Entropy-Boltzmann selection in the genetic algorithms

Article *in* IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics: a publication of the IEEE Systems, Man, and Cybernetics Society · February 2003

DOI: 10.1109/TSMCB.2003.808184 · Source: PubMed

1 author:

Chang-Yong Lee
Kongju National University
53 PUBLICATIONS **889** CITATIONS

elongated box). In more typical situations such as office environments (which offer more room for maneuvering), the hierarchical decomposition should be able to come up with a feasible path quickly. Our intention in this contribution was to provide a mathematical foundation for this approach, rather than to give detailed algorithms; for those, see [1] and [2].

A consequence of all this is that one can treat the tiles of translational freedom as simple shapes, composed from primitives that match the swept area of the prototypical rotational motions. This lifts the placement problem, in a way that is well understood, to a higher level of abstraction. Through the way it was derived, the (faster) solution to the simplified problem should still be useful, to a well understood accuracy, for the original path planning problem which motivated it.

REFERENCES

[1] N. Ahmed, "Motion planning in isothetic workspace," Ph.D. dissertation, Indian Institute Technol., Bombay, Maharashtra, India, 1992.

[2] N. Ahmed and L. Dorst, "Reduction of path planning problems to prototypical motions of prototypical shapes using Minkowski decomposition with applications to the path planning problem for a box in 2-D," Research Institute Comput. Sci., Univ. Amsterdam, The Netherlands, Internal Rep., 1999.

[3] T. Lozano-Perez and M. A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," *Commun. ACM*, vol. 22, no. 10, pp. 560–570, 1979.

Entropy-Boltzmann Selection in the Genetic Algorithms

Chang-Yong Lee

***Abstract*—A new selection method, the entropy-Boltzmann selection, for genetic algorithms (GAs) is proposed. This selection method is based on the entropy and the importance sampling methods in the Monte Carlo simulation. It naturally leads to the *adaptive* fitness in which the fitness function does not stay fixed but varies with the environment. With the selection method, the algorithm can explore as many configurations as possible while exploiting better configurations, consequently helping to solve the premature convergence problem. To test the performance of the selection method, we use the NK-model and compared the performances of the proposed selection scheme with those of the canonical GAs.**

***Index Terms*—Entropy-Boltzmann selection, genetic algorithms (GAs), NK-model, premature convergence problem.**

I. INTRODUCTION

Optimization algorithms based on the biological evolution and the natural selection have been widely studied under the name of evolutionary computations. Among them, genetic algorithms (GAs) [1], genetic programming (GP) [2], evolutionary programming (EP) [3], and evolutionary strategies (ESs) [4] are the current mainstream of the investigation. Although there are somewhat different viewpoints as to what are exactly being evolved, they have many aspects of the idea of the evolution and the natural selection in common.

As is true for all known optimization algorithms, it is difficult for the evolutionary algorithms to find the true optimum value of the complex systems having many local optima. GAs are not an exception. In the canonical GAs, the schema theorem asserts that the proportions of the better schema to the overall population increases as the generation progresses and eventually the search converges to the "best" solution with respect to the optimization function. In implementations of the algorithm, however, this convergence often occurs rather *prematurely* before the optimal solution is found.

Attempts have been made in order to overcome this premature convergence. One way to handle the premature convergence is to increase the size of the population. This, however, can be done only moderately since it eventually makes the search too inefficient. Another way is to use the mutation operation that changes the value of each allele according to some probability. The mutation alone, sometimes, is not sufficient. On one hand, when the mutation probability is low, mutated configurations do not differ enough from the original configurations to get out of a local minimum. On the other hand, when the mutation probability is high enough to escape the local minimum, it is apt to yield a random search that is far from the optimization.

The premature convergence or a similar phenomenon is ubiquitous in all optimization algorithms and manifests itself in the case where there are high barriers between local optima in the fitness landscape. This results from the premature loss of diversity in the population and, as a consequence, it is hard to search as many configurations as possible. Thus, in order to avoid premature convergence as much as possible, it is important to maintain diversity inside the population while emphasizing the exploitation of higher fitness individuals. In most optimization problems, by and large, it is necessary to maintain two basic strategies: "exploring" robustly as many configurations as possible while "exploiting" better configurations to the given population.

