AN EFFICIENT GRADIENT FORCASTING SEARCH METHOD UTILIZING THE DISCRETE DIFFERENCE EQUATION PREDICTION MODEL

Chih-Ming Chen and Hahn-Ming Lee

Department of Electronic Engineering
National Taiwan University of Science and Technology, Taipei, Taiwan

E-mail: D8602002@mail.ntust.edu.tw

Abstract

Optimization theory and method are very important in numerous different domains of engineering design and applications. Compared to many previously proposed search methods, the gradient descent method is simple and widely employed to solve numerous different optimization problems. However, the gradient descent method is easily trapped into a local minimum and converges slowly. A Gradient Forecasting Search Method (GFSM) for improving the performance of the gradient descent method for resolving optimization problems is proposed herein.

The GFSM is based on the gradient descent method and on the universal Discrete Difference Equation Prediction Model (DDEPM) proposed herein. The concept of the universal DDEPM is derived from the grey prediction model. The original grey prediction model employs mathematical hypothesis and approximation to transform a continuous differential equation into a discrete difference equation. This is not a logical approach because the forecasting sequence data is invariably of discrete type. To construct a more precise prediction model, this work adopts a discrete difference equation. The GFSM proposed herein can accurately predict the precise searching direction and trend of the gradient descent method via the universal DDEPM and can adjust prediction steps dynamically using the golden section search algorithm.

Experimental results indicate that the proposed method can accelerate the searching speed of gradient descent method and can help the gradient descent method escape from local minima. Our results further demonstrate that applying the golden section search method to achieve dynamic prediction steps of the DDEPM is an efficient approach for this search algorithm.

1. Introduction

Pioneered by Cauchy in 1847 [1], [2], the gradient descent search method has been widely applied to solve optimization problems in various different domains of engineering [1], [2]. For example, the gradient descent method can be applied to determine the near optimal solution of control parameters or to assist a fuzzy logic controller design [3], [4]. Furthermore, the gradient descent method is often used in expert systems design to tune fuzzy rules for automatically organizing the fuzzy rule base [5]. Furthermore, the fuzzy c-mean clustering algorithm also uses the gradient descent method to minimize the objective function [6]. Also, the Back-Propagation (BP) algorithm, most widely used in neural networks, is a gradient-based training algorithm [7]. However, the gradient descent method is easily trapped into a local minimum and converges slowly [1], [2], [7]. Thus, there is a need for the speed of the method to be accelerated and the ability of escaping from local minima to be increased. In this manner, many algorithms based on the gradient descent method can be improved.

Previous research reveals that the gradient descent method is the best local searching strategy when the contours of the function being searched are circular [1], [2]. In this case, the negative gradient direction points directly toward the minimal solutions. However, for most nonlinear functions, negative gradient is generally not a global searching direction [1], [2]. Numerous studies have proposed modified gradient search methods, such as Newton's method [1], [2], the conjugate gradient method [1], [2], [8], natural gradient adaptation [9], [10], the gradient descent method with momentum [11], and so on. If the searching behavior is analyzed according to these well-known search methods, the searching process can be found to be partitioned into two steps: first, to determine the searching direction, and second, to conduct an optimal line search according to the searching direction. In this work, we present a new searching direction called "gradient forecasting direction" to forecast the trend and direction of the gradient descent method by recording five gradient descent historically searched points. This new search method can effectively accelerate the searching speed of the gradient descent method and help it escape from local minima.

Generally, the learning rate of the gradient descent method is the only parameter requiring tuning, i.e. the largest influence on solution quality is the learning rate [1], [2], [7]. To avoid the phenomenon of oscillation and divergence, a comparatively small learning rate is usually set when searching for solutions in most applications. However, a smaller learning rate is the main factor in reducing searching speed and trapping the gradient descent method into local minima [1], [2], [7]. In applying the gradient descent method, many researchers have previously attempted to improve it. In neural networks, Jacobs proposed a momentum method to improve the well-known Back-Propagation algorithm in 1988 [11]. The momentum method records a previous modified value of weight, and adds it to the present weight value to obtain a new weight value. This approach can effectively accelerate the convergence speed of the Back-Propagation algorithm;

however, it can not guarantee to escape from local minima and thus obtain a better solution. Cesa-Bianchi and Nicolo [12] proposed the exponentiated gradient method, which is based on the gradient of the loss function for solving on-line regression problems. To improve the learning process and provide a global search capability for the gradient descent method, Ng, Leung et al. proposed a learning algorithm which embeds genetic search into the gradient descent algorithm [13]. These methods can effectively improve solution quality, but the searching speed is slower than the original gradient descent method. Thus, this work attempts to propose a GFSM to improve the original gradient descent method. The new search method can not only accelerate the searching speed of the gradient descent method but can also help the gradient descent method to escape from local minima.

In the proposed method, accurate prediction is critical because the proposed search method must precisely forecast the trend and direction of the gradient descent method according to some previously searched data. Hence, a precise prediction model is proposed, named the universal Discrete Difference Equation Prediction Model, as a forecasting tool. This method is a kind of time series prediction model and is derived from the grey prediction model [14], [15]. The original grey prediction model uses mathematical hypothesis and approximation to transform a continuous differential equation into a discrete difference equation. This is not a logical approach because the forecasting sequence data is always of discrete type. For some dynamic time series forecasting problems, later experiments show that its forecasting precision is insufficient. Thus, this work proposes a GFSM based on the discrete difference equation to improve the searching process of the gradient descent method. The concept of short-term memory is incorporated into the algorithm to record searched data during the searching process, and then the discrete difference equation is used to predict the direction and trend of the solution space.

