JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 17, 47-71 (2001)

Short Paper

A Theoretical Aspect of a Stochastic Sketching Method for Global Optimization

JORNG-TZONG HORNG*, YING-PING CHEN+ AND CHENG-YAN KAO++
*Department of Computer Science and Information Engineering
National Central University
Chungli, Taoyuan, Taiwan 320, R.O.C.
E-mail: horng@db.csie.ncu.edu.tw
+Department of Computer Science and Information Engineering
University of Illinois
Urbana-Champaign, Illinois, U.S.A
E-mail: ypchen@csie.ntu.edu.tw.
++Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan 106, R.O.C.
E-mail: cykao@csie.ntu.edu.tw

In this paper, we propose *Stochastic Sketching* method for global optimization based on the simulation of human behavior. Stochastic Sketching models the thought process and strategies of human beings and applying the artificial model to problems. We introduce and discuss concepts and components essential to Stochastic Sketching in detail, including the sampling guide, zooming controller, sketching model, precision threshold, and satisfaction probability. The mathematical foundations of Stochastic Sketching are discussed and a preliminary theoretical base is presented.

Keywords: unconstrained optimization, global optimization, Pincus theorem, evolutionary computation, evolution strategies, genetic algorithms

1. INTRODUCTION

Many practical applications, such as computer-aided design and construction, biological, chemical, and electrical, medical engineering, and production planning, can be formulated as global optimization problems whose objective functions often possess many local minima in the region of interest. Usually, it is desired to find the minimum among all the local minima of the objective function at which the objective function takes its lowest value, i.e., the *global minimum*.

Received March 3, 1998; revised December 7, 1998, & October 18, 1999; accepted April 20, 2000.
Communicated by Jieh Hsiang.

Once a practical application is formulated as an objective function, only certain combinations of values of the variables are possible because the coordinate system is a mapping of the parameters in the application, and the values are determined by the considered system features. This means that all possible solutions are restricted to a subregion $A \subseteq \mathbf{R}^n$[1], which is called the *feasible region*. Moreover, objective functions in practical applications may be multimodal; i.e., they may have several local minima in their feasible regions. The problem of determining not just a *local minimum*, but also the *smallest local minimum* of the objective function in the domain A is known as the *global optimization problem.*

Much research has focused on solving global optimization problems because optimization is one of the most important fields in both science and engineering. We briefly introduce several methods which are commonly adopted to solve global optimization problems in the following paragraphs.

Several traditional methods applied to global optimization are described first. In order to guarantee that the predescribed accuracy is achieved, the essential *Lipschitz constant* is needed for covering methods ([1], p. 26). However, although Lipschitz constants do exist for continuous functions defined on compact sets, they are usually not available in practical problems. Even worse, if the objective function is given by a computer procedure, then estimating the Lipschitz constant itself introduces a difficult task. Generalizing covering methods to deal with multidimensional objective functions is hard because two immediate problems are encountered. First, the combinatorial explosion of the number of sampling points may be fatal when n grows to certain amount. Second, the Lipschitz constant depends on the scaling of the variable, and the algorithm may be inefficient if the scaling is not proper ([1], p. 30). Numerous theoretical results for covering methods have been published, but various problems still remain unsolved. Applying covering methods successfully and efficiently to practical problems requires good estimation of the bounds of derivatives of objective functions because covering methods are generally very sensitive to the estimator of the Lipschitz constant.

The tunneling methods LMGC:TunnelingMethod, GL:TunnelingMethod, and LM: TunnelingMethod consist of two phases: local minimizing and *tunneling*. Levy and Montalvo [2] proposed theoretical generalization of the tunneling method to obtain the multidimensional objective functions, and experimental results were also presented.

Clustering methods employ cluster analysis techniques to solve global optimization problems. There are three main, common components of clustering methods: a sampling method, a grouping technique, and a local optimizer. A general clustering algorithm can be found in ([1], p. 112). Mainly because clustering methods combine global and local search in an efficient manner, they have succeeded in solving global optimization problems. Additionally, since many studies on clustering methods have been done, they rest on a sound theoretical base and exploit their talents on practical problems.

Some global optimization methods are based on the construction of a statistical model of the global optimization problem. The assumptions made in such a method are usually that the objective function is very expensive to evaluate, and that it is unsuitable to apply the method based on a statistical model to solve cheap objective functions.

[1] The symbols used in the paper are listed in the Appendix.

Methods based on statistical models of objective functions may be used only for global search because the function would be better depicted by a polynomial than by a statistical model in certain kinds of intervals. Consequently, a quadratic model was incorporated by Žilinskas [1] in 1976 as the local model into the global statistical model to depict the objective function. The algorithms are constructed so as to perform efficiently under average conditions of practical optimization problems in engineering based on the Bayesian philosophy of rationality, and have been successfully applied to difficult practical problems.

Unlike traditional methods, Evolutionary Algorithms try to model concepts captured from organic evolution and to benefit from implementing those ideas. Simulated Annealing assumes an analogy between a physical many-particle system and a combinatorial optimization problem and tries to solve combinatorial optimization problems by simulating the annealing process of a solid in a heat bath. We may say that both Evolutionary Algorithms and Simulated Annealing look at nature and solve difficult problems based on inspiration from the mysterious complex system.

Evolutionary Algorithms are mainly based on models of biological evolution. The corporate learning and searching process is modeled within a *population* of individuals, each of which not only represents a sampling point in the solution space, but also may be a temporal container of current obtained knowledge about the environment [3]. Evolution Strategies were first applied to hydrodynamical problems which may be impossible to solve analytically, and the primitive method was shown to be practical for some of those problems.

Evolutionary Programming, which focused on automatic programming, was introduced by L. J. Fogel [4] in the 1960's. Rudimentary Evolutionary Programming applied uniform random mutation to the underlying alphabet, and the $(\mu + \lambda)$-selection scheme was used as its selection operator. In the late 1980's, D. B. Fogel applied Evolutionary Programming to optimization problems whose objective functions were continuous variables. Evolutionary Programming is analogous to Evolutionary Strategies, but they developed independently of each other. Researchers in the fields of Evolutionary Programming and Evolution Strategies did not establish contacts until 1992.

Genetic Algorithms [5] have been extensively applied in many kinds of applications [6], such as game-playing [7], biological cell simulation [8, 9], and pattern recognition [10] (see also [6], pp. 95-97). Additionally Genetic Algorithms are also applied broadly to function optimization problems [11-23] in both theory and practice. *Simulated Annealing* was first introduced by Kirkpatrick, Gelatt, and Vecchi [24] and independently by Černy [25] in the early 1980's. The concepts of simulated annealing in combinatorial optimization are based on a strong analogy between the physical annealing process of solids and the problem of solving large combinatorial optimization problems. Kirkpatrick, Gelatt, and Vecchi successfully applied Simulated Annealing to several important combinatorial problems, such as the chip design problem and the traveling salesman problem [24]. Sasaki and Hajek derived some theoretical results on time complexity [26], and Faigle and Schrader considered convergence of Simulated Annealing algorithms to the global minimum with probability one [27].

Generalizations of Simulated Annealing have been used to solve optimization problems with objective functions of continuous variables. Aluffi-Pentini, Parisi, and Zirilli showed that with Simulated Annealing, a rather large number of function evaluations is

needed to find a solution [28]. Corana, Marchesi, Martini, and Ridella presented a method for optimization of functions of continuous variables based on Simulated Annealing [29]. Although the method can find the global minimum of test functions with many local minima, it needs a lot of function evaluations. Corana, Marchesi, Martini, and Ridella also claimed that Simulated Annealing is very reliable if it is applied to the minimization of multimodal functions at high computational cost, which may linearly with the number of dimensions of objective functions.

