Vol. 49 No. SIG 4(TOM 20) IPSJ Transactions on Mathematical Modeling and Its Applications Mar. 2008

Regular Paper

Maintaining Multiple Populations with Different Diversities for Evolutionary Optimization Based on Probability Models

Takayuki Higo[†1] and Keiki Takadama[†2]

This paper proposes a novel method, Hierarchical Importance Sampling (HIS) that can be used instead of population convergence in evolutionary optimization based on probability models (EOPM) such as estimation of distribution algorithms and cross entropy methods. In HIS, multiple populations are maintained simultaneously such that they have different diversities, and the probability model of one population is built through importance sampling by mixing with the other populations. This mechanism can allow populations to escape from local optima. Experimental comparisons reveal that HIS outperforms general EOPM.

1. Introduction

Recently, evolutionary optimization based on probability models (EOPM), for example, estimation of distribution algorithms (EDAs) [6], cross entropy methods (CEs) [14], and probabilistic model-building genetic algorithms (PMBGAs) [10], have attracted considerable attention for they possess not only the strengths of genetic algorithms (GAs) [2] but also a substantial mathematical background. The essential concept employed in these methods involves building a probability model of the population, which is a set of promising solutions, and then generating samples from the built probability model.

In general EOPM, one population is maintained and is gradually converged; thus, population convergence plays an important role in EOPM. In EDAs, the convergence is controlled by a selection operator; Boltzmann selection, for example, is one of the promising selection operators, and the standard deviation schedule can effectively control its convergence speed [7]. However, population convergence is an unstable method because the obtained solutions cannot be further improved once the population has converged.

To overcome this instability, this paper proposes a novel method, Hierarchical Importance Sampling (HIS) that can be used instead of population convergence. The basic principle is to maintain multiple populations with different diversities [⋆1]. For example, one population may be almost random and another, almost converged. HIS builds the probability model of each population, respectively, and generates samples from all the built probability models simultaneously. Therefore, the obtained samples consist of a number of sample sets, each of which is generated from a different probability distribution. The salient feature is that mixed samples are used for building probability models of the populations according to importance sampling [1),13)], which guarantees mathematical validity.

The aim of this paper is to investigate the effectiveness of the proposed method through experimental comparisons with general EOPM. Onemax, a 1D Ising model, and a 2D Ising model, are used for benchmark problems.

The outline of the paper is as follows: In Section 2, general EOPM is explained from two different viewpoints. In Section 3, the proposed method (HIS) is described. In Section 4, the proposed method is experimentally compared with EOPM. Section 5 discusses the advantages of the proposed method. Finally, Section 6 concludes this paper.

2. Evolutionary Optimization Based on Probability Models

The following section provides two viewpoints

†1 Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology
†2 Faculty of Electro-communications, The University of Electro-Communications

⋆1 The exchange Monte Carlo method (EMC) [4] uses the same concept of sampling from multiple target distributions with different diversities. EMC is one of the Markov chain Monte Carlo methods (MCMC) [1]. MCMC and EMC are essentially different from EOPM and HIS. The relationships among EOPM, MCMC, HIS, and EMC are summarized in Appendix A.1.

Estimation of Distribution Algorithm (EDA)

1 Generate samples $X_{pop}^{(1)} = \{x_i\}_1^N$ from the uniform distribution $p_1(x)$. $t \Leftarrow 1$.
2 do{
3 Build a probability model $p_t(x)$ of $X_{pop}^{(t)}$.
4 Generate samples $X_{samp}^{(t)} = \{x_i\}_1^M$ from $p_t(x)$.
5 Select promising solutions as the next population $X_{pop}^{(t+1)} = \{x_i\}_1^N$ from $X_{samp}^{(t)}$.
6 $t \Leftarrow t + 1$.
7 }until(stopping criterion reached)

Fig. 1 The pseudo-code of EDA.

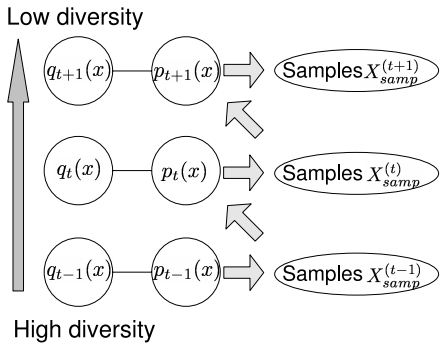


Fig. 2 Illustration of EDA and CE.

for EOPM: (1) EDA and (2) CE.

2.1 Estimation of Distribution Algorithm (EDA)

EDAs [6] mainly developed in the area of evolutionary computation and can be considered as mathematical representations of GAs. The univariate marginal distribution algorithm (UMDA) [8] is one of the basic EDAs. In this paper, the algorithm of EDA is defined as the generalization of UMDA.

The algorithm of EDA is summarized in **Fig. 1**. At the beginning, randomly generated samples are employed as the initial population, and this population is then updated iteratively. To update the current population, first, a probability model of the population is built, and samples are then generated from the probability model. The promising solutions in the generated samples are selected by means of a selection operator to form the next population, and, finally, the population is completely replaced with the selected samples, as illustrated in **Fig. 2**.

In general, maximum likelihood (ML) estimation [1] is used for building probability models in EDAs. Let $p(x)$ and $q(x)$ be a probability model and a target probability distribution, respectively. ML estimation selects the probability model that maximizes the (expected) log-likelihood, which is defined as follows:

$$L(p(x)) = \int q(x) \log p(x) dx. \tag{1}$$

In practice, the empirical log-likelihood is used as an estimator of the log-likelihood. By using the given samples X generated from $q(x)$, the empirical log-likelihood is calculated as follows:

$$L(p(x)) \simeq \frac{1}{N} \sum_X \log p(x), \tag{2}$$

where N is the number of samples in X.

In building a probability model of a population X_{pop}, it is assumed that X_{pop} is generated from a certain target distribution $q(x)$. The target distribution is naturally defined by the employed selection operator. For example, employing a truncation selection operator, which selects samples whose evaluations $f(x)$ are less than the threshold $\tilde{f}$ in a minimization problem, is almost equivalent [⋆1] to defining $q(x)$ as follows:

$$q(x|\tilde{f}) = \frac{1}{Z} \tilde{q}(x|\tilde{f}) \tag{3}$$

$$\tilde{q}(x|\tilde{f}) = I(f(x) < \tilde{f}) = \begin{cases} 1 & f(x) < \tilde{f} \\ 0 & \text{else} \end{cases}, \tag{4}$$

where $I(\cdot)$ is an indicator function and Z is the normalizing constant defined as follows:

$$Z = \int \tilde{q}(x) dx. \tag{5}$$

In this paper, this probability distribution is called a partially uniform distribution. Another candidate target distribution is the Boltzmann distribution [7].

