Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Chaotic Global Optimization by Direct Stability Control of Gradient-Based Systems

Kazuaki Masuda
Faculty of Engineering
Kanagawa University
Yokohama, Kanagawa, Japan, 221-8686
masuda@is.kanagawa-u.ac.jp

Kenzo Kurihara
Faculty of Engineering
Kanagawa University
Yokohama, Kanagawa, Japan, 221-8686
kurihara@is.kanagawa-u.ac.jp

***Abstract*—A novel chaotic global optimization (CGO) method is proposed. A class of gradient-based systems which use the value of the objective function to manipulate the stability of all local optima plays a significant role in the method. In such a system, local optima with lower values of the objective can be made more stable. Because, with an appropriate choice of its bifurcation parameter, the globally optimal solution of an optimization problem becomes the unique stable fixed point, all trajectories can converge to the global optimum under the situation. Best of authors' knowledge, this is the first appearance of a chaotic optimization method which only uses a single trajectory of gradient-based systems and never requires complicated operations like annealing the control parameters or finding multiple local minima. Additionally, in order to improve the applicability of the proposed CGO method, an adaptive method of directly controlling the stability at local optima is also proposed. Numerical results also show that the proposed method is promising for the tested problems.**

***Index Terms*—global optimization, chaos, gradient-based system, stability, iterated map**

I. INTRODUCTION

Global Optimization is getting more and more important in various fields of engineering. Although many mathematical or heuristic methods have been proposed, the activity of researches for global optimization never declines because there is no universal method to solve all kinds of problems. As the performance of optimization methods depends heavily on the structure of problems, the optimization community is requested to suggest many options and explain when and why each option works well.

A class of emerging global optimization method is the *chaotic global optimization (CGO)* [1]–[4]. In the CGO framework, a deterministic search system with nonlinearity (e.g., a gradient-based system including the steepest descent) is used for both local and global search. The CGO utilizes the fact that the stability of a nonlinear dynamical system can be varied by a certain parameter, the so called *control parameter*, within it. While various types of such nonlinear systems have been proposed, they get unstable in common as the control parameter is increased. When the system is stable, it works as an ordinary local search system in which trajectories are attracted to one of local optima. Conversely, when the system gets unstable and chaotic, almost all of trajectories wander so as to search for optimal solutions globally.

To find the global optimum in the CGO, the *chaotic annealing (CA)* has also been used in common. The CA, in imitation of the *simulated annealing (SA)* for global optimization [5], gradually decreases the control parameter from the chaotic phase to the stable phase. During the annealing process, trajectories in the search system repeatedly visit a local optimum, escape from it and move to another one in the chaotic phase, while they are expected to converge to the global optimum at the final stage. It is reported that the CA is promising for combinatorial and continuous nonconvex optimization problems.

However, it is also known that the CA may fail for some problems. Reasons for the failure can be explained as follows:

1) A nonlinear search system has a particular kind of bifurcation structure originated from the objective function to be optimized. It is not always true that the global optimum corresponds to the most stable fixed point in the system. Under the circumstances, local optima other than the global optimum are likely to be found more frequently by the CGO.
2) Even if the global optimum is actually the most stable fixed point, the success of the CGO approach heavily depends on the decreasing schedule of the control parameter during the CA process. Too fast or too slow reduction of the control parameter may lose the possibility of finding the global optimum.

Therefore, a nonlinear search system in which undesired attraction to local minima can be prevented, is required for the CGO. Eventually, a CGO method which works without the CA process is desired.

To address the above issues, a new CGO method is proposed in this paper. The proposed method has three novelties to conventional CGO methods. First, a gradient-based system is proposed in which a nonnegative and nondecreasing function with respect to the value of the objective function, the so called *stability control function* in this paper, is multiplied to the gradient so as to manipulate the stability of all local optima. In other words, local optima with better objective function value can be forced to be more stable. Second, the proposed method does not require the CA process any more as the stability of local optima can controlled only by the stability control function. All the parameters originally in the

978-1-4244-2794-9/09/$25.00

gradient-based system can be fixed. Third, when parameters are set properly so as to make the global optimum stable but other local optima unstable, trajectories of the system starting from all points can converge to the global optimum steadily. In addition, an adaptive configuration of the stability control function is proposed with some parameters introduced in it. The proposed method is demonstrated through examples for a simple one-dimensional problem to verify its effectiveness.

This paper is organized as follows. The framework of the proposed CGO method, which uses the gradient-based system with the stability control function, is introduced in Section II. Section III expansively proposes an adaptation of the stability control function. Sections II and III also present numerical experiments to verify the effectiveness of the proposed method. Brief conclusions and future works are given in Section IV.

II. Chaotic Global Optimization Based on Gradient-Based System

Consider the following optimization problem with a box constraint:

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x}), \tag{1a}$$

$$\text{s.t.} \quad \boldsymbol{l} \le \boldsymbol{x} \le \boldsymbol{u}. \tag{1b}$$

$\boldsymbol{x} = [x_1, \cdots, x_N]^T \in R^n$ denotes the decision variable; $f : R^N \to R^1$ is the continuously twice differentiable objective function; $\boldsymbol{l} = [l_1, \cdots, l_N] \in R^N$ and $\boldsymbol{u} = [u_1, \cdots, u_N] \in R^N$ are the lower and upper bound for the decision variable. Assume that $f(\boldsymbol{x})$ has at least an isolated global minimum and multiple local minima in the box, and that the Hessian matrix of $f(\boldsymbol{x})$, denoted by $H(\boldsymbol{x}) = \nabla^2 f(\boldsymbol{x})$, is positive definite at all local minima in the box. These hypotheses are naturally satisfied by most of global minimization problems. Let us note that the box constraint plays a significant role in the proposed CGO method.

