

New Topologies for Genetic Search Space

Yong-Hyuk Kim

School of Mechanical & Aerospace Engineering
Seoul National University
Sillim-dong, Gwanak-gu, Seoul, 151-744 Korea
yhdfly@soar.snu.ac.kr

Byung-Ro Moon

School of Computer Science & Engineering
Seoul National University
Sillim-dong, Gwanak-gu, Seoul, 151-744 Korea
moon@soar.snu.ac.kr

ABSTRACT

We propose three distance measures for genetic search space. One is a distance measure in the population space that is useful for understanding the working mechanism of genetic algorithms. Another is a distance measure in the solution space for K -grouping problems. This can be used for normalization in crossover. The third is a level distance measure for genetic algorithms, which is useful for measuring problem difficulty with respect to genetic algorithms. We show that the proposed measures are metrics and the measures are efficiently computed.

Categories and Subject Descriptors

G.2.3 [Mathematics of Computing]: DISCRETE MATHEMATICS—Applications

General Terms

Theory

Keywords

Distance measure, genetic algorithms

1. INTRODUCTION

Metric is one of the fundamental tools for understanding space. It gives induced topology to the space and it is the most basic way to provide the space with topology [9]. Different metrics make different topologies. The shape of the space largely depends on its metric. In understanding genetic algorithms, metric is also basic and important. In genetic algorithms, a good distance measure not only helps to analyze their search spaces [13, 17, 19], but can also improve their search capability [5]. Hamming distance has been popular in most researches for genetic algorithms that deal with discrete spaces. It has also been widely adopted in studies about the analysis of the problem space [2, 8, 16].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'05, June 25–29, 2005, Washington, DC, USA.
Copyright 2005 ACM 1-59593-010-8/05/0006 ...\$5.00.

In this paper, we propose more reasonable distance measures depending on situations in the process of genetic algorithms and show that they are actually metrics. We propose three distance measures:

- one for the population-based search such as genetic algorithms
- another for the solution space based on K -ary encoding ($K \geq 2$)
- the third as an approximate measure of performance improvement of linkage-based genetic algorithms.

The remainder of this paper is organized as follows. In Section 2, we provide some preliminaries. In Section 3, Section 4, and Section 5, we propose three distance measures for genetic algorithms, provide their applications in genetic algorithms, and show that all the proposed distances become metrics. Finally, we make conclusions in Section 6.

2. PRELIMINARIES

2.1 Metric

A metric space is a set of points with an associate *metric* on the set [9]. Given a space X , the properties of its metric $\mathfrak{d} : X \times X \rightarrow \mathbb{R}$ are in the following.

1. $0 \leq \mathfrak{d}(x, y) < \infty$ for all x and y in X .
2. $\mathfrak{d}(x, y) = 0$ if and only if $x = y$.
3. $\mathfrak{d}(x, y) = \mathfrak{d}(y, x)$ for all x and y in X .
4. (Triangle Inequality)
 $\mathfrak{d}(x, z) \leq \mathfrak{d}(x, y) + \mathfrak{d}(y, z)$ for all x, y , and z in X .

If property 2 is violated (i.e., for some x and y such that $x \neq y$, $\mathfrak{d}(x, y) = 0$), then \mathfrak{d} is called a *pseudo-metric*. If property 4 does not hold, then \mathfrak{d} is called a *semi-metric*.

2.2 The Optimal Assignment Problem

Consider a weighted complete bipartite graph with bipartition (X, Y) , where $X = \{x_1, x_2, \dots, x_K\}$, $Y = \{y_1, y_2, \dots, y_K\}$, and each edge $(x_i, y_j) \in X \times Y$ has its weight w_{ij} . The *optimal assignment problem* is the problem of finding a maximum-weight (or minimum-weight) perfect matching in this weighted graph as follows:

$$\max_{\sigma \in \Sigma_K} \left(\sum_{i=1}^K w_{i\sigma(i)} \right) \quad \text{or} \quad \min_{\sigma \in \Sigma_K} \left(\sum_{i=1}^K w_{i\sigma(i)} \right)$$

where σ is a permutation. It is also known as the bipartite weighted matching problem.