2.2 Cross Entropy method (CE)

CE [14] was originally proposed as a sampling method in the area of rare-event simulations. The difference from EDA is that the target distributions described in Section 2.1 are explicitly

⋆1 It is assumed that ML estimation affords a perfect probability model, that is, $p(x) = q(x)$.

defined instead of using a selection operator. Consequently, the empirical log-likelihood is calculated from the previously generated samples $X_{samp}^{(t)}$ through importance sampling [13] as follows:

$$L \simeq \frac{1}{M} \sum_{X_{samp}^{(t)}} \frac{q_{t+1}(x)}{p_t(x)} \log p_{t+1}(x), \qquad (6)$$

where $X_{samp}^{(t)}$ is a set of samples generated from $p_t(x)$ and M is the number of the samples. If we only know $\tilde{q}_{t+1}(x)$ and $\tilde{p}_t(x)$ which are proportional to $q_{t+1}(x)$ and $p_t(x)$, respectively, the empirical log-likelihood can be calculated as follows:

$$\frac{1}{\sum_{X_{samp}^{(t)}} \frac{\tilde{q}_{t+1}(x)}{\tilde{p}_t(x)}} \sum_{X_{samp}^{(t)}} \frac{\tilde{q}_{t+1}(x)}{\tilde{p}_t(x)} \log p_{t+1}(x). \qquad (7)$$

The validity of Eq. (7) is proven in Appendix A.2. Note that the empirical log-likelihood is simply an estimator of the expected log-likelihood, and the accuracy (e.g., the variance) depends on the similarity between $p_t(x)$ and $q_{t+1}(x)$.

2.3 Comparison between EDA and CE

From the viewpoint of CE, the selection operators in EDA can be considered equivalent to resampling according to weights $q(x)/p(x)$ in Eq. (6); consequently EDA is equivalent to CE. Since resampling requires a sufficiently large number of samples, importance sampling can be superior to resampling and averaging, and CE can be superior to EDA. However, note that selection operators are used for not only calculating the experimental log-likelihoods but also defining the target distributions. In practice, the target distributions play an important role, and EDA's selection operators such as the truncation selection operator afford good experimental results, whereas CE with the $(1-\delta)$-quantile [14] has difficulties in convergence, as revealed in our experiments described later.

3. Hierarchical Importance Sampling

3.1 Theoretical Overview

HIS maintains L number of layers, each of which consists of a population X_l, a probability model $p_l(x)$, and a target distribution $q_l(x)$. Each X_l is a set of samples generated from the corresponding probability model $p_l(x)$. Each $p_l(x)$ is built with ML estimation to approximate the corresponding target distribution $q_l(x)$, which is assumed to be previously provided here. Thus, X_l is approximately distributed according to $q_l(x)$. It is supposed that $q_l(x)$ has less diversity than $q_{l-1}(x)$. Therefore, it is also expected that $p_l(x)$ has less diversity than $p_{l-1}(x)$, and X_l contains better solutions than X_{l-1}. Normally, $q_0(x)$ is the uniform distribution, and $q_{L-1}(x)$ is the converged distribution, which generates only the best obtained solution.

Basically, HIS iterates the following two steps: (1) sampling and (2) estimation. In the sampling step, each X_l is updated by sampling from $p_l(x)$ and replacing the current population with the generated samples; the sampling step is illustrated in **Fig. 3** (a). In the estimation step, each $p_l(x)$ is updated to approximate $q_l(x)$ more accurately than the previous one. The important feature is that all the populations $X_m = X_0 \cup \cdots \cup X_{L-1}$ are used for updating each $p_l(x)$. The probability distribution of X_m is given by a mixture distribution, which is defined as follows:

$$p_m(x) = \sum_l \alpha_l p_l(x), \qquad (8)$$

$$\alpha_l = \frac{M_l}{\sum_i M_i}, \qquad (9)$$

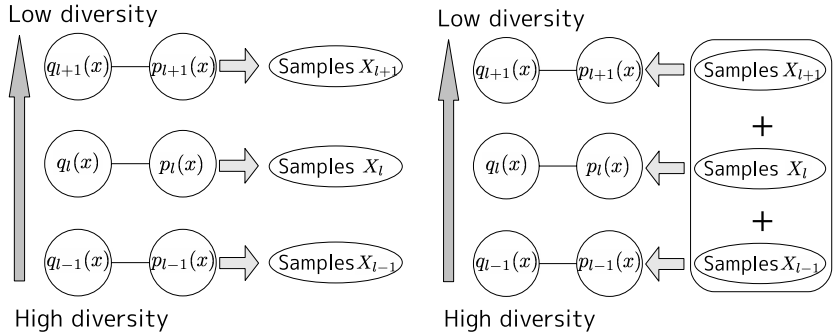
where M_l is the number of samples in X_l; thereby, the empirical log-likelihood with respect to $q_l(x)$ can be calculated via importance sampling as follows:

$$L \simeq \frac{1}{\sum_i M_i} \sum_{X_m} \frac{q_l(x)}{p_m(x)} \log p_l(x). \qquad (10)$$

This corresponds to Eq. (6) and the method of Eq. (7) would be employed in practice. The estimation step is illustrated in Fig. 3 (b).

3.2 Comparison between HIS and CE

Suppose that target distributions are previously provided in CE and HIS. Let L be the number of the layers of HIS. At time t, CE generates a probability model $p_t(x)$ approximating the corresponding target distribution $q_t(x)$, whereas HIS generates L number of probability models $p_0^{(t)}(x) \cdots p_{L-1}^{(t)}(x)$ approximating the corresponding target distributions $q_0(x) \cdots q_{L-1}(x)$, respectively. To generate $p_t(x)$, CE uses only one sample set X_{t-1}, which is generated in the previous step. On the other hand, to generate $p_l^{(t)}(x)$, HIS uses all the sample sets $X_0^{(t-1)} \cdots X_{L-1}^{(t-1)}$, generated in the previous step. In other words, the difference is that CE sequentially generates probabil-



(a) Sampling (b) Estimation

Fig. 3 Illustration of hierarchical importance sampling.

ity models and sample sets, whereas HIS generates probability models and sample sets both simultaneously and iteratively.