A. Conventional Chaotic Global Optimization with Gradient-Based System and Chaotic Annealing

In this subsection, the conventional CGO method which uses gradient-based systems and the CA method is reviewed briefly. An instance of continuous-time gradient-based systems to solve (1) is given by

$$\frac{d\boldsymbol{x}(t)}{dt} = -\text{diag}[(u_i - x_i(t))(x_i(t) - l_i)] \cdot \nabla f(\boldsymbol{x}(t)). \tag{2}$$

According to the negative gradient, $-\nabla f$, and the variable metric matrix, $\text{diag}[(u_i - x_i)(x_i - l_i)]$, being positive definite in the box, (2) is obviously a descent system. The variable metric matrix practically works to confine the decision variable $\boldsymbol{x}$ in the box. Equation (2) is partitioned into two equalities by introducing a new variable $\boldsymbol{y} = [y_1, \cdots, y_N] \in R^N$, which satisfies $d\boldsymbol{x}(t)/dt = (\partial \boldsymbol{x}/\partial \boldsymbol{y}) \cdot d\boldsymbol{y}(t)/dt$, as follows:

$$x_i(t) = g_i(y_i(t)) \overset{\text{def}}{=} \frac{u_i + l_i \exp(-y_i(t))}{1 + \exp(-y_i(t))}, \quad i = 1, \cdots, N, \tag{3a}$$

$$\frac{d\boldsymbol{y}(t)}{dt} = -\nabla f(\boldsymbol{x}(t)). \tag{3b}$$

Equation (3a) is a one-to-one mapping from $\boldsymbol{u}$ to $\boldsymbol{x}$; similarly, the inverse map from $\boldsymbol{x}$ to $\boldsymbol{u}$ is defined by

$$y_i(t) = g_i^{-1}(x_i(t)) \overset{\text{def}}{=} \log \frac{u_i - x_i(t)}{x_i(t) - l_i}. \tag{4}$$

When (3b) is discretized by the Euler's finite difference method with the step length $h > 0$, (3) is transformed as follows:

$$\boldsymbol{x}_n = \boldsymbol{g}(\boldsymbol{y}_n), \tag{5a}$$

$$\boldsymbol{y}_{n+1} = \boldsymbol{y}_n - h \cdot \nabla f(\boldsymbol{x}_n), \tag{5b}$$

where $\boldsymbol{g}(\boldsymbol{y}) = (g_1(y_1), \cdots, g_N(y_N))^T$. Since, due to the assumptions on f and the definition of $\boldsymbol{g}$, there exist at least two different points $\boldsymbol{y}^1$ and $\boldsymbol{y}^2$ such that $\nabla f(\boldsymbol{g}(\boldsymbol{y}^1)) = \nabla f(\boldsymbol{g}(\boldsymbol{y}^2)) = \boldsymbol{0}$, det $H(\boldsymbol{g}(\boldsymbol{y}^1)) > 0$ and det $H(\boldsymbol{g}(\boldsymbol{y}^2)) > 0$, it is guaranteed by [6], [7] that (5b) is chaotic for a sufficiently large h.

In the system (5), the step length h is used for the *control parameter*. When h is relatively small, every trajectory in the space of $\boldsymbol{x}$ converges to a local minimum; however, when h is enough large for (5b) to be chaotic, trajectories in the space of $\boldsymbol{x}$ wander in the box so as to visit a local minimum, escape from it and move to another. In the CGO framework, the CA method, which sets the control parameter to a sufficiently large value at first to search for better solutions globally by tracking a chaotic trajectory, then gradually decreases the control parameter during the search so as to finally converge the trajectory to the global minimum, has been conventionally applied to nonlinear dynamical systems like (5).

Although the CGO in combination with the CA, may be effective for some problems, convergence to the global optimum cannot be guaranteed. It is mainly because the stability of local optima in the search system does not always correspond to the level of the objective function. In fact, the CA is likely to give the most stable fixed point of the gradient-based system rather than the globally optimal solution of the original problem. In (5), the stability of fixed points varies according not only to the gradient ∇f and the control parameter h, but also to the nonlinear function $\boldsymbol{g}$. In addition, the possibility of finding the global optimum depends on the decreasing schedule of the control parameter. For example, too fast or too slow annealing may converge trajectories to unexpected solutions. Furthermore, preliminary experiments are also necessary for choosing the initial value of the control parameter in order for the system to be chaotic.

The above-mentioned issues in the conventional CGO will be resolved one by one below. A gradient-based system is introduced in which the stability of fixed points is manipulated according to the value of the objective function. With such systems, the global optimum can be obtained without using the CA any more.