In this paper, we propose a new method for global optimization. Inspired by the human sketching process, increase, Stochastic Sketching was developed and designed to simulate such a process and is applied to solve global optimization problems in this paper. In the beginning, there are only an empty canvas, drawing tools, and a still life. First, we usually roughly draw some lines to indicate the size and location of the still life on the canvas. Then we refine the sketch iteratively with different levels of likeness between the sketch and the still life. Although the iterations of the sketching process for human beings may not be discrete but rather continuous, in principle, we make our artwork finer and finer at each iteration in some sense. The sketch is not finished until we are satisfied with it according to certain criteria or standards we have mind. This sketching process appears to be simple and is simulated by Stochastic Sketching. Components of Stochastic Sketching are all created based on the same concept, and so is the controlling mechanism of Stochastic Sketching.

The organization of this paper is organized as follows. Stochastic Sketching, the method we propose, is introduced and discussed in detail in Section 2. Section 3 presents the mathematical foundations and the connection between Stochastic Sketching and the fundamental theorem. The convergence theorem for Stochastic Sketching is also provided. Finally, discussion and conclusions are given in Section 5.

2. STOCHASTIC SKETCHING

In the section, a new method called *Stochastic Sketching* is proposed for global optimization. We consider the following general unconstrained minimization problem. Given an objective function f: $\mathbf{R}^n \rightarrow \mathbf{R}$, find an approximate global minimum of $f(\cdot)$. Stochastic Sketching is an optimization algorithm that searches for the global minimum stochastically by simulating a specific kind of human behavior − sketching. Stochastic Sketching attempts to simulate the process a typical person uses to draw a sketch, and this is the reason why the proposed algorithm is called *Stochastic Sketching*. In the beginning, there are only an empty canvas, drawing tools, and a still life. First, we usually roughly draw some lines to indicate the size and location of the still life on the canvas. Then we refine the sketch iteratively with different levels of likeness between the sketch and the still life. Although the iterations of the sketching process for human beings may not be discrete but rather continuous, in principle, we make our artwork finer and finer at each iteration in some sense. The sketch is not finished until we are satisfied with it according to certain criteria or standards we have in mind. This sketching process appears to be simple and is simulated by Stochastic Sketching.

2.1 One-dimensional Stochastic Sketching

In this subsection, Stochastic Sketching will be discussed in detail and introduced on the basis of its essential components. A visualization example of Stochastic Sketching is presented in [30]. The one-dimensional algorithm is considered first, and then the multi-dimensional case will be addressed in the next subsection. The one-dimensional algorithm Stochastic Sketching is given in Algorithm 2.1. An instance of Stochastic Sketching mainly consists of several components, including a *sampling guide* with a sequence of *zooming controllers*, a *sketching model*, a sequence of *precision thresholds*, and a *satisfaction criterion*. In the following, we will introduce each component.

2.1.1 Sampling guide

The *sampling guide* is a probability density function which is controlled by a zooming controller and is used to confine and manipulate the distribution of sampling points with the value of the zooming controller.

In order to fulfill the requirements of a sampling guide, we currently a function family $p_{f,c}(\cdot)$ defined as follows. Given a continuous function f: $[a, b] \to \mathbf{R}$ and a parameter $c \in \mathbf{R}_+$ (i.e., the zooming controller), we define

$$p_{f,c}(x) = \begin{cases} \dfrac{\exp(-f(x)/c)}{\int_a^b \exp(-f(z)/c)dz} & , x \in [a,b] \\ 0 & , \text{otherwise.} \end{cases} \tag{1}$$

To verify whether the function defined above satisfies the requirements of a sampling guide, the reader may refer [30, 31]. Since $p_{f,c}(\cdot)$ represents the distribution of x, it "guides" sampling points so that they are spread throughout the regions of interest.

Algorithm 2.1 One-dimensional Stochastic Sketching

x_{best} is the best solution that Stochastic Sketching has found yet
Initialize N_o, c_o, c_f, c_α, ζ_o, ζ_b, P_s, and x_{best}
$c \Leftarrow c_o$
$\zeta \Leftarrow \zeta_o$
Construct the sketching function $s(x)$ with $N_o + 2$ points
Compute the sampling guide $p_{s,c}(x)$
repeat
  repeat
    Generate a sampling point, x_0, randomly with the distribution $p_{s,c}(x)$
    Evaluate $f(x_0)$
    Update $s(x)$ and $p_{s,c}(x)$ according to $(x_0, f(x_0))$
    if $f(x_0) < f(x_{best})$
      $x_{best} \Leftarrow x_0$
    endif
  until satisfaction criterion
  Calculate next c and ζ
until termination criteria
Output $(x_{best}, \quad f(x_{best}))$

2.1.2 Zooming controller

The role of the zooming controller c in Equation (1) must be identified. Consider the following objective function:

$$f(x) = \sin(x) + \sin(\frac{10x}{3}) + \log_{10}(x) - 0.84x, \quad x \in [0.1,6.0],$$

which illustrates the effect of zooming controllers.

As shown in Fig. 1, when c is high enough, $p_{f,c}(\cdot)$ approaches a uniform distribution, which is not biased to any specific region. When c decreases, certain regions with smaller function values tend to be sampled with higher probability while other regions with larger function values do not. Hence, $p_{f,c}(\cdot)$ defined in Equation (1) is currently adopted as the sampling guide.

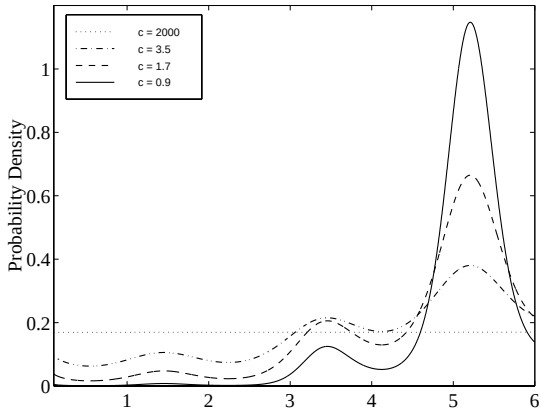

Fig. 1. Sampling guides with different zooming controllers.

Certainly the zooming controller plays the most crucial role in the sampling guide $p_{f,c}(\cdot)$ and even in the whole process of Stochastic Sketching. Since a higher value of the zooming controller results in less bias to regions with smaller function values and vice versa, it is rational to set the zooming controller high enough at the beginning of optimization and to decrease it during the process. A straightforward principle for setting the zooming controller is discussed below.

It is assumed that f_l and f_u represent the possible lower bound and upper bound of $f(\cdot)$, respectively. Thus, for a given zooming controller c, the corresponding probability densities for f_l and f_u are

$$P_{u,c} = \frac{\exp(-f_l / c)}{\int_a^b \exp(-f(z)/c)dz} \tag{2}$$

and

$$P_{l,c} = \frac{\exp(-f_u / c)}{\int_a^b \exp(-f(z)/c)dz}, \tag{3}$$

which are the possible upper bound and lower bound of the probability density function $p_{f,c}(\cdot)$, respectively.

(A) Initial Value

Because of the property of the zooming controller, we may expect its value to be so high that there is no bias at first. Let c_0 denote the initial value of the zooming controller. For a given tolerance of the initial sampling probability density ratio (0, 1), the following equation should be satisfied:

$$\frac{P_{l,c_0}}{P_{u,c_0}} \geq 1 - \varepsilon_0, \tag{4}$$

which means that the difference between the highest probability density and the lowest one must be as small as possible.

