Computing the Epistasis Variance of Large-Scale Traveling Salesman Problems

Dong-Il Seo
School of Computer Science & Engineering,
Seoul National University,
Sillim-dong, Gwanak-gu, Seoul, 151-744 Korea
diseo@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

The interaction among variables of an optimization problem is known as epistasis, and its degree is an important measure for the nonlinearity of the problem. We address the problem of enormous time complexity for computing Davidor's epistasis variance of the traveling salesman problem (TSP). To reduce the complexity, we introduce the concept of schema-linear problem (SLP), show that TSP is a SLP, and present a relevant lemma, called Summation Rule. Using the Summation Rule, we provide a closed formula for epistasis that reduces the time complexity from $O(n^n)$ to $O(n^2)$. Additionally, we propose a new more scalable measure of epistasis by a careful derivation from the original.

Categories and Subject Descriptors

G.m [**Mathematics of Computing**]: Miscellaneous

General Terms

Theory

Keywords

Epistasis, linkage, traveling salesman problem, TSP

1. INTRODUCTION

Optimization is one of the most important targets of genetic and evolutionary algorithms. Mostly, an optimization problem [1, 15] is defined as a function, sometimes called fitness function, from a set called universe to the set of real numbers $\mathbb{R}$. The universe is a set of feasible solutions, which is often represented by a number of variables, each of which has its own domain. If the domains are discrete, the problem is said to be combinatorial.

In most practical optimization problems, the contribution of each variable to the fitness depends on the states of other variables. If not, we can independently determine optimal values of the variables one by one. These interactions among variables immanent in problems is called *linkage* [13, 9] or *epistasis* [10, 4, 20].

The development history of genetic and evolutionary algorithms include the efforts to exploit epistatic properties in problem solving. The estimation of distribution algorithms (EDAs) [11] and the topological linkage-based genetic algorithms (TLBGAs) [22] are representative products of such efforts. In the approaches, the epistasis among variables are implicitly estimated and applied to the evolution of solutions. Recently, a number of approaches that attempt to detect independently optimizable linkage groups of a pseudo-Boolean function[1] have been proposed [13, 9, 23].

The Walsh transform, first introduced by Bethke [2] and popularized by Goldberg [7, 8], is one of the classical methods for analyzing epistatic properties of a pseudo-Boolean function. This method was later extended by Mason [12] to a problem using non-binary encoding. Given a pseudo-Boolean function of n variables, the Walsh transform generates 2^n coefficients, called *partition coefficients*, from 2^n fitness values. Each coefficient reflects the epistasis of the corresponding schema, which means specific pattern of the variable assignments. Rana et al. [16] showed that these coefficients for MAXSAT can be directly computed in linear time with respect to the number of clauses, and Heckendorn and Wright [9] proposed a general algorithm that detects the linkage groups and computes the coefficients concurrently.

Reeves and Wright [17, 18] considered the epistasis from the viewpoint of statistical experimental design. Here the variables' contribution to the fitness is decomposed into a number of "effects" based on a linear model. The effects are classified into *linear effects* and *interaction effects*, and the latter represent epistases of the corresponding schemata. These effects proved to be equivalent to the partition coefficients of the Walsh transform [17].

Davidor's epistasis variance [4] is the first general measure of epistasis. It measures the portion of the fitness variation due to the interaction effects. The entropic epistasis proposed by Seo et al. [21, 20] is a measure that quantifies the general dependence among variables based on the information theory, whereas the epistasis variance quantifies the nonlinearity lying in the fitness landscape.

Naudts [14] proposed a closed formula to compute the epistasis variance of Royal Road functions [5]. However, the exact computation for general combinatorial optimiza-

[1] A fitness function is called *pseudo-Boolean* if it is defined on $\{0,1\}^n$ for some $n \geq 1$.

GECCO'05, June 25–29, 2005, Washington, DC, USA.

tion problems becomes computationally prohibitive as the problem size increases. Thus the study using the measure has been mostly of theoretical interest. Moreover, since it assumes complete knowledge of the universe, the computation is unrealistic in the sense that the ultimate purpose of optimization is to find the most desirable solution in the universe.

In this context, we can consider only a portion of solutions in the computation instead of the whole solutions in the universe. Unfortunately, the results of experiments by Davidor [4] showed that sampling bias has a considerable effect on the measurement of the epistasis variance.

In this paper, we propose an efficient computation method for the epistasis variance of a symmetric traveling salesman problem (TSP). To do so, we first extend Holland's schema definition to first-order logic expressions. Based on the extension, we define *schema-linear problem (SLP)* and provide a lemma called Summation Rule applicable to SLPs. Further, we show that TSP is a SLP and derive an equation to compute the epistasis variance of symmetric TSP in polynomial time using the Summation Rule. Based on the equation, we propose a new measure which is more scalable than the original.

The rest of this paper is organized as follows: We extend the schema definition, define SLP, and derive the Summation Rule in Section 2. Then we review the epistasis variance, show that TSP is a SLP, and derive an equation to efficiently compute the epistasis variance of symmetric TSPs in Section 3. The results of simple experiments to confirm the validity of the equation are provided in Section 4. We finally provide concluding remarks in Section 5.

2. SCHEMA-LINEAR PROBLEM

2.1 Combinatorial Optimization Problem

An instance of an optimization problem is specified by a pair $(\mathcal{U}, f)$, where the universe $\mathcal{U}$ is the set of feasible solutions and the fitness function f is a mapping $f : \mathcal{U} \to \mathbb{R}$. A solution set is often represented by a set of variables, each of which has its own domain. If the domains are discrete, they are called alphabets and the problem is said to be combinatorial.

Let the variable indices of a given problem be $\mathcal{V} = \{1, \ldots, n\}$, and the alphabet for each variable be $\mathcal{A}_i, i \in \mathcal{V}$. Let the universe and the fitness function be $\mathcal{U} \subseteq \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ and $f : \mathcal{U} \to \mathcal{F}$, respectively. We assume that the alphabet of each variable is finite. Then, the set of all fitness values $\mathcal{F} \subset \mathbb{R}$ is finite as the universe is finite. We also assume that $\mathcal{U} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$.