If only the $l-1$th population X_{l-1} is used to update the lth probability model $p_l(x)$ in the estimation step of HIS, HIS, indeed, corresponds to iterative execution of CE, which means that CE is restarted from the initialization if the population converges. This implies that HIS is a mathematical extension of CE.

3.3 Target Distribution Control

HIS can theoretically operate if the target distributions are previously defined in any manner. However, in practice, HIS requires appropriate target distributions to produce good results. This section explains a manner in which the target distributions of HIS are provided. Note that the proposed target distribution control method cannot be used with any probability distribution other than the partially uniform distribution defined by Eq. (3) for target distributions. To propose a control method for other probability distribution families such as the Boltzmann distribution is left as a future work.

It is supposed that $q_0(x)$ and $q_{L-1}(x)$ are given[⋆1]; then the objective of the control method is to determine $q_l(x)$ for $l = 1 \cdots L-2$. Each $q_l(x)$ is represented by the partially uniform distribution and denoted by $q_l(x|\tilde{f}_l)$ with the threshold parameter $\tilde{f}$. In terms of importance sampling, $q_{l-1}(x)$ and the next target distribution $q_l(x)$ should be similar because the accuracy of the empirical log-likelihood given by the importance sampling depends on this similarity. Thus, the objective is to select $\tilde{f}_l$ such that $q_{l-1}(x|\tilde{f}_{l-1})$, $q_l(x|\tilde{f}_l)$, and $q_{l+1}(x|\tilde{f}_{l+1})$ are similar.

The present concept is based on the size of the search space. In the case of the partially uniform distribution, a set of drawable samples is defined by $C_l = \{x|\tilde{q}(x|\tilde{f}_l) = 1\}$, where $\tilde{q}(x|\tilde{f})$ is defined by Eq. (4), and the number of drawable samples is given by $\int_C dx = \int \tilde{q}(x)dx = Z$. Thus, the size of the search space can be provided by the normalizing constant defined by Eq. (5). Note that the normalizing constant is normally unknown, but its estimator can be calculated through importance sampling as follows:

$$\begin{aligned} Z_l(\tilde{f}) &= \int \tilde{q}(x|\tilde{f})dx \\ &\simeq \frac{1}{M}\sum_{X_{p(x)}} \frac{\tilde{q}(x|\tilde{f})}{p(x)} \\ &= \hat{Z}_l(\tilde{f}), \end{aligned} \tag{11}$$

where $X_{p(x)}$ is a set of samples generated from $p(x)$, and M is the number of the samples. In an importance sampling calculation,

$$\frac{1}{M}\sum_{X_{q_{l-1}(x)}} \frac{q_l(x)}{q_{l-1}(x)} f(x), \tag{12}$$

the probability of generating an acceptable sample, whose weight $\frac{q_l(x)}{q_{l-1}(x)}$ is not zero, is given by

$$\int_{C_{l-1}} q_{l-1}(x)\frac{\tilde{q}_l(x)}{\tilde{q}_{l-1}(x)}dx = \frac{Z_l}{Z_{l-1}}, \tag{13}$$

where it is assumed that $C_l \subseteq C_{l-1}$. It is clear that the rejected samples do not contribute to the importance sampling. In simple CE, where samples are generated from target distri-

⋆1 In the experiments, a probability distribution that generates only the best obtained sample is used for $q_{L-1}(x)$.

Hierarchical Importance Sampling (HIS)

1 Initialize the probability models $p_0(x) \cdots p_{L-1}(x)$ and the populations $X_0 \cdots X_{L-1}$.
$l \Leftarrow 0$.
2 do{
3 Adjust the target distribution $q_l(x)$ according to Eq. (15).
4 Calculate the empirical log-likelihood with respect to $q_l(x)$ from X_{l-1}, X_l, X_{l+1} through importance sampling according to Eq. (10).
5 Update the probability model $p_l(x)$ according to the empirical log-likelihood.
6 Generate samples from $p_l(x)$ and replace population X_l with the generated samples.
7 $l \Leftarrow (l+1)\%L$.
8 }until(stopping criterion reached)

Fig. 5 The pseudo-code of hierarchical importance sampling.

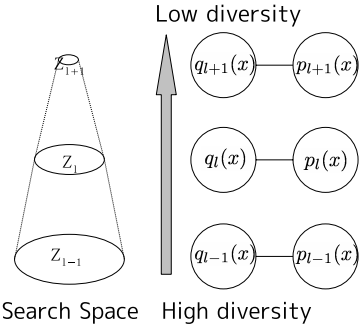


Fig. 4 Search space reduction.

butions, the sum of the number of the accepted samples is given by

$$\sum_{l=1}^{L-1} M_{l-1} \frac{Z_l}{Z_{l-1}}, \tag{14}$$

and the maximization condition of Eq. (14) is given by

$$M_{l-1} \frac{Z_l}{Z_{l-1}} = M_l \frac{Z_{l+1}}{Z_l}. \tag{15}$$

If Z_{l-1} and Z_{l+1} are given ⋆1, the target normalizing constant Z_l^* is obtained from Eq. (15). Then, the threshold parameter $\tilde{f}_l$ is updated to satisfy

$$Z_l^* = \hat{Z}_l(\tilde{f}_l), \tag{16}$$

where $\hat{Z}_l(\tilde{f}_l)$ is the estimator of the normalizing constant given by Eq. (11). A method for solving Eq. (16) is described in Appendix A.3. **Figure 4** shows an illustration of the search space reduction.

⋆1 Note that Z_0 and Z_{L-1} are normally previously provided and thus, all Z_l can be previously determined according to Eq. (15). However, this paper uses the estimators of Z_{l-1} and Z_{l+1} to determine Z_l because, in some cases, it can be difficult to build a probability model approximating a target distribution with a certain normalizing constant.

3.4 Practical Procedure

In the practical procedure of HIS, first of all, each $p_l(x)$ is initialized to a uniform distribution and each X_l is generated from $p_l(x)$. For each l, the lth layer (i.e., $q_l(x)$, $p_l(x)$, and X_l) is sequentially and iteratively updated. To update the lth layer, first, $q_l(x)$ is updated according to Eq. (15), and then $p_l(x)$ is updated. To calculate the empirical log-likelihood with respect to $q_l(x)$, we use only three populations, which are the upper one X_{l-1}, the current one X_l, and the lower one X_{l+1} ⋆2 for two reasons: calculating the marginal probability of Eq. (8) consumes a considerable amount of time, and the samples in X_i, generated from $p_i(x)$, tend not to contribute to the importance sampling of Eq. (10) if $p_i(x)$ and $q_l(x)$ are not similar. Finally, the population X_l is replaced with samples generated from $p_l(x)$. The pseudo-code of HIS is shown in **Fig. 5**.