To solve the optimal assignment problem, it is possible to enumerate all $K!$ permutations in Σ_K and find an optimal one among them. However, for a large K , such a procedure is intractable. Fortunately, Kuhn [14] proposed an efficient way to solve the problem. It is called the *Hungarian method*. Roughly speaking, starting from the initial unweighted bipartite graph with no edge, the method iteratively modifies edge weights, adds new edges into the bipartite graph, and applies the *maximum matching* (or *minimum covering*) in the resulting bipartite graph. It continues the above process until a perfect matching is found. The Hungarian method gives an optimum assignment and it can be implemented in $O(K^3)$ time [18].

2.3 Fitness Distance Correlation

The fitness distance correlation (FDC) is a measure of problem difficulty proposed by Jones and Forrest [8]. FDC is defined to be the correlation coefficient of the fitness and the distance to the nearest global optimum of sampled solutions. Thus, it ranges from -1 to 1 . The problem of maximizing the fitness is considered to become easier as the value comes close to -1 . On the other hand, that of minimizing the fitness is considered to become harder as the value approaches -1 .

When a genetic algorithm is hybridized with a local optimization, the population contains only local optima. It is thus valuable to examine FDC of the local-optimum space rather than that of the whole solution space.

3. DISTANCE MEASURE IN POPULATION SPACE

Since the genetic algorithm is a population-based search, the distance measure between populations is useful for understanding the behavior of genetic algorithms (e.g., [13]). For example, it can be used in the visualization of population convergence and in the analysis of the population space analogically to the solution space. Recently, Wineberg and Oppacher [19] pointed out the need for a population metric, proposed distance measures between populations, and showed that some measures are metric. They proposed two types of population distances. One is based on population diversity, which is defined to be the sum of the Hamming distances between all pairs of chromosomes. This measure is not intuitive. The other is defined to be the minimal number of mutations to transform one population to another. This is quite natural and intuitive. However, they considered just the gene frequency at each locus over the population, ignoring gene dependency, which is a serious drawback in obtaining reasonable distance formulas. Moreover, they considered the solution space having only the Hamming distance. In this section, we propose an intuitive and reasonable metric. We also assume that the solution space can have any metric.

3.1 New Measure

DEFINITION 1. Let K be the population size. Let population $\mathbf{p} = \{c_1, c_2, \dots, c_K\}$ and population $\mathbf{p}' = \{c'_1, c'_2, \dots, c'_K\}$. Given a metric \mathfrak{d} in solution space, we define the distance

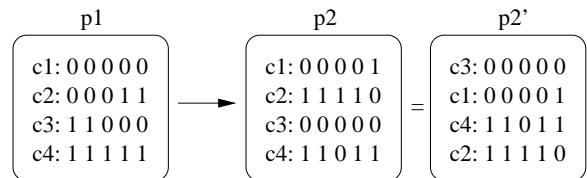


Figure 1: An example of two populations

D_K of the two populations as follows:

$$D_K(\mathbf{p}, \mathbf{p}') := \min_{\sigma \in \Sigma_K} \left(\sum_{i=1}^K \mathfrak{d}(c_i, c'_{\sigma(i)}) \right)$$

where σ denotes a permutation.

THEOREM 1. D_K is a metric in the population space.

PROOF: It is enough to show that D_K satisfies the triangle inequality.

$$\begin{aligned} & D_K(\mathbf{p}^x, \mathbf{p}^y) + D_K(\mathbf{p}^y, \mathbf{p}^z) \\ &=_{(WLOG)} \sum_{i=1}^K \mathfrak{d}(c_i^x, c_i^y) + \sum_{i=1}^K \mathfrak{d}(c_i^y, c_i^z) \\ &= \sum_{i=1}^K \mathfrak{d}(c_i^x, c_{\sigma(i)}^y) + \sum_{i=1}^K \mathfrak{d}(c_{\sigma(i)}^y, c_{\sigma'(i)}^z) \\ &= \sum_{i=1}^K (\mathfrak{d}(c_i^x, c_{\sigma(i)}^y) + \mathfrak{d}(c_{\sigma(i)}^y, c_{\sigma' \cdot \sigma(i)}^z)) \\ &\geq \sum_{i=1}^K \mathfrak{d}(c_i^x, c_{\sigma' \cdot \sigma(i)}^z) \quad (\because \mathfrak{d} \text{ is a metric}) \\ &\geq D_K(\mathbf{p}^x, \mathbf{p}^z) \quad (\because \sigma' \cdot \sigma \in \Sigma_K). \blacksquare \end{aligned}$$

Figure 1 shows an example of two populations $p1$ and $p2$ containing four binary chromosomes with five genes. Assume that the solution space has the Hamming distance. If we simply use the sum of the distances between pairs of chromosomes with the same indices, the distance between populations $p1$ and $p2$ is 8. The mutation-based distance of Wineberg and Oppacher [19] is 0 since two populations have the same gene frequency at each locus. But, since $p1$ is not equal to $p2$, it is not reasonable. We use the distance between $p1$ and $p2'$, which becomes 4.