The golden section search method [16], [17] is also applied to determine dynamically the near optimal prediction step. The golden section search method, an efficient single-variable search algorithm, has been widely applied in many engineering applications such as golden ratio scheduling for flow control in computer networks [18], and determining optimal shunt capacitor value at nonsinusoidal busbars [19]. Experimental results indicate that using the golden section search method to obtain dynamic prediction steps of the DDEPM is an efficient approach for this search algorithm. Our results further demonstrate that the proposed GFSM algorithm can effectively accelerate the searching speed of the gradient descent method, and can help the gradient-descent method escape from local minima.

2. The Universal Discrete Difference Equation Prediction Model (DDEPM)

In this section, a better prediction model than the grey prediction model, the universal Discrete Difference Equation Prediction Model (DDEPM), is proposed for forecasting discrete sequence. It is derived from the grey prediction model [14], [15].

2.1 Derivation of DDEPM

The proposed method employs the properties of the discrete difference equation to develop a new forecasting model for any discrete sequences. Figure 1 presents the configuration of the proposed model, where $x^{(0)}$ denotes original sequence and $\hat{x}^{(0)}$ signifies the predicted value. The central aim of Accumulated Generating Operation (AGO) is to preprocess the original sequence to achieve an exponential increasing sequence. Generally, applying Accumulated Generating Operation once (1-AGO) is sufficient to achieve an exponential increasing sequence for the original input sequence. Then, a second-order Discrete Difference Equation of single variable (DDE(2,1)) is constructed to approximate the sequence of 1-AGO. Via this method, some unknown data can be forecast. Because the DDE(2,1) is based on the AGO operation, the Inverse Accumulated Generating Operation (IAGO) must be taken to restore the AGO operation and obtain the predicted value.

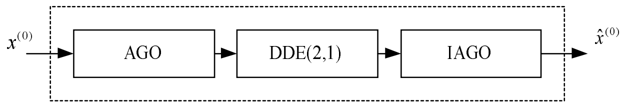


Fig. 1. The Discrete Difference Equation Prediction Model

The operation of DDEPM can be detailed as follows:

(1) Gather n original sequence data.

$$x^{(0)}=\{x^{(0)}(1), x^{(0)}(2), x^{(0)}(3),..., x^{(0)}(n)\} \quad n \in Z \tag{1}$$

where $x^{(0)}$ represents a set of n original sample data, and $x^{(0)}(n)$ is the n^{th} sample data.

(2) Apply Accumulated Generating Operation once (1-AGO) by equation (2).

Let the set $x^{(1)}=\{ x^{(1)}(1), x^{(1)}(2), x^{(1)}(3),..., x^{(1)}(n)\} \quad n\in Z$ (2)

where $x^{(1)}(p)=\sum_{i=1}^{p} x^{(0)}(i), \quad p=1,2,\ldots,n$

(3) Build a second-order discrete difference equation shown as equation (3) to approximate the sequence of 1-AGO.

Use the second-order discrete difference equation of single variable, shown as equation (3), to build the DDE(2,1).

$$x^{(1)}(p+2)+a\cdot x^{(1)}(p+1)+b\cdot x^{(1)}(p)=0 \ , \tag{3}$$

where the a and b are undecided coefficients of the second-order discrete difference equation of single variable and p is an integer.

In order to evaluate the coefficients a and b, the linear least square estimation [20] is applied to determine the two undecided coefficients. In this way, the equation (3) can be rewritten as:

$$\left[-x^{(1)}(p+1) \quad -x^{(1)}(p)\right]\begin{bmatrix}a\\b\end{bmatrix}=\left[x^{(1)}(p+2)\right]. \tag{4}$$

Let $p=1,2,...,n-2$, then equation (4) will be:

$$\begin{bmatrix}-x^{(1)}(2) & -x^{(1)}(1)\\ -x^{(1)}(3) & -x^{(1)}(2)\\ . & .\\ . & .\\ . & .\\ -x^{(1)}(n-1) & -x^{(1)}(n-2)\end{bmatrix}\begin{bmatrix}a\\b\end{bmatrix}=\begin{bmatrix}x^{(1)}(3)\\x^{(1)}(4)\\ .\\ .\\ .\\ x^{(1)}(n)\end{bmatrix}. \tag{5}$$

From equation (5), we can let

$$Y=\begin{bmatrix}x^{(1)}(3)\\x^{(1)}(4)\\ \vdots\\ x^{(1)}(n)\end{bmatrix}_{(n-2)\times 1},\quad X=\begin{bmatrix}-x^{(1)}(2) & -x^{(1)}(1)\\ -x^{(1)}(3) & -x^{(1)}(2)\\ \vdots & \vdots\\ -x^{(1)}(n-1) & -x^{(1)}(n-2)\end{bmatrix}_{(n-2)\times 2},\quad \Theta=\begin{bmatrix}a\\b\end{bmatrix}_{2\times 1}.$$

Finally, we can use the linear least square estimation [20] to get the estimated parameters Θ as equation (6).

$$\Theta=\begin{bmatrix}a\\b\end{bmatrix}=\left(X^TX\right)^{-1}X^TY. \tag{6}$$

(4) Solve the second-order discrete difference equation.