Hence, by Equation (2) and Equation (3),

$$\left[\frac{\exp(-f_u / c_0)}{\int_a^b \exp(-f(z)/c_0)dz}\right] / \left[\frac{\exp(-f_l / c_0)}{\int_a^b \exp(-f(z)/c_0)dz}\right] \geq 1 - \varepsilon_0, \tag{5}$$

is expected to hold.

Therefore, we can have that

$$\frac{\exp(-f_u / c_0)}{\exp(-f_l / c_0)} \geq 1 - \varepsilon_0,$$

$$\frac{f_l - f_u}{c_0} \geq \ln(1 - \varepsilon_0),$$

and finally,

$$c_0 \geq \frac{f_u - f_l}{-\ln(1-\varepsilon_0)}. \tag{6}$$

Although in practical problems, f_l and f_u may be not available, Equation (6) presents a primitive guideline for setting the initial value of the zooming controller and indicates that c_0 should be as high as possible so as to eliminate the sampling bias.

(B) Final value

In the final iteration, it is rational to expect that sampling points will be dedicated to some specific regions which possibly contain global minimum points. To realize such an intention in Stochastic Sketching, a similar criterion defined in the following should be satisfied:

$$\frac{P_{l,c_f}}{P_{u,c_f}} \leq \varepsilon_f, \tag{7}$$

where c_f denotes the final value of the zooming controller and ε_f is also the tolerance of the sampling probability density ratio.

After analogous operations, we obtain the guideline for setting the final value of the zooming controller:

$$c_f \le \frac{f_u - f_l}{-\ln \varepsilon_f}. \tag{8}$$

Equation (8) shows that the final value should be low enough to search some regions thoroughly and to locate the best solution that has been found.

(C) Updating rule

Having determinded the initial and final values, we will now discuss the rule for modifying the zooming controller. We also start with the sampling probability density ratio. For a given zooming controller *c*, we define

$$\gamma_c \equiv \frac{P_{l,c}}{P_{u,c}} \tag{9}$$

for convenience. As a result, we can easily obtain

$$c = \frac{f_u - f_l}{-\ln \gamma_c}. \tag{10}$$

Because γ_c represents the ratio of the lowest sampling probability density to the highest sampling probability density and is in some sense more meaningful to us, a simple rule is adopted here:

$$\gamma_c' = \alpha \cdot \gamma_c,$$

where $\alpha \varepsilon (0, 1)$ denotes the decreasing rate. It must be noted and admitted that the above rule is based on no firm theoretical foundation with respect to Stochastic Sketching and is taken from other existing methods.

Let $\gamma_{c'} = \gamma_c'$, which means that the change of *c* reflects the change made to γ_c. That is, according to Equation (10), given a decreasing rate $\alpha \varepsilon (0, 1)$,

$$\begin{aligned} c' &= \frac{f_u - f_l}{-\ln \gamma_{c'}} \\ &= \frac{f_u - f_l}{-\ln \gamma_c'} \\ &= \frac{f_u - f_l}{-\ln(\alpha \cdot \gamma_c)} \\ &= \frac{f_u - f_l}{(-\ln \alpha) + (-\ln \gamma_c)}. \end{aligned} \tag{11}$$

Because the zooming controller is always greater than zero, the inverse of the result in Equation (11) is used for further operations:

$$\frac{1}{c'} = \frac{(-\ln\alpha)+(-\ln\gamma_c)}{f_u - f_l} = \frac{-\ln\alpha}{f_u - f_l} + \frac{1}{c}. \tag{12}$$

Let

$$c_\alpha = \frac{f_u - f_l}{-\ln\alpha}, \tag{13}$$

which is called the decreasing factor of the zooming controller. Then, Equation (12) yields

$$\frac{1}{c'} = \frac{1}{c_\alpha} + \frac{1}{c}. \tag{14}$$

We finally obtain the updating rule of the zooming controller:

$$c' = \frac{c \cdot c_\alpha}{c + c_\alpha}. \tag{15}$$

Note that f_u and f_l are not really used in the derivation, but $f_u - f_l$ which represents the possible range of function values on $[a, b]$ is used. This means that if c_α should be calculated after the determination of α, an estimator of the value range of $f(\cdot)$ is needed. Currently, c_α is directly regarded as a parameter of Stochastic Sketching, and we set it as an initial parameter of the Stochastic Sketching Method.

2.1.3 Sketching model

The *sketching model* is used as the canvas and drawing tools, and the information obtained from function evaluations during the sketching process is stored in it. Functions (i.e., sketches of objective functions) constructed by the sketching model are called *sketching functions*, dented by $s(x)$. The following requirements are needed for a sketching model:

1. Sketching functions constructed by the model are continuous on $[a, b]$.
2. Values of a sketching function are available for all $x \in [a, b]$.
3. Values of the sampling probability density function that is determined by $s(x)$ and controlled by $c \in \mathbf{R}_+$ (i.e., $p_{s,c}(\cdot)$) are also available for all $x \in [a, b]$.

In the present work, a polynomial spline of order 1 with simple knots is used as the sketching model, and we will simply call it the *line model* in the rest of the paper. For instance, if the feasible region $[a, b] = [0, 20]$, and the sampling points are {(0, 2), (4, 9), (8, 5), (16, 5), (20, 7)}, then the sketching function constructed with the line model is a piecewise continuous function consisting of straight line segments and is shown in Fig. 2.

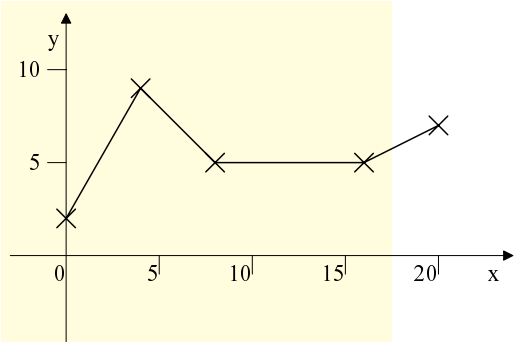

Fig. 2. Sketching function constructed with the line model.

It is obvious that the first two requirements of a sketching model, that it be continuous and evaluable, are satisfied when the line model is adopted. Because each part of the sketching function $s(x)$ is a line segment, the sampling guide defined in Equation (1) can be easily computed. The main problem for computing the sampling guide is the integral part. Based on the property of integration, only a line segment in one interval $[u, v]$ determined by any two adjacent sampling points, where $[u, v] \subseteq [a, b]$, needs to be considered. It is assumed that the line segment on $[u, v]$ is denoted by $g(x)$, and we will discuss two possible cases.

First, let $g(x) = sx + t$, where $x \in [u, v]$, and $s \neq 0$. Given a zooming controller c, the definite integral can be solved analytically:

$$\begin{aligned}\int_u^v \exp(-g(x)/c)dx &= \int_u^v \exp\left(-\frac{sx+t}{c}\right)dx \\ &= \exp\left(-\frac{t}{c}\right)\int_u^v \exp\left(-\frac{s}{c}x\right)dx \\ &= -\frac{c}{s}\exp\left(-\frac{t}{c}\right)\cdot\left[\exp\left(-\frac{s}{c}\right)\Big|_u^v\right] \\ &= -\frac{c}{s}\left[\exp\left(-\frac{sv+t}{c}\right)-\exp\left(-\frac{su+t}{c}\right)\right] \\ &= -\frac{c}{s}(\exp(-g(v)/c)-\exp(-g(u)/c)).\end{aligned} \tag{16}$$

For a degenerate case $g(x) = t$,

$$\begin{aligned}\int_u^v \exp(-g(x)/c)dx &= \exp\left(-\frac{t}{c}\right)\cdot\left[x\right]_u^v \\ &= (v-u)\cdot\exp\left(-\frac{t}{c}\right)\end{aligned} \tag{17}$$

Hence, we can see that the line model can be used as a sketching model. It has been used to implement the algorithm with good computational results as reported in [31].