3.2 Hungarian Method Can Compute D_K

The problem of computing D_K is formulated as the optimal assignment problem. Then, it can be computed by the Hungarian method. Figure 2 shows the assignment weight matrix $M = (m_{ij})$ between two populations \mathbf{p} and \mathbf{p}' . Each element m_{ij} represents $\mathfrak{d}(c_i, c'_j)$. The problem of computing D_K is exactly the problem of finding an assignment (permutation) with minimum summation.

4. DISTANCE MEASURE FOR K -GROUPING PROBLEM

We propose a distance measure for disjoint K -grouping problems such as K -way graph partitioning, graph K -colorability, bin packing, etc. These problems are well-known NP-complete problems [6]. A number of genetic algorithms have been

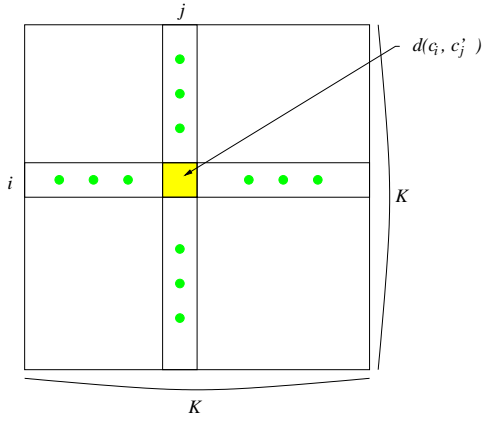


Figure 2: The assignment weight matrix between two populations

proposed for solving the problems. In these problems, since each group is not distinguishable, each solution has $K!$ representations. This makes the Hamming distance between two solutions unrealistic and undermines the effectiveness of crossover operators in genetic algorithms.

4.1 New Measure

DEFINITION 2. Let the universal set U be $\{1, 2, \dots, K\}^N$, where N is the problem size. Given two K -ary encodings $\mathbf{a}, \mathbf{b} \in U$ and a metric \mathfrak{d} in U , we define the distance measure d_K for K -grouping problem as follows:

$$d_K(\mathbf{a}, \mathbf{b}) := \min_{\sigma \in \Sigma_K} \mathfrak{d}(\mathbf{a}, \mathbf{b}_\sigma)$$

where σ is a permutation and \mathbf{b}_σ is a permuted encoding of \mathbf{b} by σ , i.e., the i^{th} element e_i of \mathbf{b} is transformed into $\sigma(e_i)$.

In the case of $K = 2$, the measure d_2 has been successfully used [3, 5, 12]. The definition generalizes it into the distance measure for K -grouping problem. We provide the following theorems.

THEOREM 2. d_K is a pseudo-metric in U .

PROOF: It is enough to show that d_K satisfies the triangle inequality.

$$\begin{aligned} d_K(\mathfrak{r}, \mathfrak{h}) + d_K(\mathfrak{h}, \mathfrak{z}) & \stackrel{(WLOG)}{=} \mathfrak{d}(\mathfrak{r}, \mathfrak{h}_\sigma) + \mathfrak{d}(\mathfrak{h}, \mathfrak{z}_{\sigma'}) \\ & = \mathfrak{d}(\mathfrak{r}, \mathfrak{h}_\sigma) + \mathfrak{d}(\mathfrak{h}_\sigma, (\mathfrak{z}_{\sigma'})_\sigma) \\ & = \mathfrak{d}(\mathfrak{r}, \mathfrak{h}_\sigma) + \mathfrak{d}(\mathfrak{h}_\sigma, \mathfrak{z}_{\sigma \cdot \sigma'}) \\ & \geq \mathfrak{d}(\mathfrak{r}, \mathfrak{z}_{\sigma \cdot \sigma'}) \quad (\because \mathfrak{d} \text{ is a metric}) \\ & \geq d_K(\mathfrak{r}, \mathfrak{z}) \quad (\because \sigma \cdot \sigma' \in \Sigma_K). \blacksquare \end{aligned}$$

Given an element $\mathbf{a} \in U$, since \mathfrak{d} is a metric, there are only $K!$ elements such that the distance d_K to \mathbf{a} is zero. If the distance d_K between two elements is equal to zero, we define them to be *in relation* \sim . Then, the following proposition holds.