Use equation (3) and let $x^{(1)}(p)=r^p$, we can have:

$$r^{p+2}+a\cdot r^{p+1}+b\cdot r^p=0 \tag{8}$$

$$r^p(r^2+a\cdot r+b)=0$$

Thus, we can derive:

$$r_1=\frac{-a+\sqrt{a^2-4b}}{2},\quad r_2=\frac{-a-\sqrt{a^2-4b}}{2} \tag{9}$$

where r_1 and r_2 are both roots of equation (3).

Case 1: If $r_1\neq r_2$, then we can obtain the solution's formula of the second-order discrete difference equation as:

$$x^{(1)}(p)=C_1\cdot r_1{}^p+C_2\cdot r_2{}^p \tag{10}$$

Also, from the initial conditions (i.e. let $p=1$, $p=2$), we can obtain two equations as equation (11) and equation (12).

$$x^{(1)}(1)=x^{(0)}(1)=C_1\cdot r_1+C_2\cdot r_2 \tag{11}$$

$$x^{(1)}(2)=x^{(0)}(1)+x^{(0)}(2)=C_1\cdot r_1{}^2+C_2\cdot r_2{}^2 \tag{12}$$

Solving the two equations, we can have the constants, C_1 and C_2, as follows:

$$C_1=\frac{r_1\cdot x^{(0)}(1)-x^{(0)}(1)-x^{(0)}(2)}{r_1\cdot r_2-r_2{}^2},$$

$$C_2=\frac{r_2\cdot x^{(0)}(1)-x^{(0)}(1)-x^{(0)}(2)}{r_1\cdot r_2-r_1{}^2}.$$

Case 2: If $r_1=r_2$, then we can obtain the solution's formula of the second-order difference equation as:

$$x^{(1)}(p)=C_1\cdot r_1{}^p+C_2\cdot p\cdot r_1{}^p \tag{13}$$

Similarly, from the initial conditions (i.e. let $p=1$, $p=2$), we can have two equations as equation (14) and equation (15):

$$x^{(1)}(1)=x^{(0)}(1)=C_1\cdot r_1+C_2\cdot r_1 \tag{14}$$

$$x^{(1)}(2)=x^{(0)}(1)+x^{(0)}(2)=C_1\cdot r_1{}^2+2C_2\cdot r_1{}^2 \tag{15}$$

Solving the two equations, we can obtain the constants, C_1 and C_2, as follows:

$$C_1=\frac{x^{(0)}(1)\times(2r_1-1)-x^{(0)}(2)}{r_1{}^2},$$

$$C_2=\frac{x^{(0)}(1)\times(1-r_1)+x^{(0)}(2)}{r_1{}^2}.$$

Case 3: If r_1 and r_2 are conjugate plural, then we can obtain the solution's formula of the second-order difference equation as:

$$x^{(1)}(p)=C_1\cdot\rho^p\cdot\sin(\phi p)+C_2\cdot\rho^p\cdot\cos(\phi p) \tag{16}$$

where $\rho=\sqrt{(-\frac{a}{2})^2+(\frac{\sqrt{4b-a^2}}{2})^2}=\sqrt{b}$,

$$\phi=\tan^{-1}(-\frac{\sqrt{4b-a^2}}{a}).$$

From the initial conditions (i.e. let $p=1$, $p=2$), we can have two equations as equation (17) and equation (18):

$$x^{(1)}(1)=x^{(0)}(1)=C_1\cdot\rho\cdot\sin\phi+C_2\cdot\rho\cdot\cos\phi \tag{17}$$

$$x^{(1)}(2)=x^{(0)}(1)+x^{(0)}(2)=C_1\cdot\rho^2\cdot\sin(2\phi)+C_2\cdot\rho^2\cdot\cos(2\phi) \tag{18}$$

Solving the two equations, we can obtain the constants, C_1 and C_2, as follows:

$$C_1=\frac{x^{(0)}(1)\cdot\rho^2\cdot\cos(2\phi)-x^{(0)}(1)\cdot\rho\cdot\cos\phi-x^{(0)}(2)\cdot\rho\cdot\cos\phi}{\rho^3(\sin\phi\cdot\cos(2\phi)-\cos\phi\cdot\sin(2\phi))}$$

$$C_2=\frac{x^{(0)}(1)\cdot\rho\cdot\sin\phi+x^{(0)}(2)\cdot\rho\cdot\sin\phi-x^{(0)}(1)\cdot\rho^2\cdot\sin(2\phi)}{\rho^3(\sin\phi\cdot\cos(2\phi)-\cos\phi\cdot\sin(2\phi))}.$$

(5) Apply Inverse Accumulated Generating Operation (IAGO).

Because the Discrete Difference Equation Prediction Model, shown as equation (3), is based on the 1-AGO numbers, we must take the Inverse Accumulated Generating Operation to restore the AGO operation and obtain the predicted value.

$$\hat{x}^{(0)}=x^{(1)}(p)-x^{(1)}(p-1) \tag{19}$$

where $\hat{x}^{(0)}$ denotes the predicted value, p is a prediction step of DDEPM.