B. Gradient-Based System with Stability Control Function

Consider a simple gradient-based system of the steepest descent:

$$\frac{d\boldsymbol{x}(t)}{dt} = -\nabla f(\boldsymbol{x}(t)), \tag{6}$$

with the box constraint in (1) ignored at first. In (6), all local minima of (1) are stable fix points, but the stability of them is related only to the gradient ∇f. In order to manipulate the stability of fixed points according to the objective function values calculated there, a new system with a nonnegative and nondecreasing *stability control function* is introduced as follows:

$$\frac{d\boldsymbol{x}(t)}{dt} = -c(f(\boldsymbol{x}(t))) \cdot \nabla f(\boldsymbol{x}(t)), \tag{7}$$

in which $c : R^1 \to [0, \infty)$ is the stability control function. If the minimum value of the objective function in the box (1b), denoted by f^*, is known in advance, an instance of such stability control functions can be given by

$$c(f(\boldsymbol{x})) = \max\{f(\boldsymbol{x}) - f^* + \epsilon,\ 0\}, \tag{8}$$

where $\epsilon > 0$ is sufficiently small. By the Euler's finite difference method with the step length $h > 0$, (7) is transformed into an discrete-time system:

$$\begin{aligned}\boldsymbol{x}_{n+1} &= \boldsymbol{x}_n - h \cdot c(f(\boldsymbol{x}_n))\nabla f(\boldsymbol{x}_n) \\ &\stackrel{\text{def}}{=} F(\boldsymbol{x}_n; h).\end{aligned} \tag{9}$$

Local minima of f are stable fix points of (9) for relatively small h. On the other hand, it is also guaranteed by [6], [7] that (9) is chaotic for sufficiently large h.

Now, the stability of local minima of the objective function f in the system (9) is analyzed as follows. When $\boldsymbol{x}^o$ is a local minimum of f, the stability of $\boldsymbol{x}^o$ is determined by the Jacobian matrix of F,

$$DF(\boldsymbol{x}^o) = I - h \cdot c(f(\boldsymbol{x}^o))H(\boldsymbol{x}^o), \tag{10}$$

where the equality $\nabla f(\boldsymbol{x}^o) = \boldsymbol{0}$ is used. By the theory of nonlinear dynamical systems, the largest eigenvalue of the right hand side of (10) defines the linear stability around stable fixed points of the system (9) [8], [9]. Let $\lambda_{\max}(\boldsymbol{x}^o; h)$ be the largest eigenvalue of $DF(\boldsymbol{x}^o)$. $\boldsymbol{x}^o$ is stable if $-1 < \lambda_{\max}(\boldsymbol{x}^o) < 1$; $\boldsymbol{x}^o$ gets more stable as $\lambda_{\max}(\boldsymbol{x}^o; h)$ is more closer to -1. Due to the assumptions of $c(f(\boldsymbol{x}^o)) > 0$ and the positive definiteness of $H(\boldsymbol{x}^o)$, it is always true that $\lambda_{\max}(\boldsymbol{x}^o; h) > -1$ for any $h > 0$. At the same time, (9) has no other stable fix points than local minima of f.

The stability control function in (9) fundamentally makes local minima with lower value of f relatively more stable. The step length $h > 0$ can be used for making $\boldsymbol{x}^o$ less stable as h is increased. When $h > h_1(\boldsymbol{x}^o)$ for some $h_1(\boldsymbol{x}^o) > 0$ satisfying $\lambda(\boldsymbol{x}^o; h_1(\boldsymbol{x}^o)) = 1$, $\boldsymbol{x}^o$ becomes an unstable fixed point in (9) even though it is a local minimum of f. In the context of nonlinear systems, for some $h_1(\boldsymbol{x}^o) < h_2(\boldsymbol{x}^o) < h_3(\boldsymbol{x}^o)$, (9) has chaotic trajectories around $\boldsymbol{x}^o$ for $h \in (h_2(\boldsymbol{x}^o), h_3(\boldsymbol{x}^o))$. It should be noted that h_1 is unique for each local minima.

When f has N local minima, $\boldsymbol{x}_1^o, \cdots, \boldsymbol{x}_N^o$, let h_{11} be the largest value of h_1:

$$h_{11} = \max_i\{h_1(\boldsymbol{x}_i^o)\}, \tag{11}$$

and i_1 be the corresponding index number:

$$i_1 = \arg\max_i\{h_1(\boldsymbol{x}_i^o)\}. \tag{12}$$

The second largest value of h_1 and its index number are also defined as

$$h_{12} = \max_{i \neq i_1}\{h_1(\boldsymbol{x}_i^o)\}, \tag{13}$$

$$i_2 = \arg\max_{i \neq i_1}\{h_1(\boldsymbol{x}_i^o)\}. \tag{14}$$

For $h \in [h_{12}, h_{11})$, (9) has only a stable fixed point $\boldsymbol{x}_{i_1}^o$ so that all convergent trajectories gather only to the fixed point. On the other hand, if there exist some i satisfying $h_2(\boldsymbol{x}_i^o) < h < h_3(\boldsymbol{x}_i^o)$ for the same h, which may happen for many f's, chaotic trajectories coexist in the system. If the stability control function is chosen properly so that the unique stable fixed point $\boldsymbol{x}_{i_1}^o$ is equivalent to the global minimum of f, the global minimum is obtained by finding any convergent trajectory in (9) under the circumstance.

C. Satisfaction of Box Constraint

In general, chaotic trajectories of nonlinear dynamical systems can exist in a certain closed region; however, trajectories of (9) can't always be confined in the box (1b) of the original minimization problem. Temporarily left in II-B, the box constraint (1b) must be satisfied during the search.