PROPOSITION 1. The relation \sim is an equivalence relation.

PROOF: It is obvious that the relation \sim is reflexive and symmetric. It is transitive as in the following.

$$\begin{aligned} \mathbf{a} \sim \mathbf{b}, \mathbf{b} \sim \mathbf{c} & \Rightarrow_{(WLOG)} \mathbf{a} = \mathbf{b}_\sigma, \mathbf{b} = \mathbf{c}_{\sigma'} \\ & \Rightarrow \mathbf{a} = (\mathbf{c}_{\sigma'})_\sigma = \mathbf{c}_{\sigma \cdot \sigma'} \\ & \Rightarrow \mathbf{a} \sim \mathbf{c} \quad (\because \sigma \cdot \sigma' \in \Sigma_K). \blacksquare \end{aligned}$$

THEOREM 3. Suppose that Q is the quotient set of U by relation \sim (i.e., $Q = U / \sim$). Then, (Q, d_K) is a metric space, i.e., d_K is a metric in Q .

PROOF: By Proposition 1, Q is well defined. Since d_K is a pseudo-metric by Theorem 2, it is clear that d_K is a metric in Q . \blacksquare

4.2 Hungarian Method Can Compute d_K

When the metric \mathfrak{d} is the Hamming distance \mathfrak{H}^1 , the problem of computing d_K is also formulated as the optimal assignment problem. Hence, it can be computed by the Hungarian method. Figure 3 shows the assignment weight matrix $M = (m_{ij})$ between two chromosomes X and Y . Each element m_{ij} means $\sum_{k=1}^N I(X_k = i, Y_k \neq j)$ or $\sum_{k=1}^N I(X_k \neq i, Y_k = j)$, where N is the length of chromosome and $I(\cdot)$ is the indicator function, i.e., $I(true) = 1$ and $I(false) = 0$. The problem of computing d_K is exactly the problem of finding an assignment (permutation) with minimum summation.

THEOREM 4. If the metric \mathfrak{d} is the Hamming distance \mathfrak{H} , then

$$d_K(X, Y) = \min_{\sigma \in \Sigma_K} \left(\sum_{i=1}^K \sum_{k=1}^N I(X_k = i, Y_k \neq \sigma(i)) \right)$$

where N is the length of chromosome and $I(\cdot)$ is the indicator function, i.e., $I(true) = 1$ and $I(false) = 0$.

PROOF:

$$\begin{aligned} d_K(X, Y) & = \min_{\sigma \in \Sigma_K} \mathfrak{H}(X, Y_\sigma) \\ & = \min_{\sigma \in \Sigma_K} \left(\sum_{k=1}^N I(X_k \neq \sigma(Y_k)) \right) \\ & = \min_{\sigma \in \Sigma_K} \left(\sum_{k=1}^N \sum_{i=1}^K I(X_k = i, \sigma(Y_k) \neq i) \right) \\ & = \min_{\sigma \in \Sigma_K} \left(\sum_{k=1}^N \sum_{i=1}^K I(X_k = i, \sigma^{-1} \cdot \sigma(Y_k) \neq \sigma^{-1}(i)) \right) \\ & = \min_{\sigma \in \Sigma_K} \left(\sum_{k=1}^N \sum_{i=1}^K I(X_k = i, Y_k \neq \sigma^{-1}(i)) \right) \\ & = \min_{\sigma \in \Sigma_K} \left(\sum_{i=1}^K \sum_{k=1}^N I(X_k = i, Y_k \neq \sigma^{-1}(i)) \right) \\ & = \min_{\sigma \in \Sigma_K} \left(\sum_{i=1}^K \sum_{k=1}^N I(X_k = i, Y_k \neq \sigma(i)) \right) \\ & \quad (\because \sigma \in \Sigma_K \Leftrightarrow \sigma^{-1} \in \Sigma_K). \blacksquare \end{aligned}$$