2.1.4 Precision threshold

Since the sketch is sketched from coarsely to delicately, the standard of satisfaction we have in mind varies during the whole sketching process. The precision threshold, denoted by ζ, is used to model such an implicit standard on the basis of individual sampling points.

Definition 2.1 (Satisfying Point). Given an objective function $f(\cdot)$, a sketching function $s(\cdot)$, and a *precision threshold* $\zeta \in \mathbf{R}_+$, a sampling point x_0 is called a satisfying point if $|s(x_0) - f(x_0)| < \zeta$.

The precision threshold decreases in each iteration by the linear the currently adopted updating rule of

$$\zeta' = \zeta_\beta \cdot \zeta, \tag{18}$$

where $\zeta_\beta \in (0, 1)$ is the decreasing rate of the precision threshold. The initial value ζ_0 and the decreasing rate may depend on the objective function, but we use a fixed valued in our software implementation.

2.1.5 Satisfaction criterion

The satisfaction criterion models the standard we have in mind from a global viewpoint though it is partially based on the precision threshold.

Definition 2.2 (Satisfying Sketch). Given an objective function $f(\cdot)$, a sketching function $s(\cdot)$, a precision threshold ζ, and a satisfaction probability $P_s \in (0, 1)$, the sketching function $s(\cdot)$ is said to be satisfying if the next sampling point is a satisfying point with probability P such that $P > P_s$.

Hence only when the sketch is fine enough in certain regions, determined by the zooming controller, of the domain of the current iteration terminates. The satisfaction criterion controls the quality of the sketch and, hence, the solution quality.

2.1.6 Satisfaction factor

First, the sampling process can be viewed as a Bernoulli process with the parameter P according to the definition that a trial is labeled as a "success" if the corresponding sampling point is a satisfying point, and is labeled as a "failure" if the sampling point is not satisfying. It must be noted that P is fixed in a Bernoulli process but not in Stochastic Sketching. Nevertheless, the arguments are valid because under the same condition, P in Stochastic Sketching tends to increase due to the increase in the number of sampling points, while the critical region gets smaller. As a consequence, the result of the test that will be constructed is still valid.

Then, assume that we can tolerate a type I error size of up to 0.01 when testing the null hypothesis

$$H_0 : P = P_s$$

against the alternative

$$H_1 : P > P_s.$$

A statistical sample is defined as repeating trails until a failure is encountered, and the critical region is now parameterized by an integer N which represents the number of continual successful trails (i.e., the test statistic). If the test statistic of a sample is greater than or equal to N, the sample is in the critical region and H_0 is rejected. Otherwise, H_0 is not rejected. When H_0 is rejected, this means that the probability of success is greater than P_s with a significance level of 0.01. That is, the satisfaction criterion in Definition 2.2 is satisfied statistically with such a significance level.

Because $H_0 : P = P_s$, the size of the critical region (i.e., the probability type I error) can be calculated as follows. The probability of a statistical sample whose number of continual successful trails is exactly k is

$$P_s^k \cdot (1 - P_s)$$

because the last trail fails, while others are all successful. Therefore, the size of the critical region is obviously

$$\begin{aligned} \sum_{k=N}^{\infty} P_s^k \cdot (1 - P_s) &= (1 - P_s) \sum_{k=N}^{\infty} P_s^k \\ &= P_s^N (1 - P_s) \sum_{k=0}^{\infty} P_s^k \\ &= P_s^N \end{aligned} \tag{19}$$

Because the tolerance of the size of the type I error is 0.01, we should choose N according to Equation 19:

The size of type I error = $P_s^N = 0.01$

$$N = \left\lceil \frac{\ln(0.01)}{\ln P_s} \right\rceil.$$

Thus, we can calculate N, which is called the *satisfaction factor* in Stochastic Sketching, for a given P_s. In practice, we do not need to repeat trails until a failure occurs. If N continual trails are all successful, it is surely that H_0 should be rejected. The rejection of H_0 means that H_1 is accepted, and that the satisfaction criterion can be satisfied with a significance level of 0.01. The criterion is easy to implement and certainly practical.

2.2 Multidimensional Case

In our present work, generalization of Stochastic Sketching to the multidimensional case employs a dimensionality reduction technique called *space-filling curves*, which maps a multidimensional domain to a one-dimensional interval. Some other possible ap-

proaches still need to be explored by using space-filling curves, which is intuitive and theoretically valid.

2.2.1 Space-filling curves

Space-filling curves have fascinated mathematicians for over a century. George Cantor in 1878 demonstrated that any two finite-dimensional smooth manifolds have the same cardinality. This means that a square $[0, 1]^2$ may have a one-to-one and onto continuous mapping to a interval [0, 1]. In 1879, E. Netto showed that such a one-to-one and onto mapping is necessarily discontinuous. Nevertheless, an onto continuous mapping does exist from [0, 1] to $[0, 1]^2$. G. Peano constructed the first such curve in 1890. Further examples by D. Hilbert (in 1891), E. H. Moore (in 1900), H. Lebesgue (in 1904), W. Sierpinski (in 1912), G. Pólya (in 1913), and others followed.

Assume that $\mathbf{E}^n$ denotes the n-dimensional Euclidean space, which is $\mathbf{R}^n$ with the Euclidean norm defining the metric. We will now briefly introduce some concepts and definitions of space-filling curves ([32], pp. 4-5).

Definition 2.3 (Direct Image). If f is a function from a subset of $\mathbf{E}^m$ into $\mathbf{E}^n$, then

$$f \bullet (A) = \{f(x) \in R(f) : x \in A \cap D(f)\}.$$

where $A \subseteq \mathbf{E}^m$, is called the *direct image* of A under f. $D(f)$ denotes the domain, and $R(f)$ the range of the function f.

The term *curve* is defined as follows:

Definition 2.4 (Curve). If f: $A \rightarrow \mathbf{E}^n$, where $A = [0, 1]$, is continuous, then the image $f\bullet(A)$ is called a *curve*. $f(0)$ is called the *beginning point* of the curve, and $f(1)$ is called its *endpoint*.

Then, space-filling curves are defined in the following:

Definition 2.5 (Space-filling Curve). If $f : A \rightarrow \mathbf{E}^n$, $A = [0, 1]$, $n \geq 2$, is continuous and $J_n(f\bullet(A)) > 0$, then $f\bullet(A)$ is called a *space-filling curve*. $J_n(\cdot)$ denotes the n-dimensional *Jordan content* (area, volume) of a *Jordan measurable* subset of $\mathbf{E}^n$ such as in $J_3([0, 1]^3) = 1$.