In the proposed method, (1b) is simply satisfied by forcibly mapping the image of F into the box set. For instance, when the image $\boldsymbol{x}_{k+1}$ from $\boldsymbol{x}_k$ goes to the outside of the box, a shift operation defined by

$$s_i(x_i) \stackrel{\text{def}}{=} x_i - (h_i - l_i) \cdot \left\lfloor \frac{x_i - l_i}{h_i - l_i} \right\rfloor, \quad i = 1, \cdots, N, \tag{15}$$

where $\lfloor x \rfloor$ denotes the greatest integer which does not exceed a real value x. By (15), $S(\boldsymbol{x}) = (s_1(x_1), \cdots, s_N(x_N))^T$ always maps $\boldsymbol{x}$ into the box. It should be cautioned that the confinement operation be defined in a deterministic manner and that fixed points in the box never be moved by the operation. Another example of such operations may be the reflection at the boundary of the box.

The proposed method uses the composite map of the gradient-based system (9) and the shift operator (15):

$$\boldsymbol{x}_{n+1} = F_s(\boldsymbol{x}_n; h) \stackrel{\text{def}}{=} S(F(\boldsymbol{x}_n; h)), \tag{16}$$

to search for the global minimum in the box. Different from the confinement by some nonlinear operations as in (5), the shift operator never changes the nature of fixed points in their neighborhood; therefore, the property of local minima in the system (16) is the same as that of (9).

Keeping trajectories by the shift operator (15) also plays a significant role: it can change chaotic trajectories in (9) into convergent ones in (16). The shift operator has the possibility to map points on chaotic trajectories to those on stable ones. When the globally optimal solution $\boldsymbol{x}^*$ of the problem (1) is forced to be the only stable fixed point of (9) in the box for a certain h, it must be the unique stable fixed point of (16). Therefore, it is expected that, under the circumstance, while

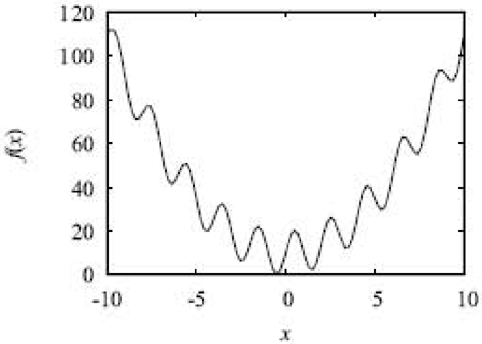


Fig. 1. Objective function of the problem (17).

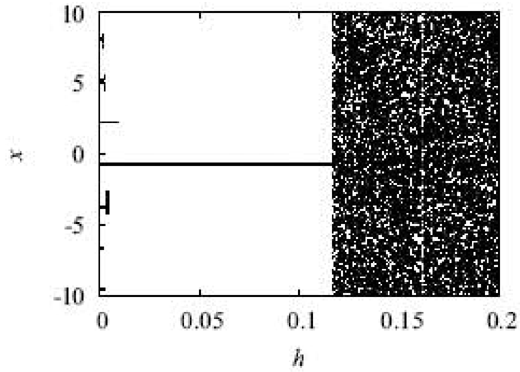


Fig. 2. Bifurcation diagram of the composite map system (16) for the problem (17). Dots show passed points by trajectories starting from $x_0 = -9.5, -9.0, \cdots, 9.0, 9.5$ with step length h given in the range $(0, 0.2)$. For each trajectory, passed points between $9900 \leq n \leq 10\,000$ are only shown to see the quasi-static state of the system. All generated trajectories with h between about 0.01 and 0.12 converge to the global optimum: $x^* = -0.7480$.

any trajectory in (16) may wander in a chaotic manner at first, it converges to the global minimum x^* in the long run.

So far, the composite map system (16) has inhered the ability to search globally at first and locally in the end, which is required in the CGO framework. Accordingly, the CA needs not to be applied any more.

D. Demonstration by a 1-dimensional problem

The proposed method is applied to solve a 1-dimensional problem

$$\min_{x} \ f(x) = x^2 + 10\sin(2x) + 10, \quad (17a)$$

$$\text{s.t.} \quad -10 \leq x \leq 10. \quad (17b)$$

In this example, how the proposed method works well is shown by the property of the composite map F_s and the trajectories of (16). Fig. 1 depicts the structure of the objective function. The function f has 10 local minima. Its global minimum is attained at $x^* = -0.7480$, and the minimum value is $f(x^*) = 0.5875$.

In this demonstration, the minimum value of f is assumed to be known in advance. Then, the following stability control function is used.

$$c(f(x)) = \max\{f(x), 0\}. \quad (18)$$

The composite map corresponding to (16) is constructed easily by computing values of the objective function and its first order derivative. Fig. 2 shows the bifurcation diagram of the composite map system. In the theory of nonlinear dynamical systems, bifurcation diagrams are widely used to see the quasi-static state of a system with respect to its control parameters. In the figure, dots show passed points by trajectories starting from $x_0 = -9.5, -9.0, \cdots, 9.0, 9.5$ with the step length $h \in (0, 0.2)$. For each trajectory, passed points only between $9\,900 \leq n \leq 10\,000$ are shown. Obviously, all the generated trajectories with h between about 0.01 and 0.12 converge to the global optimum.