2.2 Extended Schema Definition

In Holland's definition [10], schema means a set of solutions that have a specific pattern of variable assignments. For example, "$*10**$" is a schema corresponding to the solutions whose second and third variables have 1 and 0, respectively. For convenience, we use a special notation for schema as follows:

$$H_{(i_1;a_1,\ldots,i_k;a_k)}(x) \equiv (x_{i_1} = a_1) \wedge \cdots \wedge (x_{i_k} = a_k). \quad (1)$$

The schema denoted by $H_{(2;1,3;0)}$ is equivalent to $*10**$. In this paper, we extend this schema definition to a first-order logic expression (see [19]) of variables as follows:

Definition 1 (Schema). *Given a problem of n variables $x = (x_1, x_2, \ldots, x_n), x_i \in \mathcal{A}_i$, a* schema *$H(x)$ is a first-order logic expression on x.*

"$H_1(x_2, x_3) \equiv (x_2 = 1) \wedge (x_3 = 0)$" and "$H_2(x_2, x_3) \equiv x_2 \neq x_3$" are example schemata. H_1 is equivalent to $H_{(2;1,3;0)}$, but H_2 is a schema that cannot be denoted by Holland's notation.

The instance set of a schema is the set of solutions that satisfy the schema.

Definition 2 (Instance Set). *Given a problem of universe $\mathcal{U}$ and a schema H, the* instance set *$\mathcal{U}_H$ of H is defined to be the set of solutions that satisfy H, i.e., $\mathcal{U}_H = \{x : H(x) = \mathbf{true}, x \in \mathcal{U}\}$.*

From the above definitions, we obtain the following corollary.

Corollary 1. *Given a universe $\mathcal{U}$ and schemata H_1 and H_2,*

$$\begin{aligned} \mathcal{U}_{H_1 \wedge H_2} &= \mathcal{U}_{H_1} \cap \mathcal{U}_{H_2}, \\ \mathcal{U}_{H_1 \vee H_2} &= \mathcal{U}_{H_1} \cup \mathcal{U}_{H_2}, \\ \mathcal{U}_{\neg H_1} &= \mathcal{U}^c_{H_1} = \mathcal{U} \setminus \mathcal{U}_{H_1}. \end{aligned} \quad (2)$$

2.3 Schema-Linear Problem and Summation Rule

We define the fitness of a schema to be the arithmetic average of the fitness values of the corresponding solutions. That is, the fitness of a schema is defined as follows:

Definition 3 (Fitness of Schema). *Given a fitness function f and a schema H, the k^{th} fitness $f^{(k)}(H)$ of H is the average of the k^{th} power of the fitness of each solution in the instance set $\mathcal{U}_H$, i.e.,*

$$f^{(k)}(H) = \frac{1}{|\mathcal{U}_H|} \sum_{x \in \mathcal{U}_H} f(x)^k \quad (3)$$

For convenience, we denote $f^{(1)}(H)$ by $f(H)$.

A problem is said to be a *schema-linear problem (SLP)* if the fitness of a solution can be expressed as a summation formula of coefficients corresponding to the schemata that the solution belongs to. That is, schema-linear problem is defined as follows:

Definition 4 (Schema-Linear Problem). *A problem is said to be a schema-linear problem if there exist m schemata H_i and m real numbers w_i for a positive integer m such that the fitness function f can be expressed as*

$$f(x) = \sum_{i=1}^{m} 1(H_i(x)) w_i \quad (4)$$

where $1(\cdot)$ is an indicator function, i.e., $1(\mathbf{true}) = 1$ and $1(\mathbf{false}) = 0$. H_i and w_i are said to be a component schema and a component weight of the problem, respectively.

From the above definition, we obtain the following lemma.

Lemma 1 (Summation Rule). *Given a schema-linear problem of fitness function f and m components (H_i, w_i), the k^{th} fitness $f^{(k)}(H)$ of a schema H satisfies the following equation.*

$$f^{(k)}(H) = \frac{1}{|\mathcal{U}_H|} \sum_{i_1=1}^{m} \cdots \sum_{i_k=1}^{m} w_{i_1} \cdots w_{i_k} |\mathcal{U}_{H_{i_1} \wedge \cdots \wedge H_{i_k} \wedge H}| \quad (5)$$

Proof. From the definition of $f^{(k)}(H)$, we can derive the followings.

$$
\begin{aligned}
& f^{(k)}(H) \\
= \; & \frac{1}{|\mathcal{U}_H|} \sum_{x \in \mathcal{U}_H} f(x)^k \\
= \; & \frac{1}{|\mathcal{U}_H|} \sum_{x \in \mathcal{U}_H} \left(\sum_{i_1=1}^{m} 1(H_{i_1}(x)) w_{i_1} \right)^k \\
= \; & \frac{1}{|\mathcal{U}_H|} \sum_{x \in \mathcal{U}_H} \sum_{i_1=1}^{m} \cdots \sum_{i_k=1}^{m} 1(H_{i_1}(x)) \cdots 1(H_{i_k}(x)) w_{i_1} \cdots w_{i_k} \\
= \; & \frac{1}{|\mathcal{U}_H|} \sum_{x \in \mathcal{U}_H} \sum_{i_1=1}^{m} \cdots \sum_{i_k=1}^{m} 1(H_{i_1}(x) \wedge \cdots \wedge H_{i_k}(x)) w_{i_1} \cdots w_{i_k} \\
= \; & \frac{1}{|\mathcal{U}_H|} \sum_{i_1=1}^{m} \cdots \sum_{i_k=1}^{m} w_{i_1} \cdots w_{i_k} \sum_{x \in \mathcal{U}_H} 1(H_{i_1}(x) \wedge \cdots \wedge H_{i_k}(x)) \\
= \; & \frac{1}{|\mathcal{U}_H|} \sum_{i_1=1}^{m} \cdots \sum_{i_k=1}^{m} w_{i_1} \cdots w_{i_k} |\mathcal{U}_{H_{i_1} \wedge \cdots \wedge H_{i_k} \wedge H}|
\end{aligned}
$$

□

This lemma will be used in the derivation of the main equation in Section 3.3.