In the present work, the space-filling curve constructed by Peano is adopted. Peano defined a map f_p from [0, 1] to $[0, 2]^2$ in terms of the operator $\odot$,

$$\odot t_j = 2 - t_j, \tag{20}$$

where $t_j = 0, 1, 2$, as follows:

$$f_p(0_3 t_1 t_2 t_3 t_4 t_5 \ldots) = \begin{pmatrix} 0_3 t_1 (\odot^{t_2} t_3)(\odot^{t_2+t_4} t_5)\ldots \\ 0_3 (\odot^{t_1} t_2)(\odot^{t_1+t_3} t_4)\ldots \end{pmatrix}^t, \tag{21}$$

where $\odot^i$ denotes the *i*th iterate of the operator $\odot$, $\hat{\mathbf{X}}^t$ is the transpose of a vector $\hat{\mathbf{X}}$, and

$$0_{\dot{3}}t_1t_2t_3t_4t_5\ldots = \frac{t_1}{3}+\frac{t_2}{3^2}+\frac{t_3}{3^3}+\frac{t_4}{3^4}+\frac{t_5}{3^5}+\ldots, t_j = 0,1, \text{ or } 2,$$

denotes *ternaries*. An illustration of the Peano curve is shown in Fig. 3, where the dashed line indicates the order of the squares traversed.

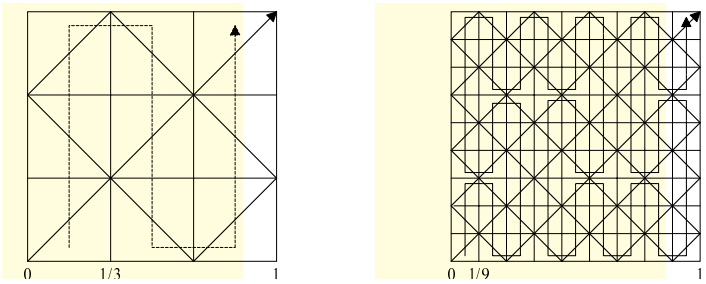

Fig. 3. First and second approximating polygon for the Peano Curve.

Definition 2.6 (Coordinate Functions). The Peano Curve is a mapping from [0, 1] to [0, 1]^2 and can be written as

$$f_p(t) = \begin{pmatrix} \psi_1(t) \\ \psi_2(t) \end{pmatrix}^t, \tag{22}$$

where $\psi_1(\cdot\ )$ and $\psi_2(\cdot\ )$ are the *coordinate functions*.

The Peano Curve is a two-dimensional space-filling curve. In order to generate an *n*-dimensional space-filling curve, the Steinhaus' result is used.

Definition 2.7 (Stochastically Independent). Given *n* measurable functions

$$\psi_1,\ldots,\psi_n : A \to \mathbf{R},$$

where A = [0, 1], which are called *stochastically independent* with respect to the Lebesgue measure if, for any *n* measurable sets $\mathbf{E}_1, \ldots, \mathbf{E}_n \subseteq \boldsymbol{R}$,

$$\mu\left[\bigcap_{j=1}^{n}\psi_j \bullet (E_j)\right] = \prod_{j=1}^{n}\mu\left[\psi_j \bullet (E_j)\right].$$

If such functions are also continuous and non-constant, an *n*-dimensional space-filling curve can be determined by them.

Theorem 2.1 (Steinhaus' Theorem). If $\psi_1,\ldots,\psi_n : A \to \mathbf{R}$, A = [0, 1], are continuous, non-constant, and stochastically independent with respect to the Lebesgue measure, then

$$f(t) = \begin{pmatrix} \psi_1(t) \\ \psi_2(t) \\ \vdots \\ \psi_n(t) \end{pmatrix} : A \to \psi_{1\bullet}(A) \times \psi_{2\bullet}(A) \times \ldots \psi_{n\bullet}(A) \tag{23}$$

is a space-filling curve.

To connect the Steinhaus' theorem and known two-dimensional space-filling curves, the concept of *uniform distribution* is needed.

Definition 2.8 (Uniformly Distributed). A function $\psi : A \to \boldsymbol{R}$, $A = [0, 1]$, is said to be *uniformly distributed* with respect to the Lebesgue measure if for any measurable set $E \subseteq \mathbf{R}$,

$$\mu [\psi_{\bullet} (E)] = \mu(E).$$

If ψ_1, ψ_2: $[0,1] \to [0,1]$ are continuous, non-constant, stochastically independent, and uniformly distributed, and if we let

$$f_s(t) = \begin{pmatrix} \psi_1(\psi_1(t)) \\ \psi_1(\psi_2(t)) \\ \psi_2(\psi_1(t)) \\ \psi_2(\psi_2(t)) \end{pmatrix}^t,$$

then $f_s(t)$ is a four-dimensional space filling curve as shown by Steinhaus.

Furthermore, an n-dimensional space-filling curve can be constructed as follows:

$$f(t) = \begin{pmatrix} \psi_1(t) \\ \psi_1 \circ \psi_2(t) \\ \psi_1 \circ \psi_3 \circ \psi_2(t) \\ \vdots \\ \psi_1 \circ \psi_2 \circ \cdots \circ \psi_2(t) \end{pmatrix}^t, \tag{24}$$

where $t \in [0, 1]$ and the operator $\circ$ denotes the composition of the functions. This has been proven in Milne [33].

2.2.2 Mapping into $\mathbf{R}^1$

Therefore, with space-filling curves, one-dimensional Stochastic Sketching can be used to solve multidimensional objective functions without modification. Given a multi-dimensional objective function f: $A \subseteq \mathbf{R}^n \to \mathbf{R}$, $n \geq 2$, a one-dimensional objective function $F(\cdot)$ is constructed for Stochastic Sketching as follows:
The function value is determined by

$$F(t) = f(\hat{X}(t)), \forall t \in A_s,$$

where $\hat{X} = (x_1(t), \ldots, x_n(t)) \in A$ defined as

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ \vdots \\ x_n(t) \end{pmatrix}^t = \begin{pmatrix} \delta_1(\Psi_1(t)) \\ \delta_2(\Psi_1 \circ \Psi_2(t)) \\ \delta_3(\Psi_1 \circ \Psi_2 \circ \Psi_2(t)) \\ \vdots \\ \delta_n(\Psi_1 \circ \Psi_2 \circ \cdots \circ \Psi_2(t)) \end{pmatrix}^t,$$

for a given $t \in A_s = [0, 1]$, and $\delta_i(\cdot)$, $i = 1, \ldots, n$, are appropriate scaling functions. Then,

$$F: A_s \to \boldsymbol{R},$$

where $A_s = [0, 1]$, is the function that is actually solved by means of Stochastic Sketching. The Stochastic Sketching method can be generalized to solve problems with multidimensional objective functions.

3. MATHEMATICAL FOUNDATIONS

To examine and analyze Stochastic Sketching, many experiments have been performed on several objective functions. The performance of the Stochastic Sketching on the test functions and a set of recommended parameter settings, and preliminary comparisons with related methods, including Evolution Strategies, Evolutionary Programming, and Genetic Algorithms, were given in [31]. We will only cite a small portion of them. In this section, the mathematical foundations and the theoretical side of Stochastic Sketching is presented. The Pincus theorem, the basis of the Stochastic Sketching is presented.

3.1 Pincus Theorem

In 1968, Pincus [34] derived a formula representing the coordinates of the point that maximizes a given function $f(\cdot)$ over the closure of a bounded domain A in n-dimensional Euclidean space. The main assumption is that $f(\cdot)$ attains a global maximum at exactly one point of A. The Pincus theorem will be introduced in the section, and the proof given by Pincus [34] will be sketched here.

Given a function f: $A \subseteq \mathbf{E}^n \to \mathbf{R}$, the maximization problem of $f(\cdot)$ was considered by Pincus. Let $\hat{X}^*$ and f^* denote the optimal point and the optimal function value, respectively. We will now present the lemma that will be needed in the proof of the Pincus theorem.