As shown in Fig. 3, convergence of trajectories to the global optimum is also verified by the iterated map of $F_s(x; h)$ for $h = 0.05$. $F_s^m(x; h)$ for $m = 1, 2, 5, 10, 100, 1000$ are depicted. Uniformly located

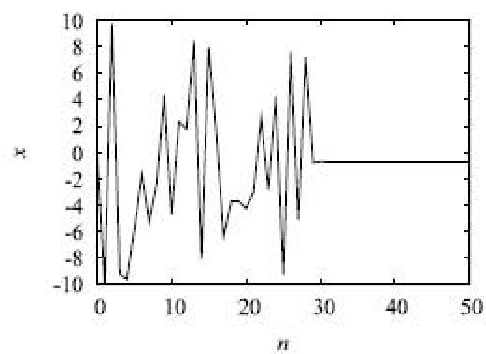


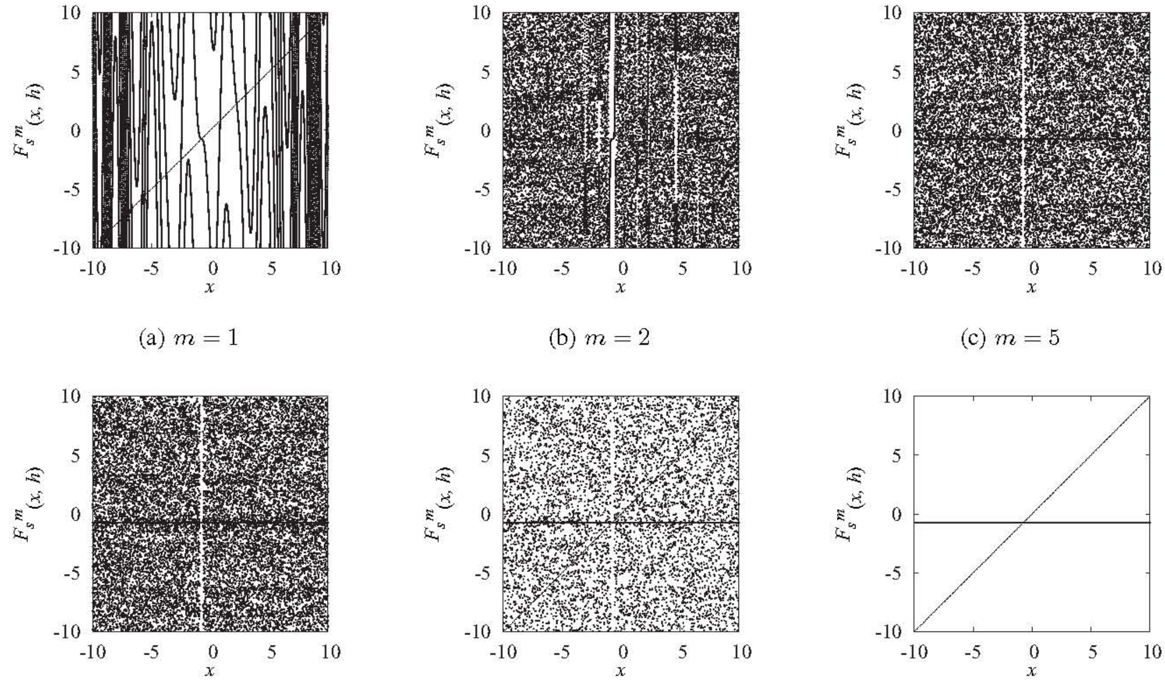
Fig. 4. A trajectory of (16) starting from $x_0 = 0$ with $h = 0.05$. It wanders at first, but finally converges to the global minimum: $x^* = -0.7480$.

points $-9.999, -9.998, \cdots, 9.998, 9.999$ are used as the argument x of the iterated map. From (a), the global minimum x^* is the unique stable fixed point of $F_s(x; h)$ because $F_s(x^*; h) = x^*$ and $|F_s'(x^*, h)| < 1$. Figures show that all points are mapped to the global minimum as the iteration number m increases.

Fig. 4 shows a representative trajectory of (16) which started from $x_0 = 0$ with $h = 0.05$. It wandered at first, but finally converged to the global minimum: $x^* = -0.7480$.

III. Adaptation of Stability Control Function

In practical optimization, it is often the case that the minimum value of the objective function is not known in advance. This section proposes an adaptation method for the stability control function by altering its lower bound and the target value of the objective function so as to find better



(a) $m = 1$ (b) $m = 2$ (c) $m = 5$

(d) $m = 10$ (e) $m = 100$ (f) $m = 1000$

Fig. 3. Iterated map of $F_s(x;h)$ for $h = 0.05$. Graphs show $F_s^m(x;h)$ for $m = 1, 2, 5, 10, 100, 1000$. Uniformly located points $-9.999, -9.998, \cdots, 9.998, 9.999$ are used as the argument x of F_s. (a) The map is equivalent to $F_s(x;h)$. The global minimum x^* is the unique stable fixed point because $F_s(x^*;h) = x^*$ and $|F_s'(x^*,h)| < 1$; (b) $F_s^2(x;h)$ maps most of points quite differently, except that points close to x^* are mapped into the neighborhood of x^*; (c) most of points are mapped quite differently as (b), but some points different from x^* are mapped into the neighborhood of x^*. It is still true that points close to x^* are mapped into the neighborhood of x^*; (d), (e) Points are mapped to x^* more frequently as the iteration number m increases; (f) All points are mapped to x^* in the end.

solutions.

A. Basic Ideas for Adaptation

First of all, now that the stability of local optima can be manipulated by the stability control function, the step length h in (16) can be fixed. Let us set $h = 1$ simply.