2.2 The Universal DDEPM

Although the forecasting process of DDEPM is very simple, it suffers from a significant disadvantage, in that it can only forecast nonnegative sequences. Because the DDEPM uses the Accumulated Generating Operation

(AGO) to obtain an exponential increasing sequence, it cannot build a correct prediction model if the sequence contains both positive and negative data. Therefore, translating the original sequence into a related positive one via data mapping concepts is proposed herein. This process is termed Mapping Generating Operation (MGO). Furthermore, by applying the Inverse Mapping Generating Operation (IMGO) to restore the forecasting sequence, the prediction sequence can easily be obtained. In this manner, the proposed universal DDEPM is well defined for all sequences. That is, regardless of whether the sequence is positive or negative, data can be forecast correctly. Figure 2 illustrates the configuration of the universal DDEPM, where $x^{(0)}$ denotes original sequence and $x_p^{(0)}$ stands for the predicted value.

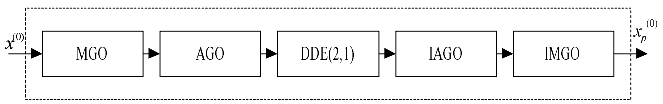


Fig. 2. The configuration of the universal DDEPM

The function f that serves as the Mapping Generating Operation must satisfy two conditions [21]: (1) f is one-to-one and continuous; (2) $f(x) \geq 0$ for all x. Herein, a linear mapping function [21] is applied as an MGO operation. The detailed operation of this function is described below:

Let $x_m^{(0)}$ represent the set of Mapping Generating Operation (MGO). The $x_m^{(0)}$ can be defined as:

$$x_m^{(0)} = \{x_m^{(0)}(1), x_m^{(0)}(2), x_m^{(0)}(3), \ldots, x_m^{(0)}(n)\}$$

$$x_m^{(0)}(i) = MGO(x^{(0)}(i)) \overset{\Delta}{=} s + \gamma \cdot x^{(0)}(i) \quad i = 1,2,3\ldots,n \quad (20)$$

where n denotes the number of original sample data, s represents a shift factor, γ stands for a scaling factor and the $x^{(0)}(i)$ is the i^{th} data of original sample sequence.

By the above definition, the Inverse Mapping Generating Operation (IMGO) can be defined as follows:

$$x_p^{(0)} = IMGO(\bar{x}^{(0)}(p)) = \frac{1}{\gamma}(\bar{x}^{(0)}(p) - s) \quad (21)$$

where $x_p^{(0)}$ denotes the final predicted value with prediction step p, $\hat{x}^{(0)}(p)$ represents the predicted value with prediction step p in IAGO process, s stands for a shift factor and γ is a scaling factor.

Obviously, the linear mapping function satisfies two conditions required by the Mapping Generating Operation. That is, these two operations, MGO and IMGO, are useful for the universal DDEPM and can overcome the disadvantage of the DDEPM.

3. The Gradient Forecasting Search Method (GFSM)

This section provides a Gradient Forecasting Search Method (GFSM) based on the universal DDEPM. This GFSM is employed to enhance the searching process of the original gradient-descent method.

3.1 The Algorithm of GFSM

First, the proposed GFSM records five historical data items generated from the original gradient descent method (experiments given later show that using 5 historically searched points can get satisfied forecasting result). Next, the modeling procedures of the universal DDEPM are followed to construct a forecasting model. This model is then employed to forecast the searching trend of the gradient descent method. After determining a prediction step by the golden section search algorithm [16], [17], the Inverse Accumulated Generating Operation is used, shown as equation (19), to obtain a new prediction point. The detailed operation of this algorithm is described below:

Step 1. Define the initial conditions.

$x^{(0)}$: initial searching point;

α : learning rate of the gradient-descent method;

ε : stop criterion (herein, the stop criterion is defined as $\left|f(x^{(k+1)}) - f(x^{(k)})\right| < \varepsilon$);

$k = 0$: iteration number.

Step 2. Execute the searching process by the gradient-descent method.

$$x^{(k+1)} = x^{(k)} - \alpha \nabla f(x^{(k)})$$

Step 3. Detect the stop criterion.

Is $\left|f(x^{(k+1)}) - f(x^{(k)})\right| < \varepsilon$?

Yes: Go to Step 7.

Step 4. Have five searched points been gathered?

No: Go to Step 2.

Step 5. Obtain a new searching point using the universal DDEPM under determining a near optimal prediction step via the golden section search algorithm.

Step 5.1. Use the five searching points obtained for the operations of MGO and 1-AGO.

Step 5.2. Use the sequence data obtained in Step 5.1 to build the universal DDEPM.

Step 5.3. Obtain a near optimal prediction step p via the golden section search algorithm.

Step 5.4. Compute the prediction value via the prediction step obtained.

Step 5.5. Obtain a new searching point using the prediction value obtained in Step 5.4 via operations of IAGO and IMGO.

Step 6. Let $k = k + 1$ go to Step 2.

Step 7. Obtain the $x^{(k+1)}$ solution and terminate this algorithm.