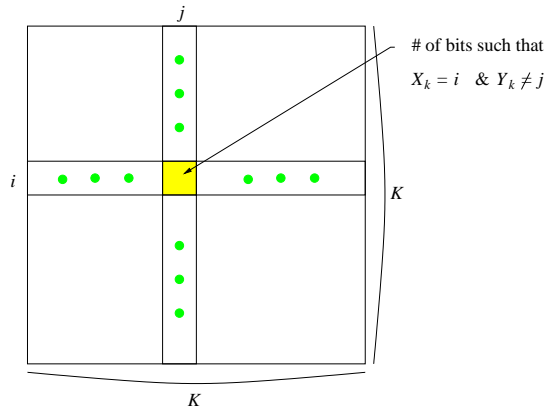


Figure 3: The assignment weight matrix between two K -ary chromosomes X and Y

Normalization is an approach that transforms the genotype of one parent to be consistent with that of the other parent. It is a method for alleviating difficulties caused by redundant encodings in genetic algorithms. It transforms a given parent to another genotype so that the genotype contexts of the parents are as similar as possible in crossover. There have been a few successful studies that used normalization [4, 10, 15]. Figure 4 shows the process of previous widely-used normalization in K -way graph partitioning. It was proposed in [15]. It has been widely used in a number of researches [5, 10]. However, it is just a greedy heuristic for finding a good assignment. On the other hand, our method using the Hungarian method always finds the optimum assignment efficiently. The heuristic of Figure 4 and our method have the same time complexity $O(N + K^3)$. Table 1 shows the average distance values of Hamming distance, the greedy heuristic, and our method over various K 's. Those of Hamming distance are theoretical expected values and the others are obtained from randomly generated 1,000 pairs of K -ary solutions with $N = 1000$. Both methods reduce genotype inconsistency in a large degree. We can see that the previous heuristic performs worse as K increases, whereas our method always returns the optima. Since the normalizations of chromosomes in [5] and [15] pursue the minimization of genotype inconsistency among chromosomes, the proposed metric is ideal in this line of work. Also, though our method is slower than the greedy heuristic, our optimal method for the proposed metric is practical even for large K 's.

5. LEVEL DISTANCE MEASURE

5.1 The Second Level Distance

The first level distance measure is commonly the Hamming distance. Other distance measures can also be used as the first level distance (e.g., normalized Hamming distance in Section 4). In this section, we define the second level distance measure. It is defined from the first level distance. Given the problem instance p , consider the graph G_p representing the first order gene interaction (e.g., [11]), i.e.,

¹It is quite easy to verify that the Hamming distance is a metric.

```

Normalization(parent1, parent2)
{
  // count[i, j] :=  $\sum_{k=1}^N I(\text{parent1}[k] = i, \text{parent2}[k] = j)$ 
  count[1..K, 1..K]  $\leftarrow$  0;
  for i  $\leftarrow$  1 to N
    count[parent1[i], parent2[i]]++;
  for i  $\leftarrow$  1 to K
  {
    Choose p and q that maximizes count[p, q];
    for j  $\leftarrow$  1 to K
      count[p, j]  $\leftarrow$   $-\infty$ , count[j, q]  $\leftarrow$   $-\infty$ ;
     $\sigma(q) \leftarrow p$ ;
  }
  for i  $\leftarrow$  1 to N
    parent2[i]  $\leftarrow$   $\sigma(\text{parent2}[i])$ ;
}

```

Figure 4: Previous normalization heuristic between two parents

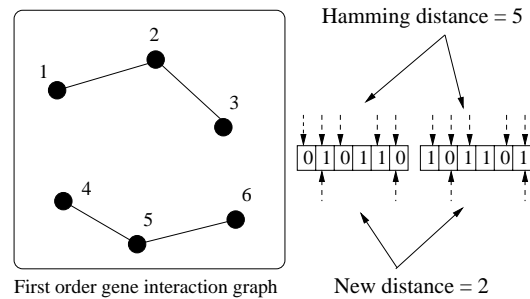


Figure 5: An example

representing only gene interactions between a pair of genes. Let A_p be the adjacency matrix of G_p .