3. EPISTASIS VARIANCE OF TSP

3.1 Experimental Design and Epistasis Variance

The experimental design is a branch of statistics that attempts to conduct the way in which experiments should be carried out so the data gathered will have statistical value. Reeves and Wright [17, 18] are the first who explained the epistatic behavior of a problem in the light of the experimental design. Generally, the design of experiments is based on an underlying linear model. Accordingly, the fitness $f(x)$ of a solution $x = (x_1, \ldots, x_n) \in \mathcal{U}$ is expressed as

$$
\begin{aligned}
f(x) \;=\; & \text{constant} + \sum_{i=1}^{n} (\text{effect of } x_i) \\
& + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (\text{joint effect of } x_i \text{ and } x_j) \\
& + \cdots \\
& + (\text{joint effect of } x_1,\ x_2, \ldots, \text{ and } x_n).
\end{aligned} \tag{6}
$$

For instance, the model for a problem of three variables can be written as

$$
\begin{aligned}
f(a_1, a_2, a_3) \;=\; & \bar{f} + C_1(a_1) + C_2(a_2) + C_3(a_3) \\
& + C_{1,2}(a_1, a_2) + C_{1,3}(a_1, a_3) \\
& + C_{2,3}(a_2, a_3) + C_{1,2,3}(a_1, a_2, a_3)
\end{aligned} \tag{7}
$$

where $\bar{f}$ is the average fitness, $C_1(a_1)$ is the effect of $x_1 = a_1$, $C_2(a_2)$ is the effect of $x_2 = a_2$, $C_{1,2}(a_1, a_2)$ is the effect of $x_1 = a_1$ and $x_2 = a_2$, and so on. In Equation (6), the terms "constant" and "$\sum_{i=1}^{n}$(effect of x_i)" are referred to as *linear effects* and the other terms as *interaction effects*. It is easy to show that the total sum of squares (SS), that measures the total variation of the fitness, is the sum of the linear effects SS and interaction effects SS, i.e.,

$$\text{Total SS} = \text{Linear effects SS} + \text{Interaction effects SS}. \tag{8}$$

Davidor's epistasis variance corresponds to the interaction effects SS. Reeves and Wright derived a normalized form of the measure by dividing it by the total SS (see [17, 18] for details). That is, they proposed an epistasis measure η defined as

$$
\eta = \frac{\sum_{x \in \mathcal{U}} (\text{interaction effect of } x)^2}{\sum_{x \in \mathcal{U}} \left(f(x) - \bar{f} \right)^2} \tag{9}
$$

where $\mathcal{U}$ is the universe and $\bar{f}$ is the average fitness of all $x \in \mathcal{U}$, i.e., $\bar{f} = f(\mathbf{true})$ by Definition 3.

Using (1) and Definition 3, we can obtain the following four equations for the parameters in (7):

$$
\begin{aligned}
\bar{f} &= f(\mathbf{true}), \\
\bar{f} + C_1(a_1) &= f(H_{(1;a_1)}), \\
\bar{f} + C_2(a_2) &= f(H_{(2;a_2)}), \\
\bar{f} + C_3(a_3) &= f(H_{(3;a_3)}).
\end{aligned} \tag{10}
$$

From the equations, the linear effects can be expressed as

$$
\bar{f} + C_1(a_1) + C_2(a_2) + C_3(a_3) = \sum_{i=1}^{3} f(H_{(i;a_i)}) - 2f(\mathbf{true}). \tag{11}
$$

Now, the interaction effects are

$$
\begin{aligned}
& f(a_1, a_2, a_3) - (\bar{f} + C_1(a_1) + C_2(a_2) + C_3(a_3)) \\
& = f(a_1, a_2, a_3) - \sum_{i=1}^{3} f(H_{(i;a_i)}) + 2f(\mathbf{true}).
\end{aligned} \tag{12}
$$

By generalizing this to the problems of size n, we can obtain the following formula for η:

$$
\eta = \frac{\sum_{x \in \mathcal{U}} \left(f(x) - \sum_{i=1}^{n} f(H_{(i;x_i)}) + (n-1)\ f(\mathbf{true}) \right)^2}{\sum_{x \in \mathcal{U}} \left(f(x) - f(\mathbf{true}) \right)^2}. \tag{13}
$$

A naive computation for this formula takes $O(2^n)$ time for a binary-encoded problem since there exist totally 2^n distinct solutions. For the same reason, a naive computation takes $O(n^n)$ time for TSP in the locus-based encoding, which will be explained in the next section.

3.2 TSP Encoding

Given n cities, the *traveling salesman problem* (TSP) is the problem of finding the shortest Hamiltonian cycle visiting the cities. It is an NP-hard problem [6], and known to be one of the most popular and important combinatorial optimization problems.

A problem instance of TSP is specified by a distance matrix (d_{ij}) where d_{ij} corresponds to the distance from city i to city j. For all i, $d_{ii} = 0$. In this paper, we consider only symmetric instances, i.e., $d_{ij} = d_{ji}$. For simplicity, we use the following conventional notations: For a distance matrix (d_{ij}), we denote the i^{th} row sum of elements by $d_{i\cdot}$, the j^{th} column sum of elements by $d_{\cdot j}$, and the whole sum of elements in the matrix by $d_{\cdot\cdot}$. That is, $d_{i\cdot} = \sum_{j=1}^{n} d_{ij}$, $d_{\cdot j} = \sum_{i=1}^{n} d_{ij}$, and $d_{\cdot\cdot} = \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij}$.

To consider the epistatic properties of TSP, we use a locus-based encoding as in [3] where one variable is allocated for each city and the value of a variable represents the index of its next city in the corresponding tour. Thus, for an instance

of size n, the alphabet $\mathcal{A}_i$ of i^{th} variable is defined as follows:

$$\mathcal{A}_i = \mathcal{V} \setminus \{i\}, i \in \mathcal{V} \tag{14}$$

where $\mathcal{V} = \{1, \ldots, n\}$. Here the universe $\mathcal{U}$ is $\mathcal{U} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ and thus the size of the universe is $|\mathcal{U}| = (n-1)^n$.

Let $\mathcal{D}$ be a set of weighted digraphs $D = (\mathcal{V}, A)$ where the out-degree of each vertex is 1, no loop is contained in A, and the weight of an arc $(i \in \mathcal{V}, j \in \mathcal{A}_i) \in A$ is d_{ij}. We can see that each solution in $\mathcal{U}$ represents a digraph in $\mathcal{D}$. It is not difficult to show that the encoding is indeed a bijective mapping from $\mathcal{U}$ to $\mathcal{D}$. Each digraph in $\mathcal{D}$ is either a directed Hamiltonian cycle or not.