4. Experiments

This section describes the experiments conducted to investigate the advantages of extending EDA and CE with HIS. There have been two types of developments in EDA and CE: one involves employing complex probability models such as Bayesian networks [6]. It is clear that HIS can employ any of the probability models used in EDA and CE. This section focuses on the simplest probability model, that is, a fully factorized one. This is because, for the first step in investigating the effects of HIS, the basic probability model is appropriate in terms of avoiding over-fitting. Instead of changing the complexity of the probability models, different kinds of problems are used for the experiments. As future work, investigations on the effects of model errors on HIS will be performed

⋆2 X_{-1} and X_L are supposed to be null sets.

by changing the complexity of the probability models.

The other development is to employ a population mechanism, that is, to maintain a part of the historical samples, whereas general EDA and CE maintain only the samples generated in the previous step. One successful method with a population mechanism is hBOA [11]. Since the population mechanism of hBOA is heuristic, it is difficult to simply extend hBOA with HIS; this line of inquiry is also set aside for future work.

Three benchmark problems, Onemax, a 1D Ising model, and a 2D Ising model, are employed. Onemax is a basic benchmark for EOPM. 1D and 2D Ising models are simple examples of Ising spin glasses, which are famous in both statistical physics and optimization [12]. A feature of 1D and 2D Ising models is the difficulty in statistically estimating their cost functions with fully factorized probability models, which intuitively implies the presence of multiple local optima.

4.1 Benchmark Problems

In the benchmark problems, the domain for each variable is $x_i \in \{0, 1\}$ and the number of the dimension d is set at 400. Minimization problems are considered.

4.1.1 Onemax

This problem is defined as

$$f(x) = -\sum_{i=1}^{d} x_i. \tag{17}$$

The optimum cost function value is $-d$, and there is no correlation between any of the variables.

4.1.2 1D Ising model

This problem is defined as follows:

$$f(x) = -\sum_{i=1}^{d} J(x_i, x_{i+1}), \tag{18}$$

$$J(x_i, x_j) = \begin{cases} 1 & x_i = x_j \\ 0 & x_i \neq x_j \end{cases}. \tag{19}$$

Periodic boundary conditions, implying that x_{d+1} is treated as x_1, are employed. The optimum cost function value is $-d$. There are correlations between two variables, as illustrated in **Fig. 6**.

4.1.3 2D Ising model

We consider $r \times r = 20 \times 20$ grids, as illustrated in **Fig. 7**. If two connected variables attain the same value, the value of the cost function is improved. 2D Ising model can be defined as

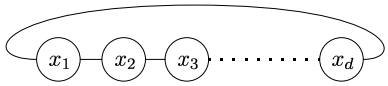


Fig. 6 1D Ising with periodic boundary conditions.

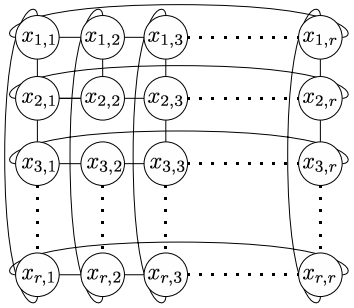


Fig. 7 2D Ising with periodic boundary conditions.

$$f(x) = -\sum_{i=1}^{i=r}\sum_{j=1}^{j=r}\{J(x_{ij}, x_{i+1,j}) + J(x_{ij}, x_{i,j+1})\}. \tag{20}$$

Periodic boundary conditions are employed. The optimum cost function value is $-2d$. This problem is basically equivalent to a check-board problem [6].

4.1.4 Adding Noise

Since the threshold of the partially uniform distribution cannot function precisely when multiple solutions have the same cost function value, the original cost function $f(x)$ is slightly altered by adding small random values ϵ as follows:

$$f'(x) = f(x) + \epsilon. \tag{21}$$

In the experiments, ϵ is $u \times 10^{-10}$, where u is a random number uniformly distributed from 0 to 1. This is applied to all the three functions described above.

4.2 Experimental Setup

4.2.1 EDA Setting

We employ UMDA [8] as the EDA. Thus, the probability model is defined as

$$p(x|w) = \prod_{i=1}^{i=d} p(x_i|w_i) \tag{22}$$

and ML estimation is employed for building the probability models. Here, the learning rate α is introduced. The parameter w is updated by the following equation:

$$w_{new} = (1 - \alpha)w_{\text{old}} + \alpha w_{\text{ML}}, \tag{23}$$

where w_{new}, w_{old}, w_{ML} are the new parameter, previous parameter, and ML estimator, respectively. This mechanism affords stable estima-

Table 1 Results of EDA for Onemax.

Samples	Cutoff	Best		Evaluations	
100	0.5	–400	(0)	8,570	(148.66)
500	0.5	–400	(0)	41,950	(610.33)
500	0.3	–400	(0)	64,750	(512.35)
1,000	0.5	–400	(0)	85,200	(1,326.65)
1,000	0.3	–400	(0)	130,200	(1,400)
500	0.1	–400	(0)	159,600	(2,406.24)
3,000	0.5	–400	(0)	260,400	(3,231.1)
1,000	0.1	–400	(0)	314,900	(4,109.74)
3,000	0.3	–400	(0)	391,800	(4,069.4)
6,000	0.5	–400	(0)	523,800	(7,613.15)
6,000	0.3	–400	(0)	792,000	(8,485.28)
3,000	0.1	–400	(0)	945,600	(5,969.92)
6,000	0.1	–400	(0)	1,891,200	(6,462.2)
100	0.3	–399.9	(0.3)	13,400	(275.68)
100	0.1	–395.1	(1.7)	36,030	(1,500.03)

tion.

The selection operator employed is the truncation selection operator. The truncation selection operator includes the cutoff rate parameter c, which represents the percentage of samples that are removed. For example, if $c = 0.3$ and the number of generated samples is 100, then the best $70 = 100\times(1-0.3)$ samples are selected and the rest are discarded. All the parameter settings are described as follows:

- The number of generated samples in one sampling M: 100, 500, 1,000, 3,000, or 6,000.
- Cutoff rate c: 0.1, 0.3, or 0.5.
- Learning rate α: 0.5.