DEFINITION 3. Suppose that there exists the inverse of $A_p \oplus I$. We define the second level distance measure D_p as follows:

$$D_p(\mathbf{a}, \mathbf{b}) := \|(A_p \oplus I)^{-1}(\mathbf{a} \oplus \mathbf{b})\|$$

where \oplus is XOR operator and $\|\cdot\|$ is a norm derived from the first level distance \mathfrak{d} (i.e., $\|\cdot\| = \mathfrak{d}(\cdot, 0)$).

THEOREM 5. D_p is a metric.

PROOF: It is enough to show that D_p satisfies the triangle inequality.

$$\begin{aligned}
 & D_p(x, y) + D_p(y, z) \\
 &= \|(A_p \oplus I)^{-1}(x \oplus y)\| + \|(A_p \oplus I)^{-1}(y \oplus z)\| \\
 &\geq \|(A_p \oplus I)^{-1}(x \oplus y) \oplus (A_p \oplus I)^{-1}(y \oplus z)\| \\
 &= \|(A_p \oplus I)^{-1}((x \oplus y) \oplus (y \oplus z))\| \\
 &= \|(A_p \oplus I)^{-1}((x \oplus z) \oplus (y \oplus y))\| \\
 &= \|(A_p \oplus I)^{-1}(x \oplus z)\| \\
 &= D_p(x, z). \blacksquare
 \end{aligned}$$

In the case that there does not exist the inverse of $A_p \oplus I$, we can extend the distance into a well-defined form as follows:

$$D_p(\mathbf{a}, \mathbf{b}) := \min_{\mathbf{r}} \|(\arg \min_{\mathbf{r}} \|\mathbf{a} \oplus \mathbf{b} \oplus (A_p \oplus I)\mathbf{r}\|)\|.$$

Table 1: Comparison of Distance Values

Method	Hamming distance	Greedy heuristic [15]		Our method	
	Ave*	Ave	CPU‡	Ave†	CPU‡
$K = 4$	750.00	727.22	0.0002	724.46	0.0018
$K = 8$	875.00	838.27	0.0003	835.35	0.0020
$K = 16$	937.50	888.03	0.0008	884.44	0.0029
$K = 32$	968.75	901.03	0.0023	896.76	0.0059
$K = 64$	984.38	890.65	0.0079	884.51	0.0191
$K = 128$	992.19	858.82	0.0308	847.51	0.0922

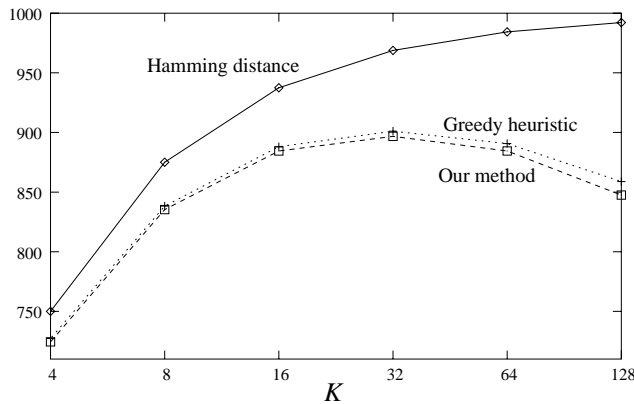
Average over 1,000 pairs of randomly generated solutions.

* The theoretical value.

† The optimum value.

‡ CPU seconds on Pentium PC 2.40GHz.

Distance value



The second level distance and its extension are efficiently computed in $O(N^3)$ by a variant of Gauss-Jordan elimination method [1].

5.2 An Example Application

Intuitively, the proposed distance can be understood as the minimum number of mutated bits for transforming one chromosome into the other in the genetic process using optimal gene rearrangement. Figure 5 shows an example about the distances between two chromosomes with optimal gene ordering in a gene interaction graph. Consider the 6-node MAX CUT problem² instance that maximizes the following formula:

$$x_1 \oplus x_2 + x_2 \oplus x_3 - x_4 \oplus x_5 - x_5 \oplus x_6,$$

where $x_i \in \{0, 1\}$ means the partition to which the vertex i belongs. In the problem instance, edges $\{1, 2\}$ and $\{2, 3\}$ positively affect the fitness and edges $\{4, 5\}$ and $\{5, 6\}$ negatively affect the fitness. It has the graph of Figure 5 as its gene interaction graph. If we use the normalized Hamming distance (for 2-grouping problem) as a distance measure, the FDC of this problem is -0.50 . When its second level distance is used instead of the normalized Hamming distance, the FDC becomes -0.95 .