In this paper, we propose the *entropy-Boltzmann selection* method to partially overcome the premature convergence problem. This selection method is a combination of the entropy sampling [5] and the importance sampling [6] methods in the Monte Carlo simulations. The entropy sampling directs the evolution to select configurations of lower entropy. In this way, the rate of rare configurations to be selected is higher than that of abundant ones. As a result, when a configuration falls into a local minimum and many of the same or similar configurations are piled up, the rate of acceptance of configurations concentrated near the local minimum is highly suppressed, which enables the system to escape easily from the local minimum. The importance sampling, on the other hand, is based on the Boltzmann factor, which gives higher selection probability for the lower energy (or higher fitness).

Therefore, by utilizing these two sampling methods simultaneously in the acceptance of offspring, one is able to not only at least partly overcome the high barriers between local minima, but also continue to search for global or nearly global minimum. Contributions have been made to emphasize the effect of the entropy sampling in GAs [7], [8] and simulated annealing (SA) [9]. There is a previous study on an implementation of the entropy and Boltzmann selection to the traveling salesman problem (TSP) [10]. In [10], however, methods of selecting offspring and entropy estimation were quite different from those of the current study. In particular, the Metropolis algorithm was not adequately applied, and the application of the Boltzmann factor had different origin. There are other studies related to the current study. Among them, the study on the self-organizing behaviors exhibited in GA in terms of entropy [11] and Boltzmann reduction operation in GA [12] are worth mentioning.

Manuscript received September 29, 2000; revised March 22, 2001 and January 21, 2001. This work was supported by Korea Research Foundation Grant KRF-99-003-E00375. This paper was recommended by Associate Editor W. Pedrycz.

The author is with the Department of Industrial Information, Kongju National University, Yesan, Chungnam 340-800, South Korea (e-mail: clee@kongju.ac.kr).

Digital Object Identifier 10.1109/TSMCB.2003.808184

To test the performance and the advantages of the proposed selection method, the NK-model [13], a generalization of the spin-glass model, was chosen with a tunable ruggedness parameter K. We demonstrate that the genetic search under the guidance of the entropy-Boltzmann selection yields better results than that of the conventional selection schemes.

II. Entropy-Boltzmann Selection

Even though there are many selection schemes that have been suggested and used in the GA literature [14], they all are based on the fitness of the individuals: the better fitness an individual has, the higher its probability of being selected in the reproduction process. It is also important to note that the selection scheme in conventional GAs just plays a role of selection pressure and the successively chosen individuals are independent among others. As a result, the fitness values of the individuals do not form any probability distribution.

The entropy-Boltzmann selection method is based on the entropy and the importance sampling methods, both of which in turn stem from Monte Carlo simulation. Importance sampling comes in if one chooses $P_B(x)$, the sampling probability of having configuration x of energy $E(x)$, to be inversely proportional to the Boltzmann factor $\beta E(x)$

$$P_B(x) = Ce^{-\beta E(x)} \tag{1}$$

where C is the normalization factor given by $C = \sum_x e^{-\beta E(x)}$ and $\beta = (kT)^{-1}$ with the inverse of temperature T and the Boltzmann constant k.

This importance sampling is generally a good method, but fails to satisfy the ergodicity if there exist many high barriers in the energy landscape. This is closely related to the premature convergence problem in GAs. One way to overcome, at least for a large part, this difficulty is to relax the Boltzmann factor. By replacing βE in (1) by some arbitrary function $J(E)$, we are free to get any probability distribution by setting $J(E)$ accordingly. In particular, if we set $J(E) = S(E)$, the entropy of the system, then we would obtain the entropy sampling. The entropy $S(E)$ of the system having the energy E is defined as $S(E) = k \ln \Omega(E)$, where $\Omega(E)$ is the number of configurations with the energy E, and the Boltzmann constant k will be set to $k = 1$ for convenience.

