may also be observable by this plotting.

5 Schema Traces

A schema is a pattern of bit strings consisting of specific symbols and asterisks; here, specific symbols represent the pattern and the asterisks represent "don't care" positions. A genetic algorithm starts with a group of random initial solutions. Of course, the quality of the solutions is low in the early stages of the genetic algorithm. However, most low-quality solutions contain some schemata common to high-quality solutions. The crossover operators of genetic algorithms generate larger schemata by juxtaposition of smaller schemata. It is important to preserve valuable schemata. A schema is prone to be destroyed by crossover operators if the positions forming the schema are scattered.

Generally speaking, it is not easy to know high-quality schemata in a problem. However, for some problems, it is possible to find high-quality schemata. Specially, in Royal Road function [13], all of the desired schemata are given in its description. We can also find high-quality schemata in some instances of graph partitioning. Caterpillar graphs are good examples. It is clear from their Sammon's mapping (see Figures 1(b) and (c)). In this subsection, we provide the visualization of high-quality schema traces for a graph partitioning problem and a Royal Road function.

First, we compare KL-GBA with BFS-KL-GBA on the graph rcat.994 (see Figure 1(c)). KL-GBA was introduced in Section 4.2. BFS-KL-GBA is an approach proposed in [2] for the purpose of transforming the shapes of valuable schemata to those advantageous for survival by using Breadth-First Search (BFS) reordering. We selected a schema consists of 156 vertices. Figure 8 shows the schema traces in the genetic search process (upper row with KL-GBA and lower row with BFS-KL-GBA). A bold dot represents the presence of the schema in a solution. A bold line mostly means the continual presence of the schema. One can observe remarkable difference between the two algorithms. Not only did KL-GBA show low frequency of schema creation, it also showed a low rate of schema survival. On the contrary, BFS-KL-GBA showed a high rate of schema survival as well as high frequency of schema creation. Since BFS reordering tends to shorten the defining lengths of high-quality schemata, the survival probabilities of those schemata become high through crossovers [3]. Without reordering, despite its early appearance, the schema did not spread all over the population as steadily as the reordered version. One can observe a high rate of schema distinction. On the contrary, the reordered version showed fairly stable preservation of the schema.



(a1) 3D (no reordering)

(b1) 2D projection (no reordering)

(c1) Variation (no reordering)

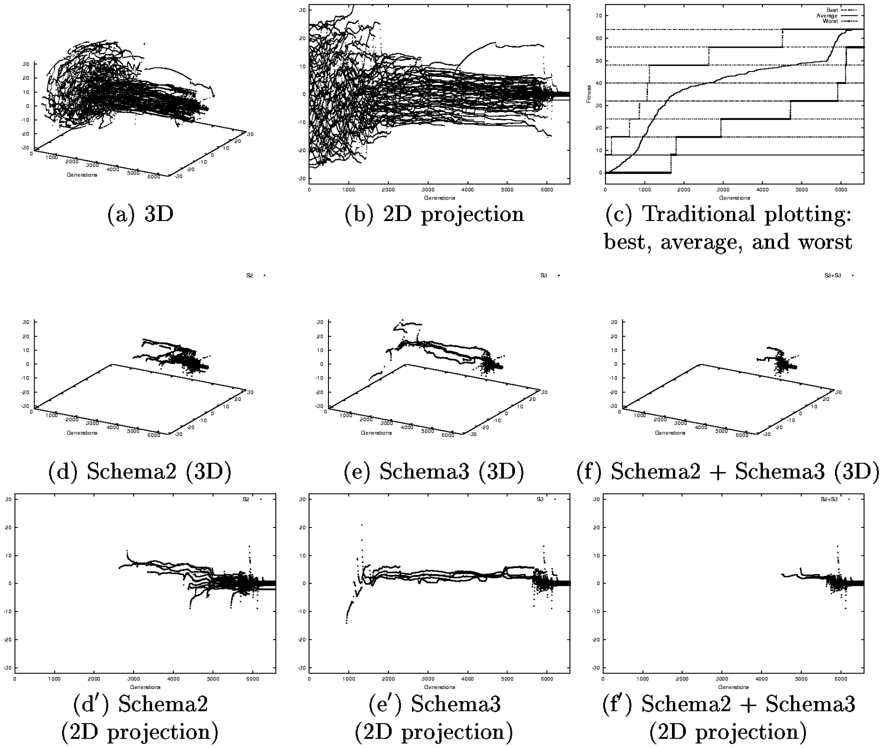
(a2) 3D (BFS reordering)

(b2) 2D projection (BFS reordering)

(c2) Variation (BFS reordering)

Fig. 8. Schema and reordering (hybrid GBA on rcat.994)