The fitness of a solution is defined as follows: If the corresponding digraph of the solution is a Hamiltonian cycle, the fitness of the solution is defined to be the sum of the arc weights of the digraph. Otherwise, it is defined to be the value obtained by adding additional penalty to the sum. For more formal definition, we define a schema H_C to decide whether the corresponding digraph is a Hamiltonian cycle or not as follows:

$$\begin{aligned} H_C(x) \quad \equiv \quad & \exists\, y_2, \ldots, y_n \in \mathcal{V}\ ((y_2 = x_1) \\ & \wedge\ (y_3 = x_{y_2}) \wedge \cdots \wedge (y_n = x_{y_{n-1}}) \\ & \wedge\ (\forall\, i, j \in \mathcal{V}\ ((i \neq j) \rightarrow (y_i \neq y_j)))). \end{aligned} \tag{15}$$

Now, given a distance matrix (d_{ij}), the fitness $f(x)$ of a solution $x = (x_1, \ldots, x_n) \in \mathcal{U}$ is defined as follows:

$$f(x) = \begin{cases} \displaystyle\sum_{i=1}^{n} d_{ix_i} & \text{if } H_C(x) = \textbf{true} \\ \displaystyle\sum_{i=1}^{n} d_{ix_i} + L & \text{otherwise} \end{cases} \tag{16}$$

where the penalty L is $L = \sum_{i=1}^{n} \max_{j \in \mathcal{A}_i}\{d_{ij}\}$. Other kinds of sufficiently large values for L are also feasible, but we will be able to see that this definition is helpful to simplify the equations in the next section.

For using Lemma 1, we need knowledge about the sizes of relevant instance sets. We summarize a number of useful equations of instance set sizes in the following.

THEOREM 1. *The following equations hold for the instance set sizes of the schemata* (1) *and* (15)*. For pairwise distinct values* $a_i \in \mathcal{A}_i, i = 1, \ldots, k$*,*

$$|\mathcal{U}_{H_C}| = (n-1)!, \tag{17}$$

$$|\mathcal{U}_{H_{(i_1;a_1,\ldots,i_k;a_k)}}| = (n-1)^{n-k}, \tag{18}$$

$$|\mathcal{U}_{H_{(i_1;a_1,\ldots,i_k;a_k)} \wedge H_C}| = (n-k-1)!. \tag{19}$$

Proof. Clearly, since there exist total $(n-1)!$ distinct directed Hamiltonian cycles in $\mathcal{D}$, (17) holds. For $x \in \mathcal{U}$, $H_{(i_1;a_1,\ldots,i_k;a_k)}(x)$ is **true** if $x_{i_1} = a_1$, …, and $x_{i_k} = a_k$. Since there exist total $(n-1)^{n-k}$ such solutions, (18) holds. Similarly, for $x \in \mathcal{U}$, $(H_{(i_1;a_1,\ldots,i_k;a_k)} \wedge H_C)(x)$ is **true** if $x_{i_1} = a_1$, …, $x_{i_k} = a_k$, and the corresponding digraph is a directed Hamiltonian cycle. Since a_i are pairwise distinct, the digraph with only the arcs (x_{i_1}, a_1), …, (x_{i_k}, a_k) contains $n-k$ connected components, and this is the same as the configuration of only $n-k$ vertices. Hence, by (17), there exist $(n-k-1)!$ distinct directed Hamiltonian cycles, which establishes (19). □

3.3 Epistasis Variance Computation

In this section, we explain a computation method of the epistasis variance of TSP in the locus-based encoding. Using the extended schema definition and the bracket notation in Sections 2.2 and 2.3, we can modify (16) to

$$f(x) = \sum_{i=1}^{n}\sum_{p=1}^{n} 1(H_{(i;p)}(x)) d_{ip} + 1(\neg H_C(x)) L \tag{20}$$

where $1(\cdot)$ is an indicator function. Thus, by the definition of SLP, we obtain the following.

THEOREM 2. *TSP is a SLP.*

Hence by applying the Summation Rule to (20), we can obtain the followings.

$$f(H) = \frac{1}{|\mathcal{U}_H|}\left(\sum_{i=1}^{n}\sum_{p=1}^{n} d_{ip} |\mathcal{U}_{H_{(i;p)} \wedge H}| + L|\mathcal{U}_{\neg H_C \wedge H}|\right) \tag{21}$$

$$\begin{aligned} & f^{(2)}(H) \\ = \ & \frac{1}{|\mathcal{U}_H|}\left(\sum_{i=1}^{n}\sum_{p=1}^{n}\sum_{j=1}^{n}\sum_{q=1}^{n} d_{ip} d_{jq} |\mathcal{U}_{H_{(i;p)} \wedge H_{(j;q)} \wedge H}| \right. \\ & + \sum_{i=1}^{n}\sum_{p=1}^{n} (d_{ip} L + L d_{ip}) \, |\mathcal{U}_{H_{(i;p)} \wedge \neg H_C \wedge H}| \\ & \left. + \ L^2 |\mathcal{U}_{\neg H_C \wedge H}|\right) \end{aligned} \tag{22}$$

$$\begin{aligned} = \ & \frac{1}{|\mathcal{U}_H|}\left(\sum_{i=1}^{n}\sum_{p=1}^{n}\sum_{j=1}^{n}\sum_{q=1}^{n} d_{ip} d_{jq} |\mathcal{U}_{H_{(i;p)} \wedge H_{(j;q)} \wedge H}| \right. \\ & + 2\sum_{i=1}^{n}\sum_{p=1}^{n} d_{ip} L |\mathcal{U}_{H_{(i;p)} \wedge \neg H_C \wedge H}| \\ & \left. + \ L^2 |\mathcal{U}_{\neg H_C \wedge H}|\right) \end{aligned}$$

From (17), (18), (19), (21), (22), $d_{ii} = 0$, and the symmetry assumption $d_{ij} = d_{ji}$, we can obtain the following equations.

PROPOSITION 1.

$$f(\textbf{true}) = \frac{1}{n-1} d_{\cdot\cdot} + \left(1 - \frac{(n-1)!}{(n-1)^n}\right) L \tag{23}$$

Proof. See Appendix A. □

PROPOSITION 2.

$$\begin{aligned} f(H_{(i;p)}) \quad = \quad & d_{ip} + \frac{1}{n-1}(d_{\cdot\cdot} - d_{i\cdot}) \\ & + \left(1 - \frac{(n-2)!}{(n-1)^{n-1}}\right) L \end{aligned} \tag{24}$$

Proof. See Appendix A. □

PROPOSITION 3.

$$\begin{aligned} & f^{(2)}(\textbf{true}) \\ = \ & \frac{1}{n-1}\sum_{k=1}^{n}\sum_{r=1}^{n} d_{kr}^2 + \frac{1}{(n-1)^2}\left(d_{\cdot\cdot}^2 - \sum_{k=1}^{n} d_{k\cdot}^2\right) \\ & + \left(\frac{2}{n-1} - \frac{2(n-2)!}{(n-1)^n}\right) d_{\cdot\cdot} L \\ & + \left(1 - \frac{(n-1)!}{(n-1)^n}\right) L^2 \end{aligned} \tag{25}$$

Proof. See Appendix A. □

By applying (23), (24), and (25) to the numerator and the denominator of (13), we can obtain the followings.