These values are experimentally determined.

4.2.2 CE Setting

CE uses the same probability model and estimation method as EDA. However, instead of truncation selection, CE employs the $(1-\delta)$-quantile method [14], which selects the best $k = M \times \delta$ samples, where M is the number of generated samples, and removes the rest. Truncation selection and the $(1-\delta)$-quantile method are basically the same: the parameter $(1-\delta)$ corresponds to the cutoff rate in truncation selection. Thus, $(1-\delta)$ is referred to as the cutoff parameter in this paper. All the parameter settings are described as follows:

- The number of generated samples in one sampling M: 100, 500, 1,000, 3,000, or 6,000.
- Cutoff rate $c = 1-\delta$: 0.3, 0.5, or 0.7.
- Learning rate α: 0.5.

These values are experimentally determined.

4.2.3 HIS Setting

HIS also uses the same probability model and estimation method as EDA. All the parameter settings are described as follows:

- The number of generated samples in one sampling M : 10 or 50.
- The number of the layers L: 10, 20, 30, or 40.
- Learning rate α: 0.5.

These values are experimentally determined. Note that the number of samples contained in X_i is denoted by M_i and $M_i = M_j = M$.

4.3 Results

Tables 1, **2** and **3** show the results of EDA. **Tables 4**, **5** and **6** show the results of CE. The values in the first and second columns are the number of generated samples per sampling and the cutoff rate value, respectively. The third column lists the average cost function value, with the standard deviation in parenthesis, of the best obtained solutions over ten independent runs. The forth column lists the number of function evaluations until the population converges. The convergence criterion is that the number of function evaluations is greater than 2.9E6 or the variance of the cost function values of the generated samples is less than 1E−20.

Clearly, the performance of CE is inferior to that of EDA, despite the fact that CE is basically equivalent to or plausibly better than EDA. This is due to the target distributions (i.e., the difference between the truncation selection operator and the $(1-\delta)$ quantile method). The results show that the populations of CE do not converge well. This problem is solved by adding a population mechanism to CE [3].

Table 7 shows the results of HIS for Onemax. The first and the second columns list the number of generated samples per sampling and the number of layers, respectively. The third

Table 2 Results of EDA for 1D Ising.

Samples	Cutoff	Best		Evaluations	
3,000	0.3	–364.8	(4.02)	1,114,800	(62,183.29)
3,000	0.5	–364.4	(2.94)	788,400	(54,212.91)
1,000	0.5	–363.8	(6.54)	228,600	(22,037.24)
6,000	0.5	–363.6	(5.78)	1,839,600	(120,365.44)
6,000	0.3	–362.6	(4.2)	2,463,000	(118,922.66)
3,000	0.1	–360.8	(3.82)	2,260,200	(49,060.78)
1,000	0.3	–359.8	(4.33)	307,800	(12,064.82)
500	0.3	–358.4	(4.96)	146,900	(13,931.62)
500	0.5	–358	(3.9)	95,700	(4,648.66)
1,000	0.1	–356.8	(4.21)	663,200	(35,312.32)
100	0.5	–354	(6.69)	13,720	(570.61)
500	0.1	–352.8	(5.38)	308,550	(22,005.06)
100	0.3	–348.6	(4.39)	20,510	(1,328.5)
100	0.1	–338.2	(5.55)	45,170	(6,008.5)
6,000	0.1	–322.4	(5.35)	2,904,000	(0)

Table 3 Results of EDA for 2D Ising.

Samples	Cutoff	Best		Evaluations	
3,000	0.5	–719	(15.68)	746,700	(63,617.69)
6,000	0.3	–714	(18.57)	2,269,800	(277,094.14)
3,000	0.3	–709.6	(8.04)	1,073,400	(130,579.63)
1,000	0.5	–706.6	(12.33)	213,100	(22,997.61)
6,000	0.5	–705.4	(12.84)	1,671,000	(165,043.63)
1,000	0.3	–705	(8.06)	321,600	(38,257.55)
3,000	0.1	–698.6	(15.07)	2,625,900	(261,206.99)
500	0.5	–697	(13.89)	94,800	(6,021.63)
500	0.3	–694.8	(9.39)	151,400	(14,902.68)
500	0.1	–688.6	(17.32)	370,050	(56,346.45)
1,000	0.1	–686	(9.34)	807,500	(118,196.66)
100	0.5	–680.8	(10.59)	14,230	(445.08)
100	0.3	–664.4	(14.31)	22,410	(1,602.78)
6,000	0.1	–649.8	(12.79)	2,904,000	(0)
100	0.1	–632.2	(12.47)	47,430	(2,609.23)

Table 4 Results of CE for Onemax.

Samples	Cutoff	Best		Evaluations	
6,000	0.7	–399.9	(0.3)	538,800	(31,269.15)
6,000	0.5	–394.6	(3.56)	2,835,000	(207,000)
3,000	0.7	–380.1	(8.83)	272,400	(7,800)
3,000	0.5	–359.2	(7.15)	2,901,000	(0)
1,000	0.7	–339.3	(9.18)	72,200	(3,124.1)
500	0.7	–319.5	(4.92)	32,300	(1,661.32)
1,000	0.5	–317.3	(8.96)	279,600	(27,122.68)
500	0.5	–298	(5.59)	74,550	(4,660.74)
100	0.7	–286.3	(4.24)	4,870	(272.21)
100	0.5	–273.9	(4.87)	8,120	(622.58)
500	0.3	–269.1	(7.33)	2,900,500	(0)
1,000	0.3	–265.7	(3.13)	2,901,000	(0)
3,000	0.3	–264.8	(1.83)	2,901,000	(0)
6,000	0.3	–264	(1.55)	2,904,000	(0)
100	0.3	–254.9	(8.35)	2,900,100	(0)

column lists the average cost function value, with the standard deviation in parenthesis, of the best obtained solutions over ten independent runs. The forth column lists the number of function evaluations.

Figures 8 and **9** show the results of HIS for the 1D and 2D Ising models, respectively. In each figure, the horizontal axis represents the number of function evaluations, while the vertical axis represents the average cost function value. Each point represents the average cost function value of the best obtained solu-

Table 5 Results of CE for 1D Ising.