Lemma 3.1. Let f: $A \subseteq \mathbf{E}^n \to \mathbf{R}$ be a continuous function. Assume that $f(\cdot)$ attains a global maximum at exactly one point $\hat{X}^* \in A$. For $\varepsilon > 0$, let

$$N_\varepsilon = \{\hat{X} \in A: \left\| \hat{X} - \hat{X}^* \right\| < \varepsilon\}.$$

Then, given $\varepsilon > 0$, there exists $\delta > 0$ such that

$$\max_{\hat{X} \in A - N_\varepsilon} \{f(\hat{X}) - f(\hat{X}^*)\} < -\delta.$$

Proof. For the proof, we assume that

$$\max_{\hat{X} \in A - N_\varepsilon} \{f(\hat{X}) - f(\hat{X}^*)\} = 0$$

and then derive a contradiction. It follows that there exists a sequence $\{\hat{X}_i\}$, $\hat{X}_i \in A - N_\varepsilon$, such that

$$\{f(\hat{X}) - f(\hat{X}^*)\} \xrightarrow[i=\infty]{} 0.$$

Since $A - N_\varepsilon$ is compact, there exists a subsequence $\{\hat{X}_{i_j}\}$ that converges. Let

$$\lim_{j \to \infty} \hat{X}_{i_j} = \hat{Y}$$

Then, $\hat{Y} \in A - N_\varepsilon$, and by continuity of $f(\cdot)$, we have

$$\lim_{j \to \infty} f(\hat{X}_{i_j}) = f(\hat{Y}) = f(\hat{X}^*).$$

Since $f(\cdot)$ attains its global maximum at only one point, it follows that $\hat{X}^* = \hat{Y}$. This is a contradiction since $\hat{Y} \in A - N_\varepsilon$.

Then, the Pincus theorem will be presented in the following and a proof is also given by Pincus, please see [34].

Theorem 3.2 (Pincus Theorem). Let f: $A \subseteq \mathbf{E}^n \to \mathbf{R}$ be a continuous function. Assume that $f(\cdot)$ attains a global maximum at exactly one point $\hat{X}^* = (x_1^*, \ldots, x_n^*)$ of A. Then

$$x_i^* = \lim_{\lambda \to \infty} \frac{\int_A x_i \cdot \exp(\lambda f(\hat{X}))d\hat{X}}{\int_A \exp(\lambda f(\hat{X}))d\hat{X}}, \forall i = 1, \ldots, n. \tag{25}$$

3.2 Mathematical Interpretation of Stochastic Sketching

We will now can discuss the connection between Stochastic Sketching and the Pincus theorem. Starting with the Pincus theorem, the following corollary seems obvious.

Corollary 3.3. Let f: $A \subseteq \mathbf{E}^n \to \mathbf{R}$ be a continuous function. Assume that $f(\cdot)$ attains a global minimum at exactly one point $\hat{X}^* = (x_1^*, \ldots, x_n^*)$ of A. Then

$$x_i^* = \lim_{\lambda \to \infty} \frac{\int_A x_i \exp(-\lambda f(\hat{X}))}{\int_A \exp(-\lambda f(\hat{X}))d\hat{X}}, \forall i = 1, \ldots, n. \tag{26}$$

Proof. Since the identity

$$\max\{f(\hat{X}): \hat{X} \in A\} = -\min\{-f(\hat{X}): \hat{X} \in A\}$$

holds, the corollary is proved.

Given a one-dimensional objective function f: $[a, b] \to \mathbf{R}$, according to Equation (26), we have the global minimum point located at

$$x^* = \lim_{\lambda \to \infty} \frac{\int_a^b x \cdot \exp(-\lambda f(x))dx}{\int_a^b \exp(-\lambda f(x))dx}. \tag{27}$$

Let $\lambda = 1/c$; through a simple substitution of variables, we obtain

$$x^* = \lim_{c \to 0^+} \frac{\int_a^b x \cdot \exp(-f(x)/c)dx}{\int_a^b \exp(-f(x)/c)dx}. \tag{28}$$

Also, by the Mean Value Theorem for Definite Integrals, we can obtain $\eta \in$ [a, b], such that

$$\exp(-f(\eta)/c) = \frac{1}{b-a}\int_a^b \exp(-f(x)/c)dx.$$

Therefore, the sampling guide we have defined in Equation (1) is written as

$$\begin{aligned} \frac{\int_a^b x \cdot \exp(-f(x)/c)dx}{\int_a^b \exp(-f(x)/c)dx} &= \int_a^b x\left[\frac{\exp(-f(x)/c)}{(b-a)\cdot\exp(-f(\eta)/c)}\right]dx, \eta \in [a,b] \\ &= \int_a^b x\left[\frac{\exp(-f(x)/c)}{\int_a^b \exp(-f(z)/c)dz}\right]dx \\ &= \int_a^b x \cdot p_{f,c}(x)dx \\ &= E_c(x), \end{aligned} \tag{29}$$

where $E_c(x)$ denotes the expected value of x with the probability density function $p_{f,c}(x)$ for a given $c > 0$.

Taking the limit and by formula [35], we finally obtain

$$\begin{aligned} x^* &= \lim_{c \to 0^+} \frac{\int_a^b x \cdot \exp(-f(x)/c)dx}{\int_a^b \exp(-f(x)/c)dx} \\ &= \lim_{c \to 0^+} E_c(x) \end{aligned} \tag{30}$$

with the probability density function $p_{f,c}(x)$.

Hence, it is known that the coordinate of the global minimum point is the expected value of x according to the probability density function $p_{f,c}(x)$ when c approaches 0 from the right hand side. From this standpoint, Stochastic Sketching is the method that can be

used to *track the expected value of x stochastically* in order to find the global minimum point of $f(\cdot)$.

3.3 Convergence of Stochastic Sketching

A lot of work has been devoted to deriving the sufficient conditions for convergence of the general global random search algorithms. The formal scheme of global random search algorithms is introduced in Algorithm 3.1, and the description is based on Zhigljavsky's book ([36], p. 85).

Algorithm 3.1 Global Random Search Algorithm

/* A is the region of interest.*/
$t \Rightarrow 1$
Choose a probability distribution P_1 on A.
repeat
  Generate sampling points, $x_1^t, \ldots, x_{N_t}^t$, with the distribution P_t
  Evaluate $f(x_1^t), \ldots, f(x_{N_t}^t)$
  Construct a probability distribution P_t on A based on a fixed rule
  $t \Leftarrow t + 1$
until termination criterion

Let $N_t = 1$, $\forall t \in \mathbf{N}$; Stochastic Sketching is certainly an instance of global random search algorithm. Thus, general results for the convergence of global random search algorithms can be applied to Stochastic Sketching. For convenience, the probabilistic convergence criterion that is needed in the convergence theorem is taken from Torn and Zilinskas' book ([1], p. 78) and described as follows:

Definition 3.1 (Convergence with Probability One). The sequence $\{\hat{X}_i\}$ converges to the random vector $\hat{X}$ with probability one *if*

$$Prob\left\{\lim_{i\to\infty}\{\hat{X}_i\} = \hat{X}\right\} = 1.$$

The following convergence theorem was proposed by Zhigljavsky ([36], p. 88). The version and the proof presented below are from Bäack ([3], p. 50). It is also known as the Borel-Cantlli lemma.