PROPOSITION 4.

$$\begin{aligned}&\sum_{x\in\mathcal{U}}\left(f(x)-\sum_{i=1}^{n}f(H_{(i;x_i)})+(n-1)f(\mathbf{true})\right)^2\\ &= \left((n-1)!-\frac{((n-1)!)^2}{(n-1)^n}\right)L^2\end{aligned} \tag{26}$$

Proof. See Appendix A. □

PROPOSITION 5.

$$\begin{aligned}&\sum_{x\in\mathcal{U}}(f(x)-f(\mathbf{true}))^2\\ &= (n-1)^n\left(\frac{1}{n-1}\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr}^2-\frac{1}{(n-1)^2}\sum_{k=1}^{n}d_{k\cdot}^2\right.\\ &\quad\left.+\frac{(n-1)!}{(n-1)^n}\left(1-\frac{(n-1)!}{(n-1)^n}\right)L^2\right)\end{aligned} \tag{27}$$

Proof. See Appendix A. □

From the above equations, we can obtain the following result:

THEOREM 3. *Given a symmetric TSP of size n and distance matrix (d_{ij}), the epistasis variance η is*

$$\eta=\frac{1}{1+\frac{\beta}{\alpha(1-\alpha)}} \tag{28}$$

where $L=\sum_{i=1}^{n}\max_{j\in\mathcal{A}_i}\{d_{ij}\}$, $\alpha=(n-1)!/(n-1)^n$, and $\beta=\sum_{k=1}^{n}\mathrm{Var}_{r\in\mathcal{A}_k}\{d_{kr}/L\}$.

Proof. Let $\alpha=(n-1)!/(n-1)^n$ and $\beta_1=\left\{\frac{1}{n-1}\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr}^2-\frac{1}{(n-1)^2}\sum_{k=1}^{n}d_{k\cdot}^2\right\}/L^2$. By applying these to (26) and (27), respectively, we can obtain

$$\begin{aligned}&\sum_{x\in\mathcal{U}}\left(f(x)-\sum_{i=1}^{n}f(H_{(i;x_i)})+(n-1)f(\mathbf{true})\right)^2\\ &= \alpha(1-\alpha)L^2(n-1)^n\end{aligned}$$

and

$$\begin{aligned}&\sum_{x\in\mathcal{U}}(f(x)-f(\mathbf{true}))^2\\ &= \left(\alpha(1-\alpha)L^2+\beta_1L^2\right)(n-1)^n.\end{aligned}$$

Hence the epistasis variance η is

$$\begin{aligned}\eta &= \frac{\alpha(1-\alpha)L^2}{\alpha(1-\alpha)L^2+\beta_1L^2}\\ &= \frac{1}{1+\frac{\beta_1}{\alpha(1-\alpha)}}.\end{aligned}$$

Now, we can modify β_1 as

$$\begin{aligned}&\beta_1\\ &= \sum_{k=1}^{n}\frac{1}{n-1}\sum_{r=1}^{n}\left(\frac{d_{kr}}{L}\right)^2-\sum_{k=1}^{n}\left(\frac{\frac{d_{k\cdot}}{L}}{n-1}\right)^2\\ &= \sum_{k=1}^{n}\left(\frac{1}{n-1}\sum_{r=1}^{n}\left(\frac{d_{kr}}{L}\right)^2-\left(\frac{1}{n-1}\sum_{r=1}^{n}\frac{d_{kr}}{L}\right)^2\right)\\ &= \sum_{k=1}^{n}\left(\underset{r\in\mathcal{A}_k}{\mathrm{Avg}}\left\{\left(\frac{d_{kr}}{L}\right)^2\right\}-\left(\underset{r\in\mathcal{A}_k}{\mathrm{Avg}}\left\{\frac{d_{kr}}{L}\right\}\right)^2\right)\\ &= \sum_{k=1}^{n}\underset{r\in\mathcal{A}_k}{\mathrm{Var}}\left\{\frac{d_{kr}}{L}\right\}\;=\;\beta.\end{aligned}$$

□

In (28), $\alpha(1-\alpha)$ is a factor depending only on the problem size n and β is a factor depending on the problem's characteristic given by the distance matrix (d_{ij}). Thus, we can catch the epistatic properties of a TSP instance by computing its β value.

Now, we see that β means the sum of variances $\mathrm{Var}_{r\in\mathcal{A}_k}\{d_{kr}/L\}$ for $k\in\mathcal{V}$. To cancel out the effect of problem size n that still remains in β, we define β times n as another parameter γ as follows:

$$\begin{aligned}\gamma &= n\beta\\ &= n\sum_{k=1}^{n}\underset{r\in\mathcal{A}_k}{\mathrm{Var}}\left\{\frac{d_{kr}}{L}\right\}\\ &= \frac{1}{n}\sum_{k=1}^{n}\underset{r\in\mathcal{A}_k}{\mathrm{Var}}\left\{\frac{nd_{kr}}{L}\right\}\\ &= \underset{k\in\mathcal{V}}{\mathrm{Avg}}\left\{\underset{r\in\mathcal{A}_k}{\mathrm{Var}}\left\{\frac{nd_{kr}}{L}\right\}\right\}.\end{aligned} \tag{29}$$

Thus, we can say that γ reflects the net epistasis in TSP instances based on the theory of [4, 17, 18]. That is, γ can be used as an alternative measure of epistasis for symmetric TSP. Note that γ is an inverse measure of epistasis since β is in inverse proportion to η in (28).

The computational complexities of η, β, and γ are identically $O(n^2)$.