This sampling method is designed to access all the energy space with an equal probability, thus it is, in general, a good method for Monte Carlo simulation in which all energy state are needed to be equally visited and explored. The entropy sampling was applied to the SA to solve TSP [9], and they obtained improved results over those using the conventional SA. In addition to this, when the entropy sampling was implemented in GAs, it was shown that entropy sampling helped to maintain the population diversity as well as providing a systematic way to escape from the local optima [7], [8].

The entropy sampling alone, however, is not suitable for the optimization problems since the entropy sampling results in just a random walk in the energy space. Thus, in order to exploit better individuals while maintaining the population diversity, it is necessary to have some selection pressure according to the fitness. To this end, we adopt a new selection scheme, the entropy-Boltzmann selection, which is a combination of entropy sampling and importance sampling. More specifically, the entropy-Boltzmann selection is realized by setting $J(E) = S(E) + \beta E$, so that the probability of occurrence of a configuration x with energy E has the form

$$P_{E\&B}(x) \propto e^{-\{S(E)+\beta E(x)\}}. \tag{2}$$

In order to get a desired probability distribution for the case of the entropy-Boltzmann sampling, we need to have some sampling procedure. One general way to produce configurations of a desired probability distribution is known as the Metropolis algorithm [15]

$$r \equiv \frac{W(x \to x')}{W(x' \to x)} = \frac{P(x)}{P(x')} = e^{\Delta F} \tag{3}$$

where $\Delta F = [S(E(x')) + \beta E(x')] - [S(E(x)) + \beta E(x)]$.

III. Estimation of the Entropy and Implication of the Entropy-Boltzmann Selection

To implement the entropy-Boltzmann selection for GAs, we need to know the entropy $S(E)$ of the system as a function of energy. In general, the entropy of a system is not known *a priori*. To find the exact entropy is in fact equivalent to solve the problem in the first place. Therefore, we have to estimate the entropy. There are a few ways to estimate the entropy of a given system [5], [16]. These methods start with an incorrect test entropy and recursively updates the entropy so that, in the large sampling limit, a correct entropy may be obtained.

Since the primary reason that the entropy is introduced in the selection method is to prevent the premature convergence problem systematically, the entropy should be used as a measure of the occurrence frequency for each configuration. If a candidate configuration belongs to a group of configurations that have previously been selected more frequently, the configuration would be less probable to be selected. From the above consideration and the general properties of the entropy, we perform the entropy estimation as follows. [16]. First, we discretize the range of energy $E(x)$ into N segments each of which is labeled μ, e.g., $\mu = 0, 1, \ldots, N-1$, so that N is the total number of energy levels. Furthermore, let $\hat{S}(\mu, t)$ be the estimated entropy of the energy state with label μ at "time" t or the tth trial configuration. With the above convention, the update of the value of entropy takes the form of

$$\hat{S}(\mu, t+1) = \hat{S}(\mu, t) + \epsilon(t)\delta_{\mu, m(x)} \tag{4}$$

where $\epsilon(t)$ is a predetermined positive function that has to be sufficiently small in the large "time" limit. $\epsilon(t)$ can be, in general, a function of "time," but we set it independent of time for simplicity. We discretize each value of energy into N equal interval and let $m(x)$ be the label of discretized energy of the configuration x. Furthermore, one can also set the initial entropy to zero, namely, $S(\mu, t = 0) = 0$, for all $\mu = 0, 1, \ldots, N-1$.

This way of estimating the entropy requires the proper choice of the parameter ϵ. When one takes ϵ too small, one effectively samples according to the random walk, not in the energy space but in the configuration space since practically almost all trial configurations will be accepted. Contrary, if one takes ϵ to large, e.g., of order one or higher, the difference of the entropy will be also large; as a result, the transition probability for the trial configuration becomes essentially zero or one, not in between. Proper values of ϵ should be chosen between $10^{-3} \leq \epsilon \leq 10^{-2}$ [16].

Even though a proper choice of the parameter β depends on the problem in which we are interested, there may be some rule-of-thumb between β and ϵ in order to take both the entropy and energy into account. One heuristic method, for example, is setting β such that βE and S are approximately of the same order. In this way, the entropy-Boltzmann selection scheme can be thought of as an interplay between the entropy and the energy of the system.