Given a graph $G = (V, E)$ and its adjacency matrix $A = (a_{ij})$, the graph partitioning problem is the problem that

²Given an undirected graph $G = (V, E)$ with edge weights $(w_{ij})_{(i,j) \in E}$, the MAX CUT problem is the problem of finding a subset $S \subset V$ which maximizes the sum of edge weights in the cut $(S, V \setminus S)$.

minimizes the following formula:

$$\frac{1}{2} \sum_{i=1}^{|V|} \sum_{j=1}^{|V|} a_{ij} (x_i \oplus x_j) + \gamma \left(\sum_{i=1}^{|V|} x_i - \frac{|V|}{2} \right)^2$$

where γ is a positive constant. If we ignore the last constraint term about balancing, we can regard the given graph as the gene interaction graph of the given problem instance. Bui and Moon [3] tried gene rearrangement from given graph in genetic algorithms for graph partitioning and showed dramatic performance improvement on some graphs. The FDC using the proposed second level distance identified the graphs that benefited most by gene rearrangement in genetic algorithms. Figure 6 shows the relationship between FDC and performance improvement for 16 graphs (8 random graphs and 8 geometric graphs) used in [7]. Here, the performance improvement means the difference in percentage between the average performances of genetic algorithms with gene rearrangement and without gene rearrangement (data from [3]). FDC values were approximated from randomly generated 10,000 local optima. When the first level distance was used, there was little correlation. But, FDC using the second level distance well accorded with the performance improvement (see Figure 6(b) and Table 2).

5.3 Higher Level Distance

In this subsection, we extend the second level distance.

DEFINITION 4. We define the i^{th} level gene adjacency matrix as follows:

$$A_p^{(i)} := I \oplus (A_p \cdot A_p^{(i-1)})$$

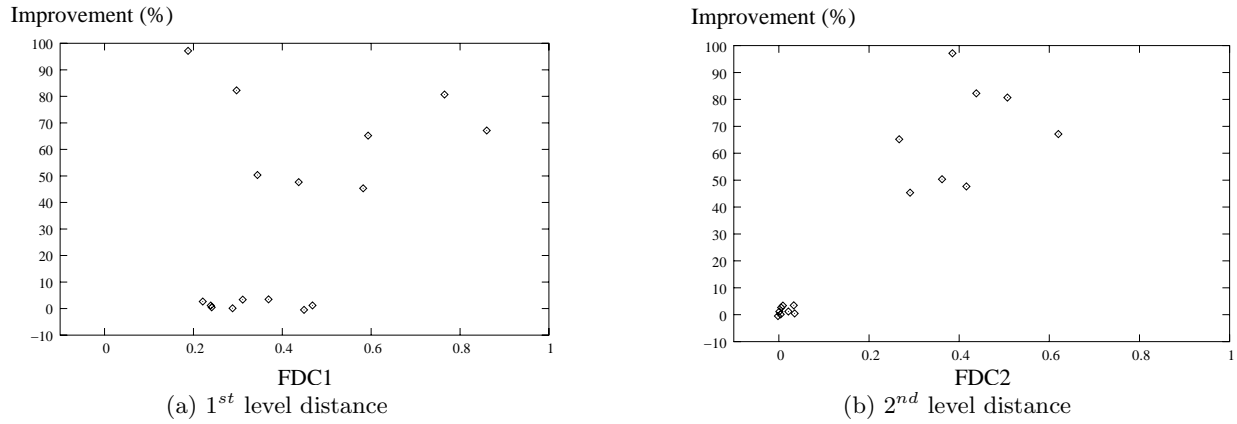


Figure 6: Relationship between FDC and performance improvement

Table 2: Comparison of FDCs with Different Level Distances

Graph	w/ $D_p^{(1)}$	w/ $D_p^{(2)}$	w/ $D_p^{(3)}$
G500.05	0.449	-0.002	0.020
G500.20	0.288	0.004	-0.005
G1000.05	0.239	0.001	0.003
G1000.20	0.468	0.021	0.017
U500.05 [†]	0.297	<u>0.438</u>	0.252
U500.20	0.593	<u>0.267</u>	0.198
U1000.05 [†]	0.188	<u>0.384</u>	0.177
U1000.20	0.582	0.291	0.165

[†] The graphs that benefited most from gene rearrangement in [3].

where $i \geq 1$ and $A_p^{(0)} = O$.