Second, whenever the best solution obtained so far, $\tilde{\boldsymbol{x}}$ satisfying $f(\tilde{\boldsymbol{x}}) = \tilde{f}$, is available, stable fixed points of (16) with the value of the objective function less than $\tilde{f}$ are to be found. In this context, a modified stability control function, parametrized by $\tilde{f}$ and ϵ:

$$c_A(f(\boldsymbol{x});\ \tilde{f}, \epsilon) = \begin{cases} f(\boldsymbol{x}) - \tilde{f} + \epsilon, & f(\boldsymbol{x}) \geq \tilde{f}, \\ \epsilon, & f(\boldsymbol{x}) < \tilde{f}, \end{cases} \tag{19}$$

where $\epsilon > 0$ is sufficiently small, will be used. After this, the system with the fixed step length $h = 1$ and the adaptive stability control function (19) is expressed by

$$\boldsymbol{x}_{n+1} = F_A(\boldsymbol{x}_n;\ \tilde{f}, \epsilon) \stackrel{\text{def}}{=} \boldsymbol{x}_n - c_A(f(\boldsymbol{x});\ \tilde{f}, \epsilon)\nabla f(\boldsymbol{x}_n). \tag{20}$$

In (20), local optima with the value of f less than $\tilde{f} - \epsilon$ are stabilized equally; contrarily, local optima with the value of f greater than the value are made made more unstable as f increases. Because the function evaluation has already been done at each step n, the value $\tilde{f}$ can be easily updated at the same time. As the trajectory may wander globally, it is also possible that $\tilde{f}$ is improved greatly during the global search.

Third, when the trajectory keeps wandering without reaching any local minima for a certain period, the system will be stabilized in the hope that the trajectory converges to a better solution. Now, the parameter ϵ will be decreased geometrically according to the repeated non-convergence. In this process, only local optima with the value of f less than $\tilde{f} - \epsilon$ are stabilized further as ϵ gets small.

Fourth, on the other hand, even if the tracked trajectory converged to a certain point, it may be a local minimum or even a non-optimal point. To avoid premature convergence, the best solution obtained so far will be forced to be unstable and the search for better solutions is continued. Now, instead of $\tilde{f}$ in (19), a virtual target value of the objective function, $\hat{f}$, which is smaller than $\tilde{f}$, is introduced. It will be given by

$$\hat{f} = \tilde{f} - b \cdot \epsilon, \quad b > 0, \tag{21}$$

where ϵ is the small value commonly used in (19). The target value $\hat{f}$ will be reduced by updating b until $\tilde{\boldsymbol{x}}$ becomes a repelling point. In practice, a trajectory starting from a near point from $\tilde{\boldsymbol{x}}$ is generated in the system (20) where $\hat{f}$ is substituted for $\tilde{f}$. If the trajectory does not go away from the neighborhood of $\tilde{\boldsymbol{x}}$ within a certain period, b is increased geometrically so that $\hat{\boldsymbol{x}}$ becomes unstable. Conversely, if $\tilde{\boldsymbol{x}}$ actually became unstable, the parameter ϵ in (20) is increased once as follows:

$$\epsilon := \tilde{f} - \hat{f} - \epsilon, \tag{22}$$

so that $\tilde{\boldsymbol{x}}$ is kept unstable for the trajectory to search for better solutions globally. Now that the trajectory will not converge to a certain point, the parameter ϵ will be decreased again because of the third idea mentioned above; however, when a better solution is found during the global search, the trajectory is expected to converge to a different point from the past ones.

Finally, the termination condition of the search is considered briefly. In the proposed method, the search will be terminated based on the predefined number of iterations. Since it repeats the local and global search alternatively, but since the global optimality condition cannot be checked without knowing the accurate minimum value of the objective function, f^*, the above method may be most practical at this time.

B. Algorithmic Description

Algorithm 1 describes the pseudo-code of the proposed method for chaotic global optimization by adaptively controlling the stability of local minima. In the algorithm, the variable *conv* shows whether the tracked trajectory converged or not: $conv = 1$ if it did converge, while $conv = 0$ if not. The variable *count* records the continuously repeated count of happening $conv = 0$. When $conv = 1$, the passed point of the tracked trajectory at $n = kT$, denoted by $\boldsymbol{x}_{kT}$, is slightly moved from $\tilde{\boldsymbol{x}}$ so that the trajectory may escape from it. In consideration of the box constraint (1b), $\boldsymbol{x}_{kT}$ is given by

$$x_{kT,i} = \begin{cases} \tilde{x}_i + MIN, & x_i \le (l_i + u_i)/2, \\ \tilde{x}_i - MIN, & \text{otherwise}, \end{cases} \quad i = 1, \cdots, N. \tag{23}$$

With the use of (23), Algorithm 1 is completely deterministic to have reproductivity and reliability of results. It is easily verified that the parameters ϵ and $\hat{f}$ for the stability control function can actually adapt as considered in III-A.

C. Demonstration to 1-dimensional Problem

In this subsection, the problem (17) is solved by Algorithm 1. Now, it is important that the structure of the objective function (17a), including its minimum value $f^* = f(x^*)$ and second order derivative at the global minimum $d^2 f(x^*)/dx$, is not known in advance. The parameters of Algorithm 1 is set as follows: $R_\epsilon = 10$, $R_f = 2$ and $MIN = 10^{-6}$. Tested pairs of $(T,\ K)$ are $(5,\ 200)$, $(10,\ 100)$, $(100,\ 10)$, $(5,\ 2\,000)$, $(10,\ 1\,000)$ and $(100,\ 100)$. The product $T \times K$ gives the total number of function value evaluations.