3.2 Determination of the Prediction Step by the Golden Section Search Algorithm

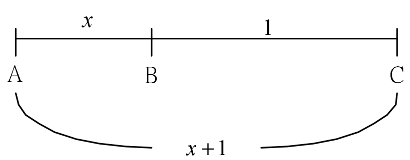
An efficient single-variable minimization routine is necessary for the proposed search algorithm because the prediction step must be determined appropriately during the searching process. In this study, the golden section search algorithm [16], [17] is employed to determine the prediction step of the GFSM search method. It is sure that the other famous line search algorithms can also be employed to determine the near optimal prediction step, such as uniform search, dichotomous search, golden section search method, Fibonacci search [22], and so on. Mokhtar S. Bazaraa et al. [22] have proven that the most efficient algorithm is Fibonacci search, followed by the golden section search method, the dichotomous search method, and finally the uniform search method. Also note that if the number of iterations is large enough, the Fibonacci search and the golden section search method are almost identical. However, the implementation of golden section search method is easier than Fibonacci search. Thus, we employ the golden section search method to determine the near optimal prediction step for GFSM. The golden section implies that a certain length is divided such that the ratio of the whole to the longer part is equal to the ratio of the longer part to the shorter part. As Fig. 3 illustrates, line $\overline{AC}$ is divided so that the ratio of $\overline{AB}$ to $\overline{BC}$ is the same as the ratio of $\overline{BC}$ to $\overline{AC}$. Let the length of longer part be 1 and the length of shorter be x, then the Eqn. (22) can be obtained by the golden section rule.

$$\frac{1}{x+1} = \frac{x}{1} \tag{22}$$

This equation can be solved to obtain the positive root:

$$x = \frac{1}{2}(\sqrt{5}-1) = 0.6180339... \approx 0.618$$

The ratio of 0.618 is the so-called golden section or divine proportion. The length of $\overline{AC}$ becomes 1.618, also known as the golden mean.



$$\frac{\overline{BC}}{\overline{AC}} = \frac{\overline{AB}}{\overline{BC}} \approx 0.618$$

Fig. 3. The golden section

Figure 4 shows the illustration of the golden section used to determine the prediction step in searching process. In this figure, assume the searching space of the prediction step is limited to the interval $[a,b]$. L denotes the width of the searching space, p signifies the prediction step and $f(p)$ is the function value under the prediction step p. Meanwhile, p_1 is the golden section point from the right-most boundary and p_2 represents another golden section point from the left-most boundary. Figure 4 illustrates that the searching interval of p_2 to b can be deleted in the first searching step because $f(p_1) < f(p_2)$. That is, the right-most searching space can be reduced from b to b_1. By the same golden section iteration, the near optimal prediction step can gradually be approached. The searching process can be terminated when the searching space gradually shrinks to satisfy the given stop criterion. Advantages of the golden section search algorithm are 38.2% of searching space can be reduced and one of two golden section points can be reserved to become the next test point [16], [17] in each searching process.

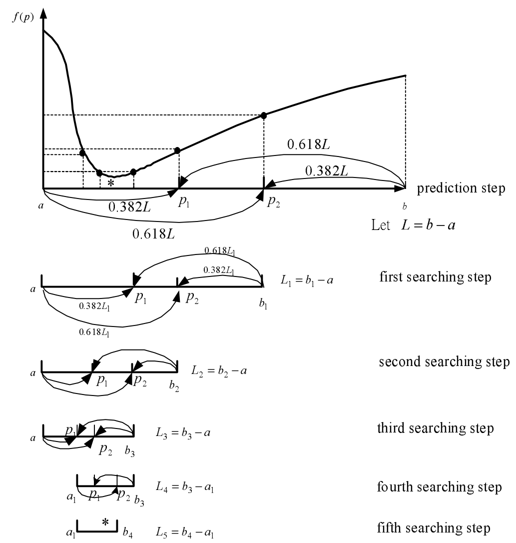


Fig. 4. The illustration of golden section searching process (* represents the optimal solution)

3.3 Comparison of gradient descent direction and gradient forecasting direction

This subsection describes how the proposed gradient forecasting direction differs from the gradient descent direction. The negative gradient descent direction is known to always be orthogonal with a contour curve in each searching process. It is a local steepest descent direction for searching minimal solutions. Figure 5 displays that the next searching direction for point x follows the negative gradient direction when seeking minimal solutions and positive gradient direction when searching for maximal solutions. In this figure, the direction d is the so-called gradient descent direction. It is the best local searching direction when the contour of the searched function is a circle, but it is generally not a global searching direction for most nonlinear functions [1], [2].

The proposed gradient forecasting direction is a trend of the gradient descent direction because some historically searched data points of the gradient descent method are recorded to predict the next searching point. Our method is to gather some local searching directions for finding the global searching direction by a forecasting mechanism. To compare these two searching processes, a simple two-variable function $f(x_1, x_2) = 8x_1^2 + 4x_1x_2 + 5x_2^2$ is employed to observe their searching processes, as shown in Figs. 6 and 7, respectively. These figures reveal that the gradient forecasting direction is a trend of the gradient descent direction and can seek the minimal solutions faster than the gradient descent search method. Furthermore, Figure 8 shows how the GFSM escaping from a local optimal can obtain a higher quality solution. In Figure 8, the five black points stand for historically searched points using the gradient descent method under some learning rate. The GFSM constructs the universal DDEPM to determine the next searching point according to these searched points. A trend solution space can be obtained by the universal DDEPM shown as the dotted line of Fig. 8. If an appropriate prediction step is determined, the local minimum shown in Figure 8 may be ignored even though a comparatively small learning rate is used. Under the same condition, the gradient descent method may trap into the local minimum in this case if a small learning rate is used. A larger learning rate might resolve this problem, but it may result in the phenomenon of oscillation or divergence. That is, the GFSM has fast searching speed and the capability of escaping from local minima even using a comparatively small learning rate. This property can show the GFSM is more robust with respect to learning rate than the other gradient-based search methods.