DEFINITION 5. Suppose that there exists the inverse of the i^{th} level gene adjacency matrix $A_p^{(i)}$. We define the i^{th} level distance as follows:

$$D_p^{(i)}(\mathbf{a}, \mathbf{b}) := \|(A_p^{(i)})^{-1}(\mathbf{a} \oplus \mathbf{b})\|.$$

In the same way as $i = 2$, $D_p^{(i)}$ is a metric for every i . Without the inverse of $A_p \oplus I$, we extend the distance as follows:

$$D_p^{(i)}(\mathbf{a}, \mathbf{b}) := \min_{\mathbf{r}} \|\arg \min_{\mathbf{r}} \|\mathbf{a} \oplus \mathbf{b} \oplus A_p^{(i)} \mathbf{r}\|\|.$$

The last column of Table 2 shows FDC values using the third level distance in graph partitioning. We need to study more on if we can use the third or higher level distances as supportive measures of the second level distance.

6. CONCLUSIONS

Most previous studies needing distances among chromosomes in genetic algorithms used the Hamming distance. The purpose of this paper is to develop more meaningful distance measures for genetic algorithms. We proposed three distance measures for genetic algorithms and showed that the proposed distance measures are metrics. We also showed that there is an efficient method to compute the metrics.

We also proposed higher level distances which see the problem spaces from a different viewpoint. Among them, the second level distance turned out to be helpful in expecting the graphs that can benefit from gene rearrangement in graph partitioning. More examination on the usefulness of higher level distances is left for future study. We hope that the proposed metrics are useful for improving GA's search capability and understanding GA's working mechanism.

7. ACKNOWLEDGMENTS

This work was supported by the Brain Korea 21 Project in 2005. This was also partly supported by a grant of the International Mobile Telecommunications 2000 R&D Project, Ministry of Information & Communication, Republic of Korea. The ICT at Seoul National University provided research facilities for this study.

8. REFERENCES

- [1] M. Anderson and T. Feil. Turning lights out with linear algebra. *Mathematics Magazine*, 71(4):300–303, 1998.
- [2] 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.
- [3] T. N. Bui and B. R. Moon. Genetic algorithm and graph partitioning. *IEEE Transactions on Computers*, 45(7):841–855, 1996.
- [4] S. S. Choi and B. R. Moon. Isomorphism, normalization, and a genetic algorithm for sorting network optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 327–334, 2002.
- [5] S. S. Choi and B. R. Moon. Normalization in genetic algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 862–873, 2003.
- [6] M. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, 1979.
- [7] D. S. Johnson, C. Aragon, L. McGeoch, and C. Schevon. Optimization by simulated annealing: An experimental evaluation, Part 1, graph partitioning. *Operations Research*, 37:865–892, 1989.

- [8] T. Jones and S. Forrest. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 184–192, 1995.
- [9] D. W. Kahn. *Topology, An introduction to the point-set and algebraic areas*. Dover Publications, Inc., New York, 1995.
- [10] S. J. Kang and B. R. Moon. A hybrid genetic algorithm for multiway graph partitioning. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 159–166, 2000.
- [11] Y. H. Kim, Y. K. Kwon, and B. R. Moon. Problem-independent schema synthesis for genetic algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1112–1122, 2003.
- [12] Y. H. Kim and B. R. Moon. Investigation of the fitness landscapes and multi-parent crossover for graph bipartitioning. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1123–1135, 2003.
- [13] Y. H. Kim and B. R. Moon. New usage of Sammon’s mapping for genetic visualization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1136–1147, 2003.
- [14] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Res. Logist. Quart.*, 2:83–97, 1955.
- [15] G. Laszewski. Intelligent structural operators for the k -way graph partitioning problem. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 45–52, July 1991.
- [16] P. Merz and B. Freisleben. Fitness landscape analysis and memetic algorithms for the quadratic assignment problem. *IEEE Transactions on Evolutionary Computation*, 4(4):337–352, 2000.
- [17] A. Moraglio and R. Poli. Topological interpretation of crossover. In *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 1, pages 1377–1388, 2004.
- [18] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, Englewood Cliffs, N.J., 1982.
- [19] M. Wineberg and F. Oppacher. Distance between populations. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1481–1492, 2003.