Theorem 3.4 (Convergence of Global Random Search Algorithms). Let $f(\cdot)$ be continuous in the vicinity of $\hat{X}$ and assume that

$$\forall_\varepsilon > 0 : \sum_{t=1}^{\infty} q_t(\varepsilon) = \infty, \tag{31}$$

where

$$q_t(\varepsilon) = \inf_{1\le i<t}\left\{Prob\{\hat{X}_i \in U_\varepsilon(\hat{X}^*)\}\right\} \tag{32}$$

and $U_\varepsilon(\hat{X}^*)$ denotes a hypersphere of radius ε, centered around $\hat{X}^*$. Then for any $\delta > 0$, the sequence of random vectors $\hat{X}_1, \hat{X}_2, \ldots$ generated by Algorithm 3.3 with $\forall t : N_t = 1$ falls infinitely often into the level set $L_{f^*+\delta}$ with probability one.

Proof. Fix $\delta > 0$ and find $\varepsilon = \varepsilon(\delta) > 0$ such that $U_\varepsilon(\hat{X}^*) \subseteq \boldsymbol{L}_{f+\delta}$. Determine the sequence of independent random variables $\{\chi_t\}$ on the two-point set $\{0, 1\}$ such that

$$Prob(\chi_t = 1\} = 1 - Prob(\chi_t = 0\} = q_t(\varepsilon). \tag{33}$$

Then, $Prob\{\hat{X}_t \in U_\varepsilon(\hat{X}^*)\} \geq Prob(\chi_t\}$, and the theorem is proved if one can show that $\{\chi_t\}$ infinitely often takes a value of one. The latter follows from Equation (31) and Borel's zero-one law, which completes the proof.

As a consequence, the convergence of Stochastic Sketching can be described by Theorem 3.4.

4. SUMMARY EXPERIMENTAL RESULTS

To examine and analyze Stochastic Sketching, many experiments were preformed on several objective functions. We list two difficult objective functions in this paper for the experimental results of other objective functions, please see [31]

1. One-dimensional Test Function [1, 37, 38]:

$$f_2(x) = -\sum_{j=1}^{5} \sin((j+1)x + j),$$
$$-20.0 \leq x \leq 20.0.$$

- Number of local minima = 39
- Number of global minima = 7
-

x^*	f^*
-19.3409	-12.0312
-13.0578	-12.0312
-6.7746	-12.0312
-0.4914	-12.0312
5.7918	-12.0312
12.0750	-12.0312
18.3582	-12.0312

2. Two-dimensional Restringing Function [1, 38, 37]:

$$f_6(\hat{X}) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2),$$
$$-5.0 \leq x_i \leq 5.0, \quad i = 1,\ 2.$$

- Number of local minima = 961
- Number of global minima = 1
- $[\hat{X}, f^*] \cong [(0.0, 0.0), -2.0]$

Based on many experimental results, Table 1 shows the recommended settings for the essential parameters of Stochastic Sketching. Under the same successful rate, by adopting the parameter setting

$(c\ , c_\alpha, \zeta_0, \zeta_\beta, P_s) = (1500,\ 3.25,\ 5,\ 0.50,\ 0.2),$

Table 1. Recommended parameter settings for stochastic sketching.

Parameter	Notation	Range
Initial Value of the Zooming Controller	c_0	[1500,2000]
Decreasing Factor of the Zooming Controller	c_α	[3.25, 3.50]
Initial Value of the Precision Threshold	ζ_0	[5, 10]
Decreasing Rate of the Precision Threshold	ζ_β	[0.45, 0.50]
Satisfaction Probability	P_s	[0.4, 0.6]

Table 2. Comparison with evolution strategies.

Function	Stochastic Sketching		Evolutional Strategies	
	Succ. Rate	Avg. Eval.	Succ. Rate	Avg. Eval.
f_1	1.0	126.67	1.0	310.50
f_2	1.0	4050.41	0.9	1645.83

Stochastic Sketching can find the global optimum with about 1836.58 function evaluations and is definitely comparable in performance to the Evolution Strategies.

5. DISCUSSION AND CONCLUSIONS

The resolution of random numbers are a serious problem that Stochastic Sketching has yet been encountered. A uniform random number generator that is capable of generating random numbers with better resolution is necessary when Stochastic Sketching is applied to relatively high-dimensional objective functions. Even though using space-filling curves to generalize Stochastic Sketching to solve multidimensional objective functions is theoretically feasible, it may introduce some severe practical difficulties, such as the resolution of random numbers. Therefore, other approaches for generalization to high dimension form the one-dimensional method should be further studied. The line model is currently used as the sketching model in the implementation. Other models fulfilling the requirements of sketching models may be used. The parameter settings always pose a problem for random search methods and, hence, stochastic algorithms. Good initial parameters were reported in [31] based on previous computing experience with the algorithms. With this fixed parameters setting, the Stochastic Sketching method performs reasonably well. In future studies, we may try to let some of the parameters of Stochastic Sketching be adjusted automatically by introducing the *self-adaptation* technique to Stochastic Sketching, which currently prevails in the field of Evolutionary Algorithms.

In summary, a new method based on the simulation of human behavior has been proposed for global optimization. All essential components of Stochastic Sketching have been introduced and discussed in detail as well as the background and concepts according to which Stochastic Sketching was designed and developed. The mathematical foundation of Stochastic Sketching is the Pincus theorem. Some multi-modal functions with good results. It seems that this method is comparable in solution quality and the number of function evaluations with the Evolution Strate-

gies method and is better than various variants of the genetic algorithms. The calculation involved in each step for Stochastic Sketching is less than that for Evolution Strategies.

Table 3. Comparison with evolutionary programming.

Function	Stochastic Sketching		Evolutional Programming	
	Succ. Rate	Avg. Eval.	Succ. Rate	Avg. Eval.
f_1	1.0	126.67	1.0	305.00
f_2	1.0	4050.41	0.9	1952.50

REFERENCES

1. A. Torn and A. Zilinskas, *Global Optimization*, Vol. 350 of LNCS, Springer-Verlag, New York, USA; Berlin, 1989.
2. A. V. Levy and A. Montalvo, "The tunneling algorithm for the global minimization of functions," *SIAM Journal on Scientific Computing*, Vol. 6, No. 1, 1985, pp. 15-29.
3. T. Back, *Evolutionary Algorithms in Theory and Practice*, Oxford University Press, New York, USA, 1996.
4. L. J. Fogel, A. J. Owens, and M. J. Walsh, *Artificial Intelligence through Simulated Evolution*, Wiley, New York, 1966.
5. K. A. D. Jong, *An Analysis of the Behavior of a Class of Genetic Adaptive Systems*, Ph.D. thesis, University of Michigan, 1975.
6. D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Company, Inc., Reading, MA, USA, 1989.
7. J. D. Babley, "The behavior of adaptive systems which employ genetic and correlation algorithms," Ph.D. thesis, University of Michigan, 1967.
8. R. S. Rosenberg, "Simulation of genetic populations with biochemical properties," Ph.D. thesis, University of Michigan, 1967.
9. R. Weinberg, "Computer simulation of a living cell," Ph.D. thesis, University of Michigan, 1970.
10. D. J. Cavicchio, "Adaptive search using simulated evolution," Unpublished doctoral dissertation, University of Michigan, 1970.
11. R. B. Hollstien, "Artificial genetic adaptation in computer control systems," Ph.D. thesis, University of Michigan, 1971.
12. J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proceedings of the First International Conference on Genetic Algorithms and Their Applications*, pp. 93-100.
13. D. E. Goldberg, "Genetic algorithms with sharing for multimodal function optimization," in Grefenstette [14], pp. 41-49.
14. D. E. Goldberg and R. E. Smith, "Nonstationary function optimization using genetic algorithms with dominance and diploidy," *Genetic Algorithms and Their Applications: Proceeding of the Second International Conference on Genetic Algorithms*, 1987, pp. 59-68.
15. K. Deb and D. E. Goldberg, "An investigation of niche and species formation in genetic function optimization," in Schaffer [16], pp. 42-50.
16. J. D. Schaffer, R. A. Caruana, L. J Eshelman, and R. Das, "A study of control pa-