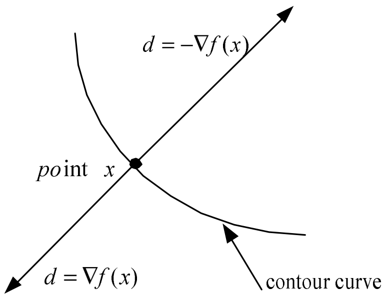


Fig. 5. Gradient descent direction

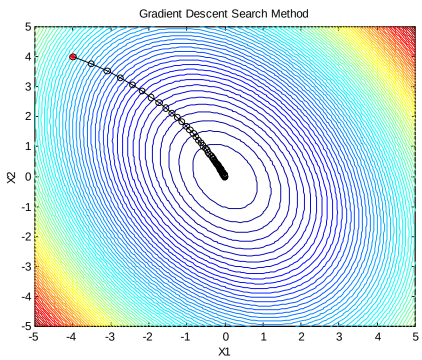


Fig. 6. Gradient descent search method

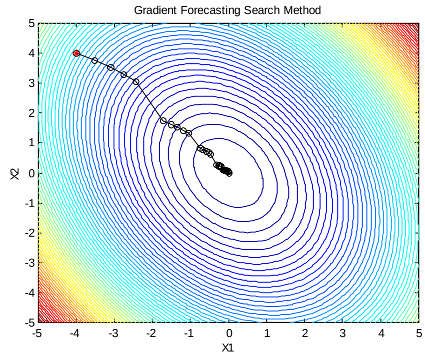


Fig. 7. Gradient forecasting search method

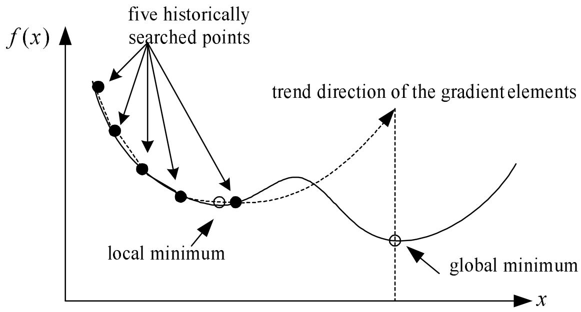


Fig. 8. The illustration of GFSM's escaping from local minima

4. Experimental Results

To illustrate the effectiveness of the proposed DDEPM and GFSM search method, time series functions are employed to compare the forecasting capability of the DDEPM model with the GM(1,1) grey prediction model. Furthermore, benchmark multi-variable functions are employed as experimental target functions to test the searching performance of the GFSM algorithm.

4.1 Time Series Forecasting

Assume that the considered time series sequence are generated by the following three functions [21]:

$f_1(t) = 3 + \cos(t) + 2\sin(3t)$,

with sample time 0.05 seconds and $0 \le t \le 10$. Here, $f_1(t)$ represents an oscillatory and bounded function. In functions $f_1(t)$, four data items are sampled in each modeling process, and these four data items are used to forecast the next four unknown data using the GM(1,1) grey prediction model and DDEPM proposed herein. After forecasting the next four unknown data items, new data are again obtained from the original time series function, and then the next four unknown data items continue to be forecast. The process is repeated in each cycle until termination. Figure 9 summarizes the forecasting results of the grey prediction model while Figure 10 illustrates the forecasting results of DDEPM. The experimental results reveal that the DDEPM has better forecasting capability than the GM(1,1) grey prediction model. Figure 9 clearly displays that the grey prediction model has poor forecasting capability at turning points. Meanwhile, in Figures 10, both the predicted and desired sequences in each sample data are very close. Clearly, the DDEPM can precisely forecast the trend of function using only a few sample data.

4.2 Rosenbrock's Function Searching

This subsection uses a benchmark Rosenbrock's function [1], [2] to explain the properties of the GFSM algorithm. The Rosenbrock's function is $f(x_1,x_2)=100(x_2-x_1^2)^2+(1-x_1)^2$. It is a well-known test function for optimal search algorithms such as gradient descent search method [1], [2], conjugate gradient method [1], [2], genetic algorithms [23], and so on. Approaching the minimal area of this function using some search algorithms is difficult because the minimal area of function value is enclosed by a long ravine which is only slightly decreasing. Restated, the function has a long ravine with very steep walls and an almost flat bottom, accounting for why many gradient descent methods fail to minimize this function.

Several well-known modified gradient search methods are compared with the GFSM on Rosenbrock's function searching, as listed in Table 1. Experimental results demonstrate the GFSM is a more robust and efficient search algorithm than other well-known tested search methods. The proposed method can not only have the fastest searching speed but also can obtain a better quality solution.