Samples	Cutoff	Best		Evaluations	
1,000	0.7	–288.4	(7.94)	2,901,000	(0)
500	0.7	–287.6	(5.99)	2,613,250	(861,750)
500	0.5	–273.2	(5.31)	2,900,500	(0)
100	0.7	–269.8	(6.72)	183,420	(206,344.9)
500	0.3	–259.2	(6.21)	2,900,500	(0)
100	0.5	–258.2	(8.12)	1,931,100	(1,221,777.89)
1,000	0.3	–251.6	(2.5)	2,901,000	(0)
3,000	0.5	–250.8	(2.56)	2,901,000	(0)
6,000	0.5	–250.2	(2.27)	2,904,000	(0)
1,000	0.5	–250	(2)	2,901,000	(0)
3,000	0.7	–249.6	(2.65)	2,901,000	(0)
6,000	0.3	–249.6	(1.2)	2,904,000	(0)
3,000	0.3	–249.4	(1.8)	2,901,000	(0)
6,000	0.7	–249.2	(1.6)	2,904,000	(0)
100	0.3	–246.6	(8.67)	2,900,100	(0)

Table 6 Results of CE for 2D Ising.

Samples	Cutoff	Best		Evaluations	
1,000	0.7	–533	(12.53)	2,901,000	(0)
500	0.7	–525	(14.81)	2,613,900	(859,800)
500	0.5	–506.6	(13.97)	2,900,500	(0)
100	0.7	–501.2	(9)	168,440	(176,983.23)
500	0.3	–484.2	(16.04)	2,900,500	(0)
100	0.5	–480.4	(10.07)	2,191,050	(937,454.9)
100	0.3	–473.4	(9.3)	2,900,100	(0)
3,000	0.5	–472.8	(2.56)	2,901,000	(0)
3,000	0.7	–472.6	(3.23)	2,901,000	(0)
6,000	0.7	–472.4	(3.56)	2,904,000	(0)
6,000	0.3	–472.4	(3.67)	2,904,000	(0)
3,000	0.3	–472	(3.22)	2,901,000	(0)
1,000	0.5	–471.4	(3.9)	2,901,000	(0)
6,000	0.5	–471.2	(2.99)	2,904,000	(0)
1,000	0.3	–470.8	(2.56)	2,901,000	(0)

Table 7 Results of HIS for Onemax.

Samples	Cutoff	Best		Evaluations	
10	10	–400	(0)	29,155	(12,095.51)
10	20	–400	(0)	32,743	(11,000.43)
50	10	–400	(0)	48,435	(20,868.97)
10	30	–400	(0)	56,170	(16,627.74)
10	40	–400	(0)	67,595	(20,897.39)
50	20	–400	(0)	82,215	(15,277.81)
50	30	–400	(0)	113,680	(23,360.03)
50	40	–400	(0)	157,715	(35,272.9)

tions over ten independent runs for the corresponding number of function evaluations performed. The standard deviations are negligibly small and can be ignored. Additionally, the results of EDA are appended for comparison. The points correspond to the results in Tables 2 or 3.

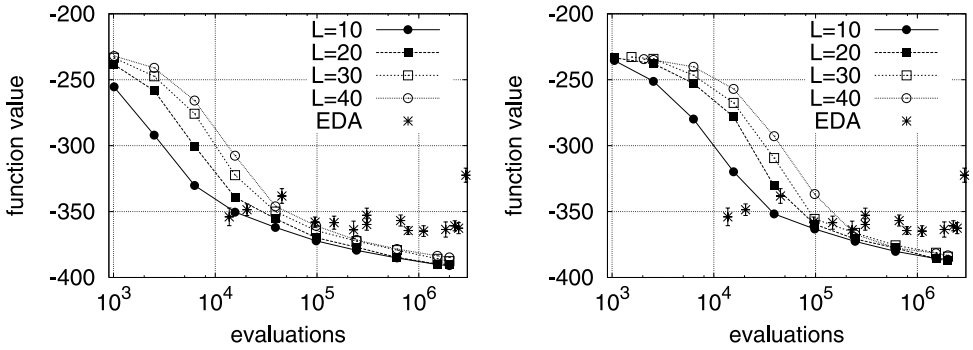
The results for Onemax show that HIS performs as well as EDA. For EDA, M should be set at more than 100; otherwise, EDA can not find the optima. Figures 8 and 9 show that HIS can find better solutions than EDA. EDA may exhibit faster convergence than HIS; however, given sufficient time (i.e., a sufficient number of function evaluations), HIS can find better solutions than EDA.

5. Discussion

5.1 Escaping Local Optima

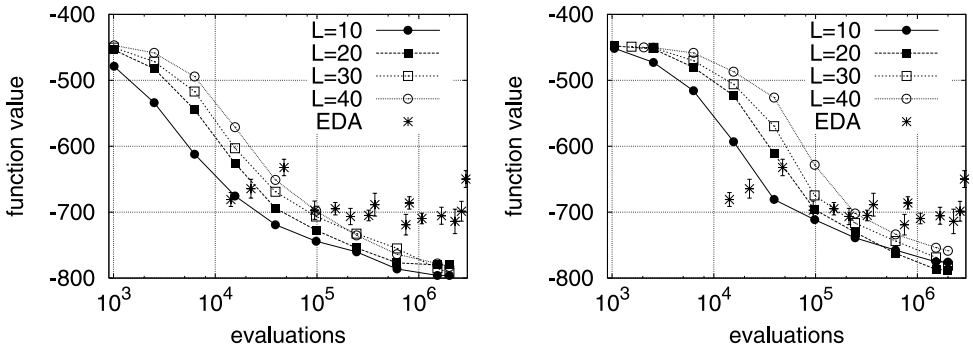
As shown in Figs. 8 and 9, it is clear that HIS can afford better solutions than EDA. The number of samples employed by HIS for building a probability model is given by

$$3 \times M. \tag{24}$$



(a) $M = 10$ (b) $M = 50$

Fig. 8 Results of HIS for 1D Ising.



(a) $M = 10$ (b) $M = 50$

Fig. 9 Results of HIS for 2D Ising.