4. EXPERIMENTS

Table 1 shows the results of simple experiments conducted to confirm the validity of Theorem 3. The experiments were performed on 14 TSP instances of sizes 4 through 11849 obtained from TSPLIB[2] on Intel Pentium III 1.0 GHz system running Linux. The first seven instances in the table are those reduced from lin318 by removing cities except the first n cities for each problem size n. Each row of the table shows instance name, problem size, the epistasis variance η_{ex} computed from (9), the computation time of η_{ex}, the epistasis variance η_{eq} computed from (28), the computation time of η_{eq}, two relevant parameters α and β, and the new inverse measure γ in sequence. We omitted the third and fourth

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

Table 1: Epistasis Variances η_{ex} by Equation (9) vs. η_{eq} by Equation (28)

Instance	Size	η_{ex}	Time$_{ex}$ (s)	η_{eq}	Time$_{eq}$ (s)	α	β	γ
lin4	4	0.699170	0.0002	0.699170	0.00001	0.074074	0.029511	0.118043
lin6	6	0.231814	0.04	0.231814	0.00002	0.007680	0.025255	0.151527
lin8	8	0.052798	19.29	0.052798	0.00004	0.000874	0.015671	0.125366
lin10	10	0.009462	14103.83	0.009462	0.00006	0.000104	0.010894	0.108940
lin16	16	–	–	0.000031	0.0002	0.000000	0.006330	0.101272
lin32	32	–	–	0.000000	0.0007	0.000000	0.002015	0.064485
lin64	64	–	–	0.000000	0.003	0.000000	0.000896	0.057334
lin105	105	–	–	0.000000	0.01	0.000000	0.000636	0.066769
lin318	318	–	–	0.000000	0.07	0.000000	0.000164	0.052093
att532	532	–	–	0.000000	0.23	0.000000	0.000118	0.062920
dsj1000	1000	–	–	0.000000	0.66	0.000000	0.000059	0.059227
pcb3038	3038	–	–	0.000000	6.04	0.000000	0.000016	0.049391
fnl4461	4461	–	–	0.000000	13.01	0.000000	0.000012	0.052881
rl11849	11849	–	–	0.000000	97.06	0.000000	0.000004	0.045158

column values of the instances of more than 10 cities by the huge time requirement of computation. For example, it took 14103.83 seconds for computing η_{ex} of lin10, while it took 0.00006 seconds for computing η_{eq} of the same instance.

By these experiments, we could confirm that the results η_{ex} from (9) and η_{eq} from (28) are exactly the same. We also observed that η and α rapidly converged to zero as the problem size increases. Moreover, β also went to zero along with η, although the speed of convergence was slower to some extent. On the contrary, we observed that γ was not considerably affected by the problem size, that means it is more scalable than the original.

The overall results of the experiments say that Reeves and Wright's normalization method for Davidor's epistasis variance is less scalable to symmetric TSP, and the new inverse measure γ, induced from the original measure η by eliminating the effect of problem size, is tolerant well of the size effect.

5. CONCLUDING REMARKS

In this paper, we addressed the problem of enormous computation time of the epistasis variance, an epistasis measure proposed by Davidor [4] and normalized later by Reeves and Wright [17, 18]. To reduce the computational complexity of the original equation (9), we defined a new category of combinatorial optimization problem, called schema-linear problem (SLP), and showed that TSP is a SLP. To define SLP, we extended Holland's schema definition to first-order logic expressions of the solution variables. Using the Summation Rule, which is a useful lemma applicable to SLP, we devised a reduced version (28) of (9). By the equation, the epistasis variance can be computed in $O(n^2)$ time, which is a dramatic reduction, instead of original $O(n^n)$ time.

Simple experiments showed that the normalized epistasis variance is less scalable to symmetric TSP. Thus, we proposed a new inverse measure γ of epistasis that inherits only the epistasis factor of η by eliminating the size factor in η. The experimental results showed that the new inverse measure γ is not considerably affected by the problem size. We hope this measure will be widely used for further studies about the epistatic properties of TSP and expect that the scheme introduced in this paper will be able to be applied to other practical combinatorial optimization problems.

Acknowledgments

This work was supported by the Brain Korea 21 Project. The ICT at Seoul National University provided research facilities for this study.

6. REFERENCES

[1] Aarts, E. H. and Lenstra, J. K. (1997). Introduction. In Aarts, E. H. and Lenstra, J. K., editors, *Local Search in Combinatorial Optimization*, pages 1–17, John Wiley & Sons, New York, NY.

[2] Bethke, A. D. (1981). *Genetic Algorithms as Function Optimizers*. PhD thesis, University of Illinois, Urbana, IL.

[3] Bui, T. N. and Moon, B. R. (1994). A new genetic approach for the traveling salesman problem. In Michalewicz, Z. et al., editors, *IEEE Conference on Evolutionary Computation*, pages 7–12, IEEE Service Center, Piscataway, NJ.

[4] Davidor, Y. (1990). Epistasis variance: Suitability of a representation to genetic algorithms. *Complex Systems*, 4:369–383.

[5] Forrest, S. and Mitchell, M. (1993). Relative building-block fitness and the building-block hypothesis. In Whitley, L. D., editor, *Foundations of Genetic Algorithms 2*, pages 109–126, Morgan Kaufmann Publishers, San Francisco, CA.

[6] Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, NY.

[7] Goldberg, D. E. (1989a). Genetic algorithms and Walsh functions: Part I, a gentle introduction. *Complex Systems*, 3:129–152.

[8] Goldberg, D. E. (1989b). Genetic algorithms and Walsh functions: Part II, deception and its analysis. *Complex Systems*, 3:153–171.

[9] Heckendorn, R. B. and Wright, A. H. (2004). Efficient linkage discovery by limited probing. *Evolutionary Computation*, 12(4):517–545.

[10] Holland, J. (1992). *Adaptation in Natural and Artificial Systems*. The MIT Press, Cambridge, MA.

[11] Larrañaga, P. and Lozano, J. A. (2002). *Estimation of*

Distribution Algorithms: A New Tool for Evolutionary Computation. Kluwer Academic Publishers, Boston, MA.

[12] Mason, A. J. (1991). Partition coefficients, static deception and deceptive problems for non-binary alphabets. In Belew, R. K. and Booker, L. B., editors, *International Conference on Genetic Algorithms*, pages 210–214, Morgan Kaufmann Publishers, San Francisco, CA.

[13] Munetomo, M. and Goldberg, D. E. (1999). Linkage identification by non-monotonicity detection for overlapping functions. *Evolutionary Computation*, 7(4):377–398.

[14] Naudts, B. and Suys, D. and Verschoren, A. (1997). Epistasis as a basic concept in formal landscape analysis. In T. Bäck, editor, *International Conference on Genetic Algorithms*, pages 65–72, Morgan Kaufmann Publishers, San Francisco, CA.

[15] Papadimitriou, C. H. and Steiglitz, K. (1998). *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, Mineola, NY.

[16] Rana, S., Heckendorn, R. B., and Whitley, D. (1998). A tractable Walsh analysis of SAT and its implications for genetic algorithms. In Rich, C. and Mostow, J., editors, *National Conference on Artificial Intelligence*, pages 392–397, AAAI Press, Menlo Park, CA.

[17] Reeves, C. R. and Wright, C. C. (1995a). An experimental design perspective on genetic algorithms. In Whitley, L. D. and Vose, M. D., editors, *Foundations of Genetic Algorithms 3*, pages 7–22, Morgan Kaufmann Publishers, San Francisco, CA.