4.3 Powell's Function Searching

This subsection uses Powell's function [1] as target functions to test the performance of the proposed GFSM search method. The function is:

$$f_2(x_1,x_2,x_3,x_4)=(x_1+10x_2)^2+5(x_3-x_4)^2+(x_2-2x_3)^4+10(x_1-x_4)^4$$

Experimental results are presented as Table 2. These experiments also demonstrate that the proposed GFSM search method is a more robust and faster search algorithm than other well-known tested search methods. In Table 2, the GFSM has fastest searching speed and obtains a best quality solution on Powell's function.

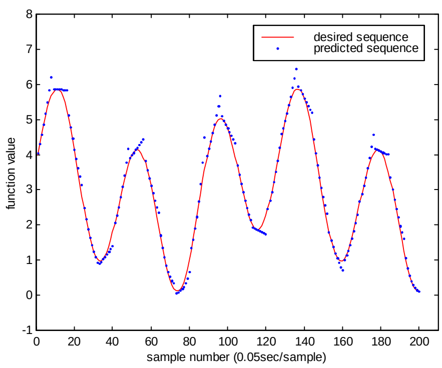


Fig. 9. Forecasting results of $f_1(t)$ by the GM(1,1) grey prediction model (The dotted and solid line respectively stand for predicted sequence of the GM(1,1) grey prediction model and desired sequence of $f_1(t)$)

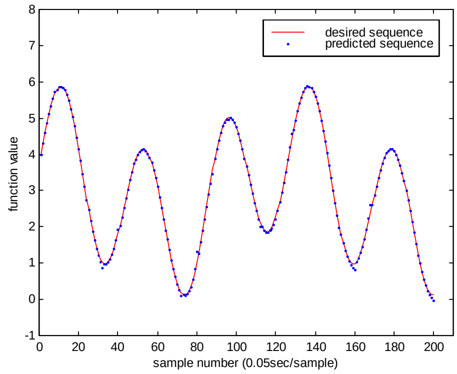


Fig. 10. Forecasting results of $f_1(t)$ by the DDEPM (The dotted and solid line respectively stand for predicted sequence of the DDEPM and desired sequence of $f_1(t)$)

Table 1. Comparison results of the different gradient search algorithms on Rosenbrock's function ($x^{(0)}=[-1.2,1.0]^T$)

Items / Method	Learning rate	Momentum term	Prediction step	Final solution	Function value	Searching epochs	CPU time
Gradient descent	0.001	---	---	$x_1=0.9965$ $x_2=0.9929$	0.000013	11911	18.6 sec
Cauchy's gradient [1], [2]	*	---	---	$x_1=0.9980$ $x_2=0.9960$	0.000005	4338	7.52 sec
Gradient with momentum [11]	0.001	0.9	---	$x_1=0.9967$ $x_2=0.9933$	0.000011	10549	16.4 sec
Conjugate gradient [1], [2]	*	---	---	$x_1=0.9976$ $x_2=0.9952$	0.000006	4270	7.25 sec
Natural gradient [9],[10]	0.001	---	---	$x_1=0.9950$ $x_2=0.9900$	0.000025	11597	19.1 sec
Natural gradient [9],[10]	*	---	---	$x_1=0.9977$ $x_2=0.9953$	0.000005	4394	12.4 sec
GFSM	0.001	---	*	$x_1=0.9979$ $x_2=0.9958$	0.000005	1322	3.02 sec

* indicates that the golden section search algorithm is used to determined a near optimal parameter

Table 2. Comparison results of the different gradient search algorithms on Powell's function ($x^{(0)}=[3,1,0,1]^T$)

Items / Method	Learning rate	Momentum term	Prediction step	Final solution	Function value	Searching epochs	CPU time
Gradient descent	0.002	---	---	$x_1=0.0467$ $x_2=-0.0047$ $x_3=0.0232$ $x_4=0.0233$	0.00001	21456	40.1 sec
Cauchy's gradient [1], [2]	*	---	---	$x_1=0.0529$ $x_2=-0.0053$ $x_3=0.0263$ $x_4=0.0234$	0.000016	22999	51.1 sec
Gradient with momentum [11]	0.002	0.1	---	$x_1=0.0460$ $x_2=-0.0046$ $x_3=0.0229$ $x_4=0.0230$	0.000009	20117	43.7 sec
Conjugate gradient [1], [2]	*	---	---	$x_1=0.0633$ $x_2=-0.0063$ $x_3=0.0315$ $x_4=0.0317$	0.000033	29999	45.8 sec
GFSM	0.002	---	*	$x_1=-0.0167$ $x_2=0.0017$ $x_3=-0.0125$ $x_4=-0.0125$	0.000001	3451	7.3 sec

* indicates that the golden section search algorithm is used to determined a near optimal parameter

5. Conclusion

This study has demonstrated that the proposed GFSM and the universal DDEPM have the following properties:

(1) The GFSM can accurately predict the direction and trend of the solution in the searching process. Also, it can predict the possible direction of the solution with appropriate prediction steps. Thus, the GFSM has a faster convergent speed than the gradient descent method and other tested well-known modified gradient algorithms, and thus produces a better quality solution.