The number of samples that EDA uses for building a probability model is given by

$$(1 - c) \times M. \tag{25}$$

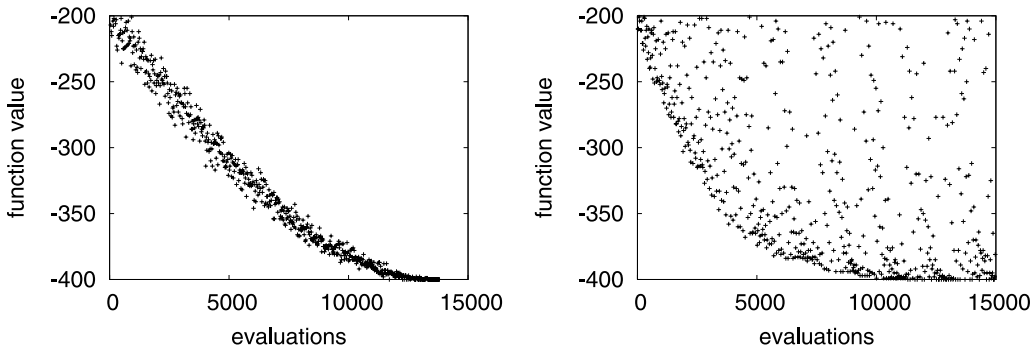
When $M = 10$, HIS uses 30 samples; on the other hand, when $M = 100$ and $c = 0.3$, EDA uses 70 samples. This implies that HIS can escape from local optima by using fewer samples.

In EDA and CE, the entropy of the target distribution is decreased in a stepwise fashion and the target distribution is tracked by a probability model. For tracking the target distribution, the expected log-likelihood must be estimated. The accuracy of an estimator of the expected log-likelihood is dependent on the accuracy of the approximation of the probability model. Thus, once an inferior probability model is built, the accuracy of the estimator of the log-likelihood with respect to the next target distribution is also compromised. And, subsequently, acceptable probability models cannot be generated. This phenomenon can be understood as dropping into local optima.

On the other hand, HIS overcomes this problem by maintaining multiple probability models. In HIS, the larger is the entropy of a target distribution, the easier it is to approximate it. More specifically, low layers tend to have good probability models and high layers tend to have bad probability models. HIS iteratively improves the probability models in the higher layers with samples generated from the lower layers. Thus, if the lower layers have good probability models, the expected log-likelihood can be estimated well at the layers above them. Once a good probability model is built, it tends not to make a change for the worse. Consequently, HIS sequentially improves all the probability models from the lowest layer.

5.2 Iterative EDA

The more the number of function evaluations performed, the better the solutions afforded by HIS. This is because the samples generated by HIS always have certain diversity. **Figure 10** shows the cost function values of the samples generated by HIS and EDA. The horizontal axis represents the number of function evaluations, while the vertical axis represents the cost function value. HIS has no convergence, and



(a) EDA (b) HIS

Fig. 10 Evolution of EDA ($M = 100$, $c = 0.3$) and HIS ($L = 10$, $M = 10$) for 400-dimensional Onemax.

therefore, can find the optimum solution eventually. However, this is not an advantage of HIS because *no convergence* can also be realized by iterative EDA.

As the results of EDA for 1D Ising and 2D Ising show, iterative EDA do not perform as well as HIS because the standard deviations of the best values are insufficiently small. For example, the 10 best obtained solutions in 100 trials of EDA with $M = 3{,}000$ and $c = 0.5$ for the 2D Ising are -746, -736, -732, -732, -730, -730, -730, -728, -726, and -726. Remember that HIS is an extension of iterative CE. The advantage of HIS is the use of the samples and probability models of other trials, whereas each trial in iterative EDA or CE is executed independently.

5.3 Parameters

In sampling-based optimization, there exists a trade-off between the number of function evaluations and the quality of the obtained solutions. In other words, the greater is the number of function evaluations, the better are the solutions afforded. In EDA, the number of function evaluations depends on the parameters: the number of generated samples in one sampling and the cutoff rate. If a solution with a certain quality is needed, it becomes necessary to provide good parameters.

On the other hand, HIS does not converge, and the best obtained value is gradually improved. Thus, it can be said that the setting of the parameters in HIS is easier than in EDA. However, both the number of function evaluations necessary and the efficiency of HIS depend on the number of layers. A greater number of layers in HIS affords greater similarity between adjacent target distributions (i.e., $q_l(x)$ and $q_{l-1}(x)$), implying that it is easier for HIS with a greater number of layers to escape from local optima. On the other hand, HIS with a greater number of layers requires more function evaluations because the samples generated from bad probability models are useless, and the probability models in the higher layers tend to be bad at the early stages. The number of layers may be expected to be determined adaptively according to the accuracy of the probability models: this will be the subject of future work.

5.4 Computational Cost

HIS can provide better results, but at greater computational cost than EDA. First, HIS requires L times the memory space required by EDA: L number of probability models and L number of sample sets maintained in HIS. Second, HIS consumes greater computational time than EDA: the calculation of the probability of the mixture distribution given by Eq. (8) requires considerable time.

5.5 Mixture Model-based EDAs

In terms of using a mixture distribution or multiple populations, some mixture model-based EDAs such as Ref. 9) can be considered similar works. However, they are classified as normal EDAs because they simply split a single population into a number of groups and gradually converge each group, whereas HIS organizes the diversity of all populations. Thus, the optimization process of them is almost equivalent to one illustrated in Fig. 10 (a).

Note that HIS can simply employ a mixture distribution for the probability model of each population. In terms of statistical estimation, the model error can be reduced by using a mixture model. On the other hand, HIS improves

the accuracy of the empirical log-likelihood in terms of importance sampling.

6. Conclusions

This paper proposed Hierarchical Importance Sampling (HIS), a method that can be used instead of the population convergence for evolutionary optimization based on probability models (EOPM). Experimental comparisons between HIS and general EOPM revealed that HIS outperforms general EOPM when applied to problems with local optima. The advantages of HIS can be summarized as follows: (1) it affords better solutions than general EOPM by escaping from local optima, and (2) it allows parameters to be set easily.

Future works are summarized as follows: (1) The target distribution control method should be generalized for other probability distribution families; the Boltzmann distribution, in particular, can be important for applying HIS to continuous function optimization. (2) A population mechanism should be added. The population mechanism of Ref. 3) can calculate the empirical log-likelihood and HIS may therefore be naturally combined with it. (3) The number of the layers should be controlled adaptively. (4) Experimental comparisons involving changing the complexity of the probability models may afford interesting results.

Acknowledgments This work was supported by a Grant-in-Aid for JSPS Fellows (54103) and the 21st Century COE Program "Creation of Agent-Based Social System Sciences."