The entropy-Boltzmann selection can be characterized as follows. First, it maintains a balance between the "exploration" and the "exploitation." With the entropy-Boltzmann selection, we are able to not only search the whole energy space with an equal probability by utilizing the entropy sampling, but also maintain fitter individuals by applying the Boltzmann factor. Second, the entropy-Boltzmann selection scheme can be regarded as a selection with an *adaptive fitness*, i.e.,

varying fitness function in the course of the evolution. That is, the adaptive fitness consists of the usual fitness together with the entropy measure, which may vary from generation to generation. In this way, one has an adaptive fitness function which varies according to the environment.

IV. Genetic Algorithms With the Entropy-Boltzmann Selection

In practice, GAs with the entropy-Boltzmann selection consist of the following steps.

Step 1) Prepare M configurations $P_1, P_2, \ldots, P_M$ to form the initial population and set the initial entropy $S(\mu, t=0)=0$ for all $\mu = 0, 1, \ldots, N-1$. Choose randomly an initial test configuration x from the initial population and calculate the corresponding fitness $E(x)$.

Step 2) To obtain the $(k+1)$th trial offspring x' represented by C_{k+1}, $k = 0, 1, \ldots, M-1$, select parents (P_i, P_j) randomly from the population and apply the crossover and mutation operators. A x' is chosen randomly out of two offspring. Calculate the fitness $E(x')$ of the trial configuration x' and find its corresponding entropy $S(\mu, t)$. To accept x' as a member of the next generation, apply the detailed balance condition. This can be done by evaluating the ratio r from (3). If $r \geq 1$, then the trial configuration x' is accepted, otherwise x' is accepted with the probability r. If the configuration x' is accepted, then the trial configuration x' becomes test configuration x and update the entropy according to (4). If rejected, select parents from the population randomly and repeat the above procedure until a trial configuration is accepted.

Step 3) Repeat Step 2 until the number of accepted configurations is the same as that of the population. Now, the set of accepted configurations $(C_1, C_2, \ldots, C_M)$ becomes the new population $(P_1, P_2, \ldots, P_M)$ for the next generation.

Step 4) Repeat Steps 2 and 3 for a desired number of generations.

The main differences between conventional GAs and this algorithm are twofold: First, in the conventional GA, the better fitness a chromosome has, the higher its probability of being selected in the reproduction process. This implies that the chromosome of better fitness function *always* has higher probability to survive. In the proposed selection method; however, it is the *adaptive* fitness that is the measure of the survival for the next generation. Thus, even if a chromosome has a lower fitness in the usual sense, it could have higher probability to be selected provided it has a sufficiently lower entropy. Second, in conventional GAs, competitions are made in the course of the reproduction, that is, competitions occur in the selection procedure among the parents. Once the reproduction is carried out, the offspring are produced in parallel and there is no direct correlation among offspring except through their parents. Whereas in the proposed selection scheme, parents are selected randomly and the competitions are made in the course of the acceptance of the new configurations. That is, each produced offspring is tested (to be accepted or rejected) against the previously accepted offspring through the Metropolis algorithm. Thus there is probabilistic competitions between the subsequent offspring. In this way, the next generation consists of samples of the desired probability distribution.

V. Test of the Entropy-Boltzmann Selection Using NK-Model

We test the entropy-Boltzmann selection in GAs against the NK-model. The NK-model was created by Kauffman [13] and can be thought of as a generalization of spin-glass models. The NK-model defines a class of parameterizable fitness landscape whose