Table I shows the convergence rates to the global minimum x^* to an accuracy of less than 10^{-4} in $10\,000$ trials starting from uniform random points in the box (1b). The rates are low for three cases where $T \times K = 1\,000$, while they get significantly higher as function value evaluations are increased

Algorithm 1 Chaotic global optimization by adaptively controlling the stability of local minima.

// parameter settings
set the following predefined values:
- $T > 0$ (length of a period)
- $K > 0$ (maximum number of periods)
- $R_\epsilon > 1$ (scaling factor for ϵ)
- $R_f > 1$ (scaling factor for $\hat{f}$)
- MIN (infinitesimal number)

// initial settings
$\boldsymbol{x}_0 \leftarrow$ a point in the box (1b)
$\tilde{\boldsymbol{x}} \leftarrow \boldsymbol{x}_0$
$\tilde{f} \leftarrow f(\tilde{\boldsymbol{x}})$
$\epsilon \leftarrow 1$
$conv \leftarrow 0$
$count \leftarrow 0$
// search for better solutions
for $k = 0$ to $K - 1$ **do**
  if $conv = 0$ **then**
    // stabilize the system (20) for local search
    $\epsilon \leftarrow \epsilon / R_\epsilon$ // decrease ϵ
    for $t = 0$ to $T - 1$ **do**
      $\boldsymbol{x}_{kT+t+1} \leftarrow F_A(\boldsymbol{x}_{kT+t};\ \tilde{f}, \epsilon)$
      if $f(\boldsymbol{x}_{kT+t+1}) < \tilde{f}$ **then**
        $\tilde{\boldsymbol{x}} \leftarrow \boldsymbol{x}_{kT+t+1}$
        $\tilde{f} \leftarrow f(\tilde{\boldsymbol{x}})$
      end if
    end for
    if $||\boldsymbol{x}_{(k+1)T} - \boldsymbol{x}_{(k+1)T-1}|| < MIN$ **then**
      $conv \leftarrow 1$
      $count \leftarrow 1$
    end if
  else
    // destabilize the system (20) for global search
    $\hat{f} \leftarrow \tilde{f} - R_f^{count-1} \cdot \epsilon$ // set virtual target value of f
    $\boldsymbol{x}_{kT} \leftarrow$ a point in the neighborhood of $\tilde{\boldsymbol{x}}$
    for $t = 0$ to $T - 1$ **do**
      $\boldsymbol{x}_{kT+t+1} \leftarrow F_A(\boldsymbol{x}_{kT+t};\ \hat{f}, \epsilon)$
      if $f(\boldsymbol{x}_{kT+t+1}) < \tilde{f}$ **then**
        $\tilde{\boldsymbol{x}} \leftarrow \boldsymbol{x}_{kT+t+1}$
        $\tilde{f} \leftarrow f(\tilde{\boldsymbol{x}})$
      end if
    end for
    if $||\boldsymbol{x}_{(k+1)T} - \boldsymbol{x}_{(k+1)T-1}|| < MIN$ **then**
      $count \leftarrow count + 1$
    else
      $\epsilon \leftarrow \tilde{f} - \hat{f} - \epsilon$
      $conv \leftarrow 0$
      $count \leftarrow 0$
    end if
  end if
end for
// termination
output $\tilde{\boldsymbol{x}}$ and $\tilde{f}$

TABLE I
CONVERGENCE RATES TO GLOBAL MINIMUM FOR PROBLEM (17).

period length (T)	maximum periods (K)	convergence rate (in %)
5	200	37.93
10	100	39.41
100	10	57.86
5	2000	57.11
10	1000	74.28
100	100	**100.00**

Note: (i) Algorithm 1 was used with the parameters: $R_\epsilon = 10$, $R_f = 2$ and $MIN = 10^{-6}$. (ii) The convergence rate is to the global minimum $x^* = -0.7480$ to an accuracy of less than 10^{-4} in 10000 trials from uniform random points in the box (1b).

up to $T \times K = 10\,000$. However, solutions near the global optimum were obtained at higher rates in the results; for example, the convergence rates to the global minimum x^* to an accuracy of less than 10^{-2} in the same trials as in Table I were 62.53%, 77.07% and 85.17% for $(T,\ K) = (5,\ 200)$, $(10,\ 100)$ and $(100,\ 10)$, respectively. From the results, small values of the period length T pose insufficient convergence of trajectories; therefore, T must be set to an enough large number to successfully search for local minima with high accuracy.

On the other hand, another reason for low convergence rates to the global minimum, in particular for the cases with the smaller number of function value evaluations, is the trap to locally optimal solutions. It was resolved to a large extent by increasing the maximum number of periods, K. Fig. 5 shows the passed points of a trajectory, x_n, and the history of the best solution $\tilde{x}$ found by tracking the trajectory; Fig. 6 shows the history of the parameter ϵ in the same search. The period T in Algorithm 1 was set to 100 in this trial.

In Fig. 5, the best solution so far was updated some times until the trajectory converged to the global minimum. Afterward, the point was destabilized and finally became unstable; then, the trajectory started wandering as expected. Because the global minimum had already been found in this trial, the trajectory came back there again after the stabilization of the best solution obtained so far. However, if there had been better solutions and the best solution $\tilde{x}$ had been updated during the global search, the trajectory would have converged to another optimal solution better than those found in the past.