In the next experiment, we observe the schema traces with a 64-bit Royal Road function. The fitness of a chromosome is determined by the presence of predefined 8^{th} order schemata. It defines a tailor-made fitness landscape for GA's search and provides an ideal laboratory for studying GA's behavior. Dynamics of the search process can be studied by tracing individual schemata. We used a steady-state GA with population size 50, 1-point crossover, 0.5% mutation probability per bit, and GENITOR-style replacement. The Hamming distance is used as the distance measure. Figure 9 shows Sammon's mappings over the generations ((a) and (b)), a traditional plotting (c), and the schema traces ((d), (e), and (f)). Figures 9(a) and (b) clearly reflect strong convergence. It is surprising that the average fitness value nearly reflects the distribution of the population (compare figure (b) with the average line in figure (c)). The second and third rows of Figure 9 show the traces of two low-order high-quality schemata ((d) and (e)) and the high-order schema merging them (f). It visualizes only the individuals containing the schema. Here, a dotted line does not mean the discontinuity of schema presence but usually corresponds to a new appearance of an individual containing the schema. One can observe that schema2 and schema3 first appeared at around 2500^{th} and 1000^{th} generation, respectively, and they were successfully combined to a large schema at around 4500^{th} generation. Although schema3 is appeared earlier than schema2, schema2 spreads over the population faster than schema3. Figures (d), (e), and (f) visualize a process of GA by tracing the lives of particular schemata. Figures (d′), (e′), and (f′) are the 2D projections of figures (d), (e), and (f), respectively.



(a) 3D

(b) 2D projection

(c) Traditional plotting: best, average, and worst

(d) Schema2 (3D)

(e) Schema3 (3D)

(f) Schema2 + Schema3 (3D)

(d′) Schema2 (2D projection)

(e′) Schema3 (2D projection)

(f′) Schema2 + Schema3 (2D projection)

Fig. 9. Royal Road function with 8 schemata

6 Conclusions

Our approach goes beyond those of Boese *et al.* [1] and Dybowski *et al.* [6]. To get insights into fitness landscapes and GA's working mechanism, we introduced visualization techniques using Sammon's mapping and analyzed various experimental results. A steady-state GA for graph partitioning was mainly used in this paper. We could obtain some useful insights from the visualization. They could not be explained by previous visualization experiments. Our approach will be also useful for other optimization problems.

Sammon's mapping is one of the possible mapping methods. We may consider other mapping methods. It is of particular interest as well to investigate the visualization with respect to genetic operators.

Acknowledgments. This work was partly supported by KOSEF through Statistical Research Center for Complex Systems at Seoul National University (SNU) and Brain Korea 21 Project. The RIACT at SNU provided research facilities for this study.

References

1. K. D. Boese, A. B. Kahng, and S. Muddu. A new adaptive multi-start technique for combinatorial global optimizations. *Operations Research Letters*, 15:101–113, 1994.
2. T. N. Bui and B. R. Moon. Hyperplane synthesis for genetic algorithms. In *Fifth International Conference on Genetic Algorithms*, pages 102–109, July 1993.
3. T. N. Bui and B. R. Moon. Genetic algorithm and graph partitioning. *IEEE Trans. on Computers*, 45(7):841–855, 1996.
4. T. D. Collins. Genotypic-space mapping: Population visualization for genetic algorithms. The Knowledge Media Institute, The Open University, Milton Keynes, UK, Technical Report KMI-TR-39, 30th September 1996.
5. D. De Ridder and R.P.W. Duin. Sammon's mapping using neural networks: a comparison. *Pattern Recognition Letters*, 18(11–13):1307–1316, 1997.
6. R. Dybowski, T.D. Collins, and P. Weller. Visualization of binary string convergence by Sammon mapping. In *Fifth Annual Conference on Evolutionary Programming*, pages 377–383, 1996.
7. W. Dzwinel. How to make Sammon mapping useful for multidimensional data structures analysis. *Pattern Recognition*, 27(7):949–959, 1994.
8. M. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, 1979.
9. E. Hart and P. Ross. GAVEL – a new tool for genetic algorithm visualization. *IEEE Transactions on Evolutionary Computation*, 5(4):335–348, 2001.
10. J. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.
11. T. Jones and S. Forrest. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In *Sixth International Conference on Genetic Algorithms*, pages 184–192, 1995.
12. B. Kernighan and S. Lin. An efficient heuristic procedure for partitioning graphs. *Bell Systems Technical Journal*, 49:291–307, Feb. 1970.
13. M. Mitchell, S. Forrest, and J. H. Holland. The royal road for genetic algorithms: Fitness landscapes and GA performance. In *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, pages 245–254, Cambridge, MA, 1992. MIT Press.
14. E. Pekalska, D. De Ridder, R.P.W. Duin, and M.A. Kraaijveld. A new method of generalizing Sammon mapping with application to algorithm speed-up. In *Fifth Annual Conference of the Advanced School for Computing and Imaging*, pages 221–228, 1999.
15. H. Pohlheim. Visualization of evolutionary algorithms – set of standard techniques and multidimensional visualization. In *Genetic and Evolutionary Computation Conference*, pages 533–540, 1999.
16. J. W. Sammon, Jr. A non-linear mapping for data structure analysis. *IEEE Transactions on Computers*, 18:401–409, 1969.
17. E. D. Weinberger. Fourier and Taylor series on fitness landscapes. *Biological Cybernetics*, 65:321–330, 1991.
18. D. Whitley and J. Kauth. GENITOR: A different genetic algorithm. In *Rocky Mountain Conference on Artificial Intelligence*, pages 118–130, 1988.