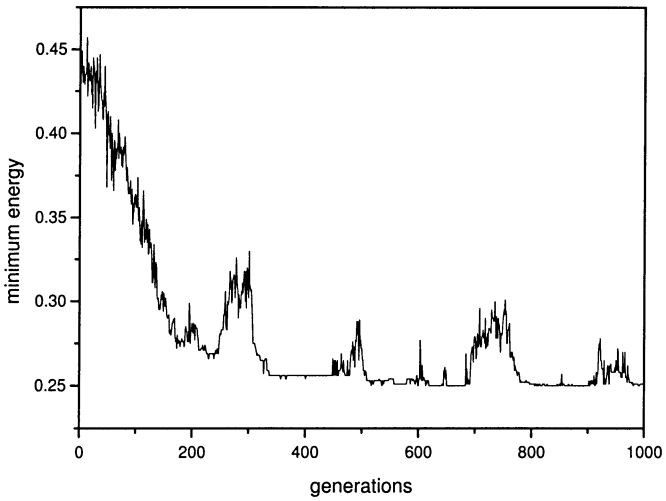

Fig. 1. Minimum energies as a function of generations are plotted up to 1000 generations. The unit of the energy is dimensionless and the other parameters used are $\boldsymbol{N = 100}$, $\boldsymbol{K = 6}$, $\boldsymbol{\beta = 0.17}$, and $\boldsymbol{\epsilon = 0.002}$ with population size 50.

search space is composed of all possible configurations of N loci, denoted by $\mathbf{s} = \{s_i\}$, $i = 1, \ldots, N$. It is a simple class of landscapes the ruggedness of which can be controlled by the parameter K, the degree of epistasis. K is the number of other loci on which the fitness contribution of each locus i depends. For each locus i, the K other loci can be chosen either randomly or according to some topology. If we restrict possible values of each locus s_i to be zero and one, then the size of the search space is 2^N. If we denote $\vec{n_i} = (n_1, n_2, \ldots, n_K)$ as the set of the K loci on which the fitness of the locus i depends, then the interaction space $(s_i, \vec{n_i})$ consists of 2^{K+1} possible configurations of $K+1$ alleles and assign a randomly chosen fitness E_i between zero and one to each such configuration. The fitness of the each chromosome is then defined as the average of the fitness for each locus

$$E(\mathbf{s}) = \frac{1}{N}\sum_{i=1}^{N} E_i(\mathbf{s}). \tag{5}$$

There are two major reasons to choose NK-model to demonstrate the effectiveness of the entropy-Boltzmann selection. First, this model provides a natural means of the coding, that is, it is free of the representation problem and there is no ambiguity of the definition of the fitness. With this model, one can test the proposed selection scheme without having dependence of the coding and the ordering problems. Second, the ruggedness can be controlled. That is, by adjusting K, one can tune the ruggedness of the fitness landscape of the model so that one can have a tunable degree of the local minima. $K = 0$, for example, yields all loci are independent among others and there is no ruggedness in the fitness landscape. As K increases, each locus is more dependent on other loci, which, in turn, give more ruggedness in the fitness landscape [17].

We apply the entropy-Boltzmann selection scheme to find the minimum energy (or maximum fitness) of the NK system. Fig. 1 shows typical behavior of the minimum energy as a function of generations. As seen in Fig. 1, the energy reaches a local minimum and stays near it until we reach approximately the 450th generation. During the period when the system stays near the local minimum, the histograms in the vicinity of the local minimum pile up; as a result, the corresponding entropy become larger than that of higher energy. This makes configurations of the higher energy have lower entropy and, subsequently, have higher probabilities to be accepted. The selection of the offspring tends toward the higher energy state which is manifested as the first bump in