rameters affecting online performance of genetic algorithms for function optimization," in *Proceedings of the Third International Conference on Genetic Algorithms*, pp. 51-60.
17. M. F. Bramlette, "Initialization, mutation and selection methods in genetic algorithms for function optimization," in Belew and Booker [19], pp. 100-107.
18. M. Muselli and S. Ridella, "Global optimization of functions with the interval genetic algorithm," *Complex System*, Vol. 6, No. 3, 1992, pp. 193-212.
19. W. E. Hart and R. K. Belew, "Optimizing an arbitrary function is hard for the genetic algorithm," in *Proceedings of the Fourth International Conference on Genetic Algorithms*, pp. 190-195.
20. V. S. Gordon and D. Whitley, "Serial and parallel genetic algorithms as function optimizers," in Forrest [39], pp. 177-183.
21. D. Powell and M. M. Skolnick, "Using genetic algorithms in engineering design optimization with non-linear constraints," in Forrest [39], pp. 424-431
22. J. Arabas, J. M. Mulawka, and J. Pokrasniewicz, "A new class of the crossover operators for the numerical optimization," in Eshelman [23], pp. 42-48.
23. Z. Michalewica, "Genetic algorithms, numerical optimization, and constraints," in *Proceedings of the Sixth International Conference on Genetic Algorithms*, pp. 151-158.
24. S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, Vol. 220, No. 4598, 1983, pp. 671-680.
25. V. Cerny, "Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm," *Journal of Optimization Theory and Application*, Vol. 45, 1985, pp. 41-51.
26. G. H. Sasaki and B. Hajek, "The time complexity of maximum matching by simulated annealing," *Journal of ACM*, Vol. 35, No. 2, 1988, pp. 387-403.
27. U. Faigle and R. Schrader, "On the convergence of stationary distributions in simulated annealing algorithms," *Information Processing Letters*, Vol. 27, No. 4, 1988, pp. 189-194.
28. F. Aluffi-Pentini, V. Parisi, and F. Zirille, "Global optimization and stochastic differential equations," *Journal of Optimization Theory and Application*, Vol. 47, No. 1, 1985, pp. 1-16.
29. A. Coranna, M. Marchesi, C. Martini, and S. Ridella, "Minimizing multimodal functions of continuous variables with the *simulated annealing* algorithm," *ACM Transactions on Mathematical Software*, Vol. 13, No. 3, 1987, pp. 262-280.
30. Y. P. Chen, "Stochastic sketching: A new method for global optimization," Master's thesis, Dept. of CSIE, National Taiwan University, 1997.
31. Y. P. Chen, J. T. Horng, and C. Y. Kao, "Stochastic sketching: A new method for global optimization," *Soft Computing*, Vol. 3, Issue 2, 1999, pp. 101-110.
32. H. Sagan, *Space-Filling Curves*, Springer-Verlag, New York, USA, Berlin, 1994.
33. St. C. Mine, "Peano curves and smoothness of functions," *Advances in Mathematics*, Vol. 35, 1980, pp. 129-157.
34. M. Pincus, "A closed form solution of certain programming problems," *Operations Research*, Vol. 16, No. 3, 1968, pp. 690-694.
35. G. B. Thomas, Jr. and R. L. Finney, *Calculus and Analytic Geometry*, Addison-Wesley Publishing Company, Inc., Reading, MA, USA, 1996.

36. A. A. Zhigljavsky, *Theory of Global Random Search, Mathematics and Its Applications*, Vol. 65, Kluwer Academic Publishers, Norwell, Massachusetts, USA, 1991.
37. B. C. Cetin, J. Barhen, and J. W. Burdick, "Terminal repeller unconstrained subenergy tunneling (TRUST) for fast global optimization," *Journal of Optimization Theory and Applications*, Vol. 77, No. 1, 1993, pp. 97-126.
38. G. H. Koon and A. V. Sebald, "Some interesting test functions for evaluation evolutionary programming strategies," in *Proceedings of the Fourth Annual Conference on Evolutionary Programming*, 1995, pp. 479-499.
39. L. J. Eshelman and J. D. Schaffer, Crossover's Niche, in *Proceedings of the Fifth International Conference on Genetic Algorithms*, pp. 9-14. University of Illinois at Urbana-Champaign, July 17-21, 1993. Office of Naval Research, Naval Research Laboratory, Philips Laboratory, North American Philips Corporation and International Society for Genetic Algorithms, Morgan Kaufmann Publishers, Inc.

6. APPENDIX

Table 4. Notations of stochastic sketching.

Notation	Description
$s(\cdot)$	sketching function
$P_{f,c}(\cdot)$	sampling guide for a given $f(\cdot)$ and c
N_0	number of initial sampling points
c	zooming controller
c_α	decreasing factor of the zooming controller
ζ	precision threshold
ζ_β	decreasing rate of the precision threshold
Ps	satisfaction probability

Table 5. Other notation.

Notation	Description
$\mu(\cdot)$	Lebesgue measure function
$Prob(\cdot)$	probability function
ϕ	empty set
$\mathbf{B}$	{0,1}
$\mathbf{E}$	$\mathbf{R}$ with the Euclidean norm
A	feasible region of the objective function
$x, y, \ldots$	scalars
$\hat{X}\,\hat{Y}, \ldots$	n-dimensional vectors
$\hat{0}$	zero vector (0,0,…,0)
f^*	global optimum
$x^*, \hat{X}^*$	global optimum point
f	local optimum
$x^*, \hat{X}^*$	local optimum point

f^*	optimum reported by an algorithm
x^+, $\hat{X}^+$	optimum point reported by an algorithm

Jorng-Tzong Horng (洪炯宗) was born in Nantou, Taiwan, on April 10, 1960. He received the Ph.D. degree in Computer Science and Information Engineering from National Taiwan University, Taipei, Taiwan, in April 1993. He is currently an Associate Professor of the Department of Computer Science and Information Engineering at National Central University, Chung-Li, Taiwan. His current research interests include object-oriented database systems, distributed database systems, query processing and optimization, genetic algorithms, and bioinformatics.

Ying-Ping Chen (陳穎平) received both the B.S. degree and the M.S degree in Computer Science and Information Engineering from the National Taiwan University in 1995 and 1997, respectively. From 1999, he is a PhD student in the Department of Computer Science at the University of Illinois at Urbana-Champaign. His research interests are in the field of Genetic Algorithms and Evolutionary Computation.

Cheng-Yan Kao (高成炎) was born in Taipei, Taiwan, 1948. He received B.S. in mathematics from National Taiwan University, Taipei, Taiwan, in 1971, and the M.S. degree in computer science in 1976, the M.S. degree in statistics in 1978, and the Ph.D. degree in computer science in 1981, all from the University of Wisconsin-Madison. He worked for Ford Aerospace, the Unisys Corporation, and worked for General Electric from 1980 to 1989 at the Johnson Space Center, NASA, Houston, TX. He has been a professor with the Department of Computer Science and Information Engineering, National Taiwan University since 1990. He has published more than 40 technical papers in various journals and conference records. His research interests include evolutionary computation, bioinformatics, optimization, and artificial intelligence.