References

1) Bishop, C.M.: *Pattern Recognition and Machine Learning*, Springer (2006).
2) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Company (1989).
3) Higo, T. and Takadama, K.: Resampling-based Population Mechanism for Evolutionary Algorithms based on Probability Models, *11th Asia-Pacific Workshop on Intelligent and Evolutionary Systems* (2007).
4) Hukushima, K. and Nemoto, K.: Exchange Monte Carlo Method and Application to Spin Glass Simulations, *Journal of the Physical Society of Japan*, Vol.65, No.6, pp.1604–1608 (1996).
5) Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P.: Optimization by Simulated Annealing, *Science*, Vol.220, pp.671–680 (1983).
6) Larrañaga, P. and Lozano, J.A. (eds.): *Estimation of Distribution Algorithm*, Kluwer Academic Publishers (2002).
7) Mahnig, T. and Mühlenbein, H.: A New Adaptive Boltzmann Selection Schedule SDS, *Proc. 2001 Congress on Evolutionary Computation* (2001).
8) Mühlenbein, H. and Paaß, G.: From Recombination of Genes to the Estimation of Distributions I. Binary Parameters, *Parallel Problem Solving from Nature IV*, pp.178–1187 (1996).
9) Pelikan, M. and Goldberg, D.E.: Genetic Algorithms, Clustering, and the Breaking of Symmetry, *Parallel Problem Solving from Nature - PPSN VI 6th International Conference*, pp.840–846 (2000).
10) Pelikan, M. and Goldberg, D.E.: Research on the Bayesian Optimization Algorithm, *Optimization by Building and Using Probabilistic*, Las Vegas, Nevada, USA, pp.216–219 (2000).
11) Pelikan, M. and Goldberg, D.E.: Escaping Hierarchical Traps with Competent Genetic Algorithms, *Proc. Genetic and Evolutionary Computation Conference* (*GECCO-2001*), pp.511–518 (2001).
12) Pelikan, M. and Goldberg, D.E.: Hierarchical BOA Solves Ising Spin Glasses and MAXSAT, *Genetic and Evolutionary Computation Conference 2003* (*GECCO-2003*), pp.1271–1282 (2003).
13) Rubinstein, R.Y.: *Simulation and the Monte Carlo Method*, Wiley-Interscience (1981).
14) Rubinstein, R.Y. and Kroese, D.P.: *The Cross-Entropy Method*, Springer (2004).

Appendix

A.1 Relation between MCMC and EOPM

Calculating the expectation value is common to Markov chain Monte Carlo (MCMC) methods [1] and EOPM. **Table 8** briefly shows the relationship between MCMC and EOPM. The key concepts behind MCMC are local transition, which realizes effective sampling, and designing the transition as a Markov chain by satisfying *detailed balance*, which guarantees mathematical validity. On the other hand, the principle feature of EOPM is estimating an effective sampling distribution and sampling from it. The mathematical validity is guaranteed by importance sampling.

In the practical methods, there exist correspondence relations. For example, simulated annealing (SA) [5] corresponds to general EDA and CE in terms of sequentially tracking a target distribution, and the exchange Monte Carlo

Table 8 MCMC and EDA.

	MCMC	EOPM
Mathematical Validity	Detailed Balance	Importance Sampling
Effective Sampling	Local Transition	Estimated Probability Model
Sequential	SA	EDA, CE
Parallel	EMC	HIS

method (EMC) [4] corresponds to HIS in terms of sampling from multiple target distributions.

A.2 Normalized Importance Sampling

The validity of calculation (7) is confirmed by the following equations:

$$1 = \int \frac{q(x)}{p(x)} p(x) dx \tag{26}$$

$$\simeq \frac{1}{M} \sum_{p(x)} \frac{q(x)}{p(x)} \tag{27}$$

$$= \frac{1}{M} \frac{Z_p}{Z_q} \sum_{p(x)} \frac{\tilde{q}(x)}{\tilde{p}(x)}, \tag{28}$$

$$\frac{1}{\sum_{p(x)} \frac{\tilde{q}(x)}{\tilde{p}(x)}} \simeq \frac{1}{M} \frac{Z_p}{Z_q}, \tag{29}$$

where Z_p and Z_q are the normalizing constants of $\tilde{p}(x)$ and $\tilde{q}(x)$, respectively, $\sum_{p(x)}$ denotes summation over samples generated from $p(x)$, and M is the number of the samples.

A.3 Adjust Threshold $\tilde{f}$

This section describes a method for approximately solving Eq. (16). The estimator of the normalizing constant is given by Eq. (5) by using the samples in X_m, which is generated from a mixture distribution defined by Eq. (8). The estimator of the normalizing constant is a monotonically decreasing step function with respect to $\tilde{f}$, and its change-points are given by $f(x_1) \cdots f(x_M)$, where $x_1 \cdots x_M \in X_m$. Thus, the solution is selected from $f(x_1) \cdots f(x_M)$. Assuming $f(x_1) < \cdots < f(x_M)$, we have the following:

$$\hat{Z}(\tilde{f}(x_{i+1})) = \hat{Z}(\tilde{f}(x_i)) + \frac{1}{M \times p_m(x_{i+1})}. \tag{30}$$

A linear search on $f(x_1) \cdots f(x_M)$ can afford an approximate solution. In the experiments, for $\tilde{f}$, we select $f(x_k)$ such that $|\hat{Z}(f(x_k)) - Z^*|$ is minimized under $Z^* < \hat{Z}(f(x_k))$.

(Received August 8, 2007)
(Revised September 23, 2007)
(Accepted October 27, 2007)

Takayuki Higo received his B.E. and M.E. degrees from Tokyo Institute of Technology, Japan, in 2003 and 2005, respectively. He is now pursuing a doctorate at Tokyo Institute of Technology. He is a member of IEICE, IPSJ, JSAI, JSIAM, and SICE.

Keiki Takadama received his M.E. degree from Kyoto University, Japan, in 1995 and got Doctor of Engineering Degree from the University of Tokyo, Japan, in 1998, respectively. He joined Advanced Telecommunications Research Institute (ATR) International from 1998 to 2002 as a visiting researcher and worked at Tokyo Institute of Technology from 2002 to 2006 as a lecturer. He is currently an associate professor at The University of Electro-Communications. His research interests include multiagent system, distributed artificial intelligence, autonomous system, reinforcement learning, learning classifier system, and emergent computation. He is a member of IEEE and a member of major AI- and informatics-related academic societies in Japan.