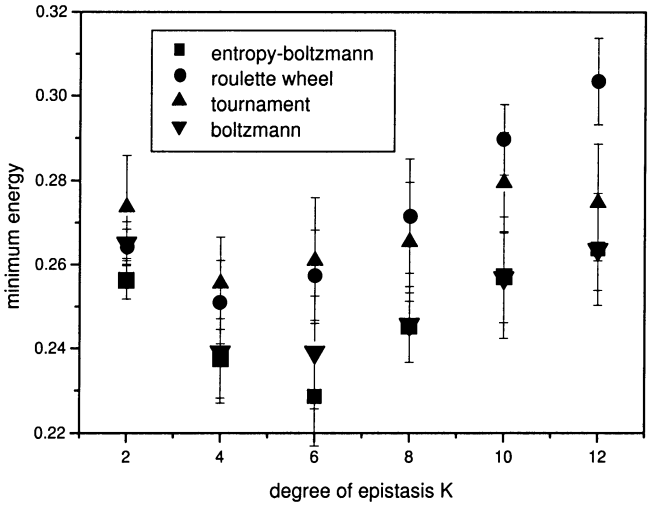

Fig. 2. Result of a simulation of finding the minimum energy of the NK-model with various values of K. The entropy-Boltzmann selection is tested against the tournament, roulette-wheel, and Boltzmann selections. For each K, 20 independent trials are averaged for all selection schemes and the error bars are corresponding standard deviations.

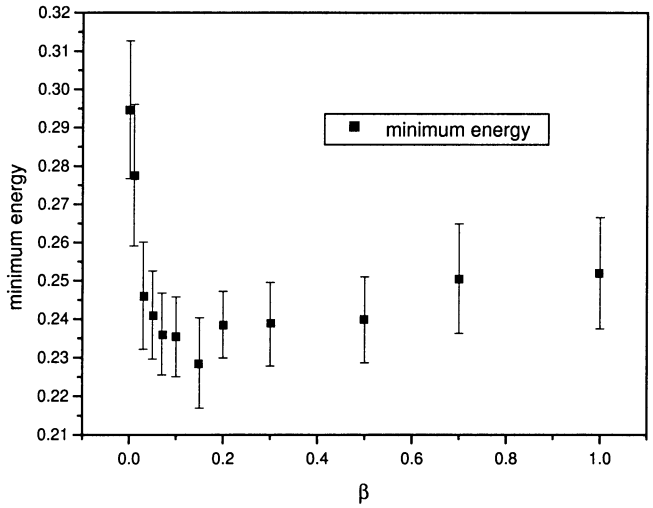

Fig. 3. Result of a simulation of the finding the minimum energy of the NK-model with various values of β holding $\epsilon = 0.001$ fixed after 1000 generations. For each value of β, 20 independent trials are averaged and the error bars are corresponding standard deviations. The other parameters are $N = 100$, $K = 6$, and with population size 50.

TABLE I
THE COMPARISON OF COMPUTATIONAL TIME, IN THE UNIT OF SECOND, OF THE ENTROPY-BOLTZMANN SELECTION WITH TOURNAMENT SELECTION FOR EACH $K = 2, 4, 6, 8, 10,$ AND 12. TEN INDEPENDENT SIMULATIONS WERE CARRIED OUT FOR EACH K, AND THE AVERAGE VALUES ARE LISTED. THE OTHER PARAMETERS ARE $N = 100$ WITH POPULATION SIZE 50

	K=2	K=4	K=6	K=8	K=10	K=12
Tour.	7.01	7.43	7.61	6.96	7.34	7.35
E&B	20.29	22.49	22.12	20.24	26.44	21.92

TABLE II
MULTIMODAL FUNCTIONS THAT HAVE BEEN TESTED IN THIS PAPER. N STANDS FOR THE DIMENSION OF THE FUNCTION SETTING $N = 30$, AND S THEIR VARIABLE RANGES

Test Functions	S
$f_1(x) = \sum_{i=1}^{N}\{x_i^2 - 10\cos(2\pi x_i) + 10\}$	$(-5.12, 5.12)$
$f_2(x) = -20exp\{-0.2\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}\}$ $-exp\{\frac{1}{N}\sum_{i=1}^{N}\cos(2\pi x_i)\} + 20 + e$	$(-32, 32)$
$f_3(x) = \frac{1}{4000}\sum_{i=1}^{N} x_i^2 - \Pi_{i=1}^{N}\cos(\frac{x_i}{\sqrt{i+1}})$	$(-600, 600)$

Fig. 1. After producing configurations of higher energy for some generations, the entropy in the higher energy area become large and the updated entropy forces the system to produce the offsprings of lower energy. In this way, the population can get out of the local minimum and continue to search the global minimum. As the system repeats this process, there is a better chance for the system to reach lower energy than previous ones.