[18] Reeves, C. R. and Wright, C. C. (1995b). Epistasis in genetic algorithms: An experimental design perspective. In Eshelman, L. J., editor, *International Conference on Genetic Algorithms*, pages 217–224, Morgan Kaufmann Publishers, San Francisco, CA.

[19] Russell, S. and Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Prentice Hall, Englewood Cliffs, NJ.

[20] Seo, D. I., Choi, S. S., Kim, Y. H., and Moon, B. R. (2005). A new measure of entropic epistasis for combinatorial optimization problems. *Evolutionary Computation*, submitted.

[21] Seo, D. I., Kim, Y. H., and Moon, B. R. (2003). New entropy-based measures of gene significance and epistasis. In Cantú-Paz, E. et al., editors, *Genetic and Evolutionary Computation Conference*, pages 1345–1546, Springer-Verlag, Berlin, Germany.

[22] Seo, D. I. and Moon, B. R. (2003). A survey on chromosomal structures and operators for exploiting topological linkages of genes. In Cantú-Paz, E. et al., editors, *Genetic and Evolutionary Computation Conference*, pages 1357–1368, Springer-Verlag, Berlin, Germany.

[23] Streeter, M. J. (2004). Upper bounds on the time and space complexity of optimizing additively separable functions. In Deb, K. et al., editors, *Genetic and Evolutionary Computation Conference*, pages 186–197, Springer-Verlag, Berlin, Germany.

APPENDIX

A. PROOFS

Proof of Proposition 1.

$$
\begin{aligned}
& f(\mathbf{true}) \\
= \;& \frac{1}{|\mathcal{U}_{\mathbf{true}}|}\left(\sum_{k=1}^{n}\sum_{r=1}^{n}\left(d_{kr}|\mathcal{U}_{H_{(k;r)}}|\right)+L|\mathcal{U}_{\neg H_C}|\right) \\
= \;& \frac{1}{(n-1)^n}\left(\sum_{k=1}^{n}\sum_{r=1}^{n}(d_{kr}(n-1)^{n-1})\right. \\
& \left.+L((n-1)^n-(n-1)!)\right) \\
= \;& \frac{1}{(n-1)^n}\left((n-1)^{n-1}\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr}\right. \\
& \left.+((n-1)^n-(n-1)!)L\right) \\
= \;& \frac{1}{n-1}d_{..}+\left(1-\frac{(n-1)!}{(n-1)^n}\right)L
\end{aligned}
$$

□

Proof of Proposition 2.

$$
\begin{aligned}
& f(H_{(i;p)}) \\
= \;& \frac{1}{|\mathcal{U}_{H_{(i;p)}}|}\left(\sum_{k=1}^{n}\sum_{r=1}^{n}\left(d_{kr}|\mathcal{U}_{H_{(k;r)}\wedge H_{(i;p)}}|\right)\right. \\
& \left.+L|\mathcal{U}_{\neg H_C\wedge H_{(i;p)}}|\right) \\
= \;& \frac{1}{(n-1)^{n-1}}\left(d_{ip}|\mathcal{U}_{H_{(i;p)}\wedge H_{(i;p)}}|\right. \\
& \left.+\sum_{\substack{k=1,\\k\neq i}}^{n}\sum_{r=1}^{n}\left(d_{kr}|\mathcal{U}_{H_{(k;r)}\wedge H_{(i;p)}}|\right)+L|\mathcal{U}_{\neg H_C\wedge H_{(i;p)}}|\right) \\
= \;& \frac{1}{(n-1)^{n-1}}\left(d_{ip}(n-1)^{n-1}+\sum_{\substack{k=1,\\k\neq i}}^{n}\sum_{r=1}^{n}(d_{kr}(n-1)^{n-2})\right. \\
& \left.+L((n-1)^{n-1}-(n-2)!)\right) \\
= \;& \frac{1}{(n-1)^{n-1}}\left((n-1)^{n-1}d_{ip}+(n-1)^{n-2}\sum_{\substack{k=1,\\k\neq i}}^{n}\sum_{r=1}^{n}d_{kr}\right. \\
& \left.+((n-1)^{n-1}-(n-2)!)L\right) \\
= \;& \frac{1}{(n-1)^{n-1}}((n-1)^{n-1}d_{ip}+(n-1)^{n-2}(d_{..}-d_{i.}) \\
& +((n-1)^{n-1}-(n-2)!)L) \\
= \;& d_{ip}+\frac{1}{n-1}(d_{..}-d_{i.})+\left(1-\frac{(n-2)!}{(n-1)^{n-1}}\right)L
\end{aligned}
$$

□

Proof of Proposition 3.