In Fig. 6, the parameter ϵ was reduced at first, but increased once at about $n = 600$. It was because the update defined by (22) was done at that time. Afterward, the trajectory started wandering again, then converged to the global minimum as ϵ was decreased sufficiently.

It should also be noted in Table I that the convergence rate to the global minimum reached 100% for $(T,\ K) = (100,\ 100)$. Although further investigations may be required, the parameter setting can be thought to be effective in balancing between exploration and exploitation: for the test problem, $T = 100$ was sufficiently large for local search with high accuracy, and $K = 100$ was also sufficiently large for global search so as to escape from local minima and find the global minimum constantly.

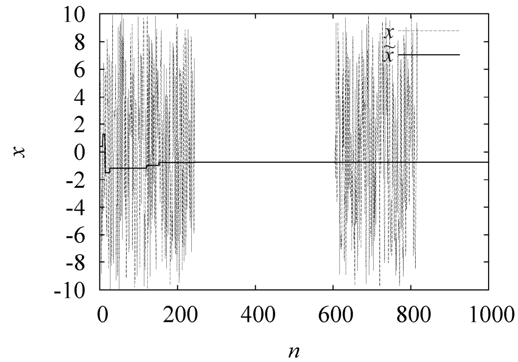


Fig. 5. Passed points by a trajectory (dashed line), x_n, and the history of the best solution (solid line), $\tilde{x}$. The period T was set to 100 in this trial. The best solution varied frequently until about $n < 150$. It stayed at 0.3988, 1.2845, -1.4965, -1.1775 and -0.9773; then, it moved to -0.7651 before approaching the global minimum $x^* = -0.7480$ at about $n = 250$. After the convergence, the global minimum made unstable at about $n = 600$ so that the trajectory was wandering globally until about $n = 820$. Thereafter, the trajectory came back to the global minimum again.

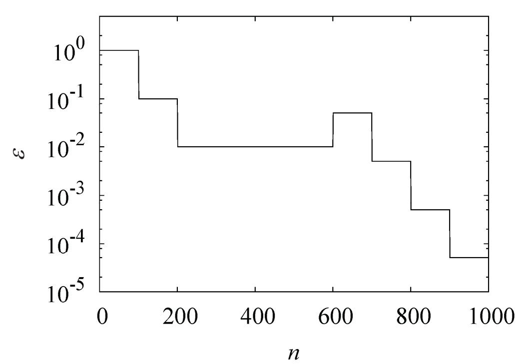


Fig. 6. Change of the parameter ϵ for the same trial as shown in Fig. 5. The parameter value was reduced at first, but increased once about $n = 600$, which helped the trajectory start wandering again. Afterward, it was reduced so that the trajectory went back to the global minimum.

IV. CONCLUSION

A novel chaotic global optimization (CGO) method was proposed. For a nonlinear continuous optimization problem with box constraints, a gradient-based system in which the stability of local minima can be manipulated by the value of objective function at the points was introduced. A composite map with the confinement of the trajectories of the above gradient-based system into the box in a simple but deterministic manner, was also proposed. Property of such systems was also studied in the context of nonlinear systems. In addition, an adaptive method of directly controlling the stability of local optima so as to improve the applicability of the proposed CGO method was also proposed. Demonstrations of the proposed method by solving a 1-dimensional problem were also provided. It was confirmed that for the test problem, all trajectories in the composite system could converge to the global optimum.

This is a pioneer study for the CGO without using the conventional CA method, so there are many issues to be challenged in the future. First, the proposed method will be tested by solved various problems, including complex and large-scale ones. Second, the theory of the adaptive stability control can be refined so as to guarantee that the globally optimal solutions are always obtained for a certain class of problem, or at least to improve the possibility to do so. Third, if the above issue is solved positively, the dependency of the presented algorithm on its parameters will also be studied comprehensively.

REFERENCES

[1] K. Aihara, T. Tanabe and M. Toyoda, "Chaotic Neural Networks," *Physics Letters A*, vol. 144, no. 6–7, pp. 333–340, 1990.

[2] L. Chen and K. Aihara, "Chaotic Simulated Annealing by a Neural Network Model with Transient Chaos," *Neural Networks*, vol. 8, no. 6, pp. 915–930, 1995.

[3] C-. S. Zhou and T-. L. Chen, "Chaotic Annealing for Optimization," *Physical Review E*, vol. 55, no. 3, pp. 2580–2587, 1997.

[4] T. Kwok and K. A. Smith, "A Unified Framework for Chaotic Neural-Network Approaches to Combinatorial Optimization," *IEEE. Trans. Neural Networks*, vol. 10, no. 4, pp. 978–981, 1999.

[5] S. Kirkpatrick, C. D. Geratt, Jr. and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[6] M. Yamaguti and H. Matano, "Euler's Finite Difference Scheme and Chaos," *Proc. Japan Acad.*, vol. 55 (A), no. 3, pp. 78–80, 1979.

[7] M. Hata, "Euler's Finite Difference Scheme and Chaos in R^n," *Proc. Japan Acad.*, vol. 58 (A), no. 5, pp. 178–181, 1982.

[8] R. C. Hilborn, "Chaos and Nonlinear Dynamics," Oxford Univ. Press, 1994.

[9] S. H. Strogatz, "Nonlinear Dynamics and Chaos," Westview Press, 19994.