We also test the proposed entropy-Boltzmann selection against other selection methods with various K. The K other loci for each locus are chosen randomly. The parameters that we have used for various selection methods are the following: 1) number of the population: 50; 2) crossover operator: 1 point crossover; 3) mutation probability: 0.01; 4) tournament size 5, for the tournament selection; 5) $\epsilon = 0.001$ and $\beta = 0.15$, for the entropy-Boltzmann selection; and 6) number of generation: 1000 generations.

In Fig. 2, we compare the performance of the entropy-Boltzmann selection with the conventional tournament, roulette-wheel, and Boltzmann selections. We especially plot the minimum energy of the NK-model obtained from the various selection methods as a function of the epistasis K. As can be seen in Fig. 2, in general, the entropy-Boltzmann selection yields better results than the conventional selection schemes. Note here that, for simplicity, we fixed the parameter ϵ and β for all K. However, the best choice for the parameters may be different for each K and a better choice of parameters might yield a better performance.

The entropy-Boltzmann selection, however, is computationally more expensive than usual selection methods in conventional GAs. This is mainly due to the usage of the Metropolis algorithm for selecting offspring. In Table I, we compare computational time for the entropy-Boltzmann selection with that of the tournament selection. From Table I, one can infer that the entropy-Boltzmann selection takes, in terms of computing time, about three times as much as the tournament selection.

In the entropy-Boltzmann selection, the important parameters are ϵ and β which are closely related. As we have mentioned in Section III, the best choice of the parameter β may depend on the choice of ϵ. To investigate this, we carry out two different sets of the experiments: one with varying β for a fixed value of ϵ, another with both β and ϵ vary.

We first test the performance of the selection scheme with various values of β while keeping ϵ constant. A typical result is shown in Fig. 3. When $\beta \to 0$, the entropy-Boltzmann selection reduces to a selection using the entropy only; as a result, it becomes a random walk in the energy space. When $\beta \to 1$, the ratio of ϵ to β becomes $\epsilon/\beta \ll 1$ and the selection scheme virtually reduces to a selection without entropy. The best results are obtained when β is in between the two extremes. As an example, the best simulation result was obtained when the value of $\beta = 0.15$ for a fixed $\epsilon = 0.001$. This suggests that one should take both the entropy and the Boltzmann factor simultaneously into account to get better results.

In order to test the entropy-Boltzmann selection further, we applied this method together with various selection schemes to the minimization of multimodal functions with many local minima. Table II lists test functions and their ranges of variables. These are benchmarking functions commonly used in the evolutionary computation experiments [18]. In applying the GAs to these functions, we used the floating point implementation [19] of the chromosome and maintained usual genetic operations. For continuous functional optimization, it is generally known that EP or ESs work better than the GAs. This is, however, not critical in this case as we test the behavior of different selection

TABLE III
THE MINIMUM VALUES FOR EACH FUNCTION LISTED IN TABLE II FOR VARIOUS SELECTION METHODS AFTER 500TH GENERATION. THE VALUES ARE AVERAGED OVER 50 INDEPENDENT RUNS AND THE CORRESPONDING STANDARD DEVIATIONS ARE IN THE PARENTHESES

Selection	f_1	f_2	f_3
Tournament	245.4(29.6)	18.6(0.6)	227.1(48.5)
Roulette wheel	229.9(33.8)	18.1(0.9)	181.9(53.3)
Boltzmann	236.1(39.7)	18.4(0.6)	208.5(55.8)
E&B	213.3(25.7)	18.8(0.9)	116.1(31.6)

schemes within the GAs. Table III shows results for the minimization of these functions. From the table we can see that the entropy-Boltzmann selection performs better than or comparable to other selection methods for these three functions, but not statistically significantly so.

VI. SUMMARY AND CONCLUSION

We proposed the entropy-Boltzmann selection method based on the entropy and importance sampling. While keeping the merits of the conventional GAs, this selection method utilized the idea of the entropy and importance sampling in the Monte Carlo simulation. Using this selection scheme, in particular, one can balance between the two seemingly conflicting strategies in the optimization algorithms: explorations and exploitations.