(2) When using an appropriate prediction step, this search algorithm can effectively escape from local minima and achieve a high quality solution. The golden section search algorithm is also applied to dynamically determine prediction step.

(3) Having a higher prediction accuracy than the GM(1,1) grey prediction model for dynamic time series forecasting, the universal DDEPM does not require extensive sample data (merely 4 to 5 sample data items) to model a prediction model, and its computation is simple.

(4) The universal DDEPM is well defined for any kinds of sequence, and can accurately forecast sequences that simultaneously contain both positive and negative data.

References

[1] G. V. Reklaitis, A. Ravindran & K. M. Ragsdell, Engineering Optimization Methods and Applications, *A Wiley-Interscience Publication*, 1983.

[2] Philip E. Gill, Walter Murray and Margaret H. Wright, Practical Optimization, A *Subsidiary of Harcourt Brace Jovanovich, publishers*, 1981.

[3] Min-Shin Chen, ''Control of Linear Time-Varying Systems by The Gradient Algorithm, " *IEEE Conference on Decision & Control*, pp. 4549-4553, 1997.

[4] Sang-Ho So and Dong-Jo Park, "Design of Gradient Descent Based Self-Organizing Fuzzy Logic Controller with Dual Outputs," *IEEE International Conference on Fuzzy Systems*, Vol. 1, pp. 460-464, 1999.

[5] Yan Shi, Masaharu Mizumoto, Naoyoshi Yubazaki and Masayuki Otani, "A Learning Algorithm for Tuning Fuzzy Rules Based on the Gradient Descent Method," *IEEE International Conference on Neural Networks*, Vol. 1, pp. 55-61, 1996.

[6] Dong C. Park and Issam Dagher, "Gradient Based Fuzzy C-means (GBFCM) Algorithm," *IEEE International Conference on Fuzzy Systems*, Vol. 3, pp. 1626-1631, 1994.

[7] D. E. Rumelhart, G. E. Hiton & R. J. Williams, "Learning Internal Representation by Error Propagation, " *Parallel Distributed Processing*, Vol. 1, pp. 318-362, 1986.

[8] C. Charalambous, "Conjugate Gradient Algorithm for Efficient Training of Artificial Neural Networks," *IEEE Proceedings-G*, Vol. 139, No. 3, pp. 301-310, JUNE 1992.

[9] S. Amari and S. C. Douglas, "Why Natural Gradient?," *Proceedings of IEEE International Conference Acoust., Speech, Signal Processing*, pp. 1213-1216, 1998.

[10] Shun-ichi Amari, " Natural Gradient Works Efficiently in Learning," *Neural Computation*, Vol. 10, pp. 251-276, 1998.

[11] Jacobs, R. A., "Increased Rates of Convergence Through Learning Rate Adaptation," *Neural Networks*, Vol. 1, pp. 295-307, 1988.

[12] Cesa-Bianchi & Nicolo, ''Analysis of Two Gradient-based Algorithms for On-line Regression," *Proceedings of the 10th Annual Conference on Computational Learning Theory*, pp. 163-170, 1997.

[13] Ng, S.C., Leung S.H., Chung C.Y., Luk A. and Lau W.H., ''A New Learning Algorithm for Adaptive IIR Filtering," *IEEE Signal Processing Magazine*, Vol. 13, No. 6, pp. 38-46, 1996.

[14] J. L. Deng , "Control Problems of Grey System," *Systems & Control Letters*, Vol. 1, pp. 288-294, 1982.

[15] J. L. Deng , "Introduction to Grey System Theory," *The Journal of Grey System*, Vol. 1, pp. 1-24, 1989.

[16] Tirupathi R. Chandrupatla, "An Efficient Equdratic Fit-Sectioning Algorithm for Minimization without Derivatives," *Computer Methods in Applied Mechanics and Engineering*, Vol. 152, No. 1-2, pp. 211-217, 1998.

[17] Luc Pronzato, Henry P. Wynn and Anatoly A. Zhigljavsky, "A Generalized Golden-Section Algorithm for Line Search," *IMA Journal of Mathematical Control and Information*, Vol. 15, No. 2, pp. 185-214, 1998.

[18] Shivendra S. Panwar, Thomas K. Philips and Mon-Song Chen, "Golden Ratio Scheduling for Flow Control with Low Buffer Requirements," *IEEE Transaction on Communications*, Vol. 40, No. 4, pp. 765-772, 1992.

[19] I. M. Ei-Amin, S. O. Duffuaa and A. U. Bawah, "Optimal Shunt Compensators at Nonsinusoidal Busbars," *IEEE Transaction on Power System*, Vol. 10, No. 2, pp. 716-722, 1995.

[20] T. C. Hsia. System Identification: Least Square Methods, *University of California, Davis*.

[21] Ming-Feng Yeh, "Studies and Applications of GM(1,1) Model and Grey Relational Analysis," *Dissertation for Ph.D. Degree, Department of Electrical Engineering, Tatung Institute of Technology, Taiwan*, 1999.

[22] M. S. Bazaraa, H. D. Sherali, & C. M. Shetty, Nonlinear Programming Theory and Algorithms, *John Wiley & Sons, Inc*., 1993.

[23] David E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning," *Addison-Wesley Publishing Company, INC*, 1989.