$$
\begin{aligned}
& f^{(2)}(\mathbf{true}) \\
= \; & \frac{1}{|\mathcal{U}_{\mathbf{true}}|}\left(\sum_{k=1}^{n}\sum_{r=1}^{n}\sum_{l=1}^{n}\sum_{s=1}^{n}(d_{kr}d_{ls}|\mathcal{U}_{H_{(k;r)}\wedge H_{l;s}}|)\right. \\
& \left.+2\sum_{k=1}^{n}\sum_{r=1}^{n}(d_{kr}L|\mathcal{U}_{H_{(k;r)}\wedge\neg H_C}|)+\ L^2|\mathcal{U}_{\neg H_C}|\right) \\
= \; & \frac{1}{|\mathcal{U}_{\mathbf{true}}|}\left(\sum_{k=1}^{n}\sum_{r=1}^{n}\left(d_{kr}\left(d_{kr}|\mathcal{U}_{H_{(k;r)}\wedge H_{k;r}}|\right.\right.\right. \\
& \left.\left.\left.+\sum_{\substack{l=1\\l\neq k}}^{n}\sum_{s=1}^{n}(d_{ls}|\mathcal{U}_{H_{(k;r)}\wedge H_{(l;s)}}|)\right)\right)\right. \\
& \left.+2\sum_{k=1}^{n}\sum_{r=1}^{n}(d_{kr}L|\mathcal{U}_{H_{(k;r)}\wedge\neg H_C}|)+\ L^2|\mathcal{U}_{\neg H_C}|\right) \\
= \; & \frac{1}{(n-1)^n}\left(\sum_{k=1}^{n}\sum_{r=1}^{n}\left(d_{kr}\left(d_{kr}(n-1)^{n-1}\right.\right.\right. \\
& \left.\left.\left.+\sum_{\substack{l=1\\l\neq k}}^{n}\sum_{s=1}^{n}(d_{ls}(n-1)^{n-2})\right)\right)\right. \\
& +2\sum_{k=1}^{n}\sum_{r=1}^{n}(d_{kr}L((n-1)^{n-1}-(n-2)!)) \\
& \left.+\ L^2((n-1)^n-(n-1)!)\right) \\
= \; & \frac{1}{(n-1)^n}\left(\sum_{k=1}^{n}\sum_{r=1}^{n}\left(d_{kr}\left((n-1)^{n-1}d_{kr}\right.\right.\right. \\
& \left.\left.\left.+(n-1)^{n-2}\left(\sum_{l=1}^{n}d_{l\cdot}-d_{k\cdot}\right)\right)\right)\right. \\
& +2((n-1)^{n-1}-(n-2)!)L\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr} \\
& \left.+((n-1)^n-(n-1)!)L^2\right) \\
= \; & \frac{1}{(n-1)^n}\left(\sum_{k=1}^{n}\sum_{r=1}^{n}(d_{kr}((n-1)^{n-1}d_{kr}\right. \\
& +(n-1)^{n-2}(d_{\cdot\cdot}-d_{k\cdot}))) \\
& +2((n-1)^{n-1}-(n-2)!)d_{\cdot\cdot}L \\
& \left.+((n-1)^n-(n-1)!)L^2\right) \\
= \; & \frac{1}{(n-1)^n}\left((n-1)^{n-1}\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr}^2\right. \\
& +(n-1)^{n-2}\left(d_{\cdot\cdot}\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr}-\sum_{k=1}^{n}\left(d_{k\cdot}\sum_{r=1}^{n}d_{kr}\right)\right) \\
& +2((n-1)^{n-1}-(n-2)!)d_{\cdot\cdot}L \\
& \left.+((n-1)^n-(n-1)!)L^2\right) \\
= \; & \frac{1}{n-1}\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr}^2+\frac{1}{(n-1)^2}\left(d_{\cdot\cdot}^2-\sum_{k=1}^{n}d_{k\cdot}^2\right) \\
& +\left(\frac{2}{n-1}-\frac{2(n-2)!}{(n-1)^n}\right)d_{\cdot\cdot}L+\left(1-\frac{(n-1)!}{(n-1)^n}\right)L^2
\end{aligned}
$$

□

Proof of Proposition 4.

$$
\begin{aligned}
& \sum_{x\in\mathcal{U}}\left(f(x)-\sum_{i=1}^{n}f(H_{(i;x_i)})+(n-1)f(\mathbf{true})\right)^2 \\
= \; & \sum_{x\in\mathcal{U}}\left(f(x)-\sum_{i=1}^{n}\left(d_{ix_i}+\frac{1}{n-1}(d_{\cdot\cdot}-d_{i\cdot})\right.\right. \\
& \left.+\left(1-\frac{(n-2)!}{(n-1)^{n-1}}\right)L\right) \\
& \left.+(n-1)\left(\frac{1}{(n-1)}d_{\cdot\cdot}+\left(1-\frac{(n-1)!}{(n-1)^n}\right)L\right)\right)^2 \\
= \; & \sum_{x\in\mathcal{U}}\left(f(x)-\sum_{i=1}^{n}d_{ix_i}-d_{\cdot\cdot}-n\left(1-\frac{(n-2)!}{(n-1)^{n-1}}\right)L\right. \\
& \left.+d_{\cdot\cdot}+(n-1)\left(1-\frac{(n-1)!}{(n-1)^n}\right)L\right)^2 \\
= \; & \sum_{x\in\mathcal{U}}\left(f(x)-\sum_{i=1}^{n}d_{ix_i}-\left(1-\frac{(n-1)!}{(n-1)^n}\right)L\right)^2 \\
= \; & \sum_{x\in\mathcal{U}_{H_C}}\left(\sum_{i=1}^{n}d_{ix_i}-\sum_{i=1}^{n}d_{ix_i}-\left(1-\frac{(n-1)!}{(n-1)^n}\right)L\right)^2 \\
& +\sum_{x\in\mathcal{U}_{\neg H_C}}\left(\sum_{i=1}^{n}d_{ix_i}+L-\sum_{i=1}^{n}d_{ix_i}\right. \\
& \left.-\left(1-\frac{(n-1)!}{(n-1)^n}\right)L\right)^2 \\
= \; & \sum_{x\in\mathcal{U}_{H_C}}\left(\left(1-\frac{(n-1)!}{(n-1)^n}\right)L\right)^2+\sum_{x\in\mathcal{U}_{\neg H_C}}\left(\frac{(n-1)!}{(n-1)^n}L\right)^2 \\
= \; & \left(\left(1-\frac{(n-1)!}{(n-1)^n}\right)L\right)^2(n-1)! \\
& +\left(\frac{(n-1)!}{(n-1)^n}L\right)^2((n-1)^n-(n-1)!) \\
= \; & \left((n-1)!-\frac{((n-1)!)^2}{(n-1)^n}\right)L^2
\end{aligned}
$$

□

Proof of Proposition 5.

$$
\begin{aligned}
& \sum_{x\in\mathcal{U}}(f(x)-f(\mathbf{true}))^2 \\
= \; & \sum_{x\in\mathcal{U}}\left(f(x)^2-2f(x)f(\mathbf{true})+f(\mathbf{true})^2\right) \\
= \; & (n-1)^n\left(f^{(2)}(\mathbf{true})-f(\mathbf{true})^2\right) \\
= \; & (n-1)^n\left(\frac{1}{n-1}\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr}^2+\frac{1}{(n-1)^2}\left(d_{\cdot\cdot}^2-\sum_{k=1}^{n}d_{k\cdot}^2\right)\right. \\
& +\left(\frac{2}{n-1}-\frac{2(n-2)!}{(n-1)^n}\right)d_{\cdot\cdot}L+\left(1-\frac{(n-1)!}{(n-1)^n}\right)L^2 \\
& \left.-\left(\frac{1}{n-1}d_{\cdot\cdot}+\left(1-\frac{(n-1)!}{(n-1)^n}\right)L\right)^2\right) \\
= \; & (n-1)^n\left(\frac{1}{n-1}\sum_{k=1}^{n}\sum_{r=1}^{n}d_{kr}^2-\frac{1}{(n-1)^2}\sum_{k=1}^{n}d_{k\cdot}^2\right. \\
& \left.+\frac{(n-1)!}{(n-1)^n}\left(1-\frac{(n-1)!}{(n-1)^n}\right)L^2\right)
\end{aligned}
$$

□