In the conventional GAs, configurations of higher fitness always have higher probability to be selected, thus it is hard to escape from a local optimum once the algorithm falls in it. The proposed selection scheme tends to overcome these shortcomings by utilizing the entropy and Boltzmann selection scheme.

Within the framework of this selection method, one can naturally leads to an adaptive fitness: the adaptive fitness consists of the usual fitness and entropy change due to the environment. This viewpoint is further supported by the observation that the evolution can be carried out through the adaptation of the individuals to their environment.

We applied this selection scheme particularly to the minimization of the energy in the NK-model and multimodal functions with many local optima, and compare the performances of the new selection method with that of the conventional methods. We obtained better performance than the conventional methods at least for the given conditions. It should be clear that the proposed selection method is just one of the many alternatives to overcome the premature convergence, and obviously cannot be the best selection method for *all* optimization problems.

Since the entropy-Boltzmann selection is a general selection scheme, it can be readily applied to other optimization algorithms, such as EP and EPs, not to mention the SA. The global convergence property of the algorithms and practical applications to other benchmarking optimization problems can be the direction of further works.

REFERENCES

[1] D. Goldberg, *Genetic Algorithm in Search, Optimization and Machine Learning*. Reading, MA: Addison-Wesley, 1989.
[2] J. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA: MIT Press, 1992.
[3] D. Fogel, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. Piscataway, NJ: IEEE Press, 1995.
[4] T. Bäck and H.-P. Schwefel, "An overview of evolutionary algorithms for parameter optimization," *Evol. Comput.*, vol. 1, no. 1, pp. 1–23, 1993.
[5] J. Lee, "New Monte Carlo algorithm: Entropic sampling," *Phys. Rev. Lett.*, vol. 71, pp. 211–214, 1993.
[6] K. Binder and D. Heermann, *Monte Carlo Simulation in Statistical Physics*. New York: Springer-Verlag, 1988.
[7] C.-Y. Lee and S. Han, "Genetic-entropic algorithm: An application to NK-model and the statistical analysis," in *Proc. 7th Int. Conf. Genetic Algorithms*, 1997, pp. 254–259.
[8] ——, "Evolutionary optimization algorithm by entropic sampling," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 57, no. 3-B, pp. 3611–3617, 1998.
[9] J. Lee and M. Y. Choi, "Optimization by multicanonical annealing and the traveling salesman problem," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 50, pp. R651–R654, 1994.
[10] K. K. Lee and S. K. Han, "Entropic and Boltzmann selection in the genetic algorithm and the traveling salesman problem," in *Proc. Int. Conf. Neural Information Processing*, 1994, pp. 162–167.
[11] L. Qu and R. Sun, "A synergetic approach to genetic algorithms for solving traveling salesman problem," *Inf. Sci.*, vol. 117, pp. 267–283, 1999.
[12] J. Lozano, P. Larrañaga, M. Granña, and F. Albizuri, "Genetic algorithms: Bridging the convergence gap," *Theor. Comput. Sci.*, vol. 229, pp. 11–22, 1999.
[13] S. Kauffman, "Adaptation on rugged fitness landscapes," in *Lectures in the Science of Complexity*, D. Stein, Ed. Reading, MA: Addison-Wesley, 1989, vol. 1, Santa Fe Institute Studies in the Science of Complexity, Lect., pp. 527–618.
[14] M. Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, MA: MIT Press, 1999.
[15] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equation of state calculations for fast computing landscape," *J. Chem. Phys.*, vol. 2, pp. 1087–1093, 1953.
[16] M. Thill. (1997) On the free energy Monte Carlo algorithm. [Online]. Available: http://arXiv:cond-matt/9 703 234
[17] S. Kauffman, *At Home in the Universe*. London, U.K.: Oxford Univ. Press, 1995, For the ruggedness of the K-dependence, p. 175.
[18] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Trans. Evol. Comput.*, vol. 3, no. 2, pp. 82–102, 1999.
[19] Z. Michalewicz, *Genetic Algorithms + Data Structure = Evolution Programs*. New York: Springer-Verlag, 1992.