ICIC Express Letters
Volume **17**, Number **10**, October **2023**

ICIC International ©2023 ISSN 1881-803X
pp. **1125–1133**

MULTI-POINT SEARCH BASED OPTIMIZATION ALGORITHM AND APPLICATION IN SYSTEM IDENTIFICATION

Lianming Sun[1] and Jianing Liu[2]

[1]Department of Information Systems Engineering
Faculty of Environmental Engineering
[2]Graduate School of Environmental Engineering
The University of Kitakyushu
1-1 Hibikino, Wakamatsu-ku, Kitakyushu 808-0135, Japan
sun@kitakyu-u.ac.jp; c1dca401@eng.kitakyu-u.ac.jp

Received February 2023; accepted April 2023

Abstract. *Nonlinear optimization is an essential problem in engineering applications, where the global optima are expected to be obtained through numerical optimization algorithms, and the gradient-based methods are the common choices. However, many gradient-based methods often converge to local minima, whereas the global optima cannot be guaranteed in some application problems. In this paper, a multi-point search scheme combining the global and local search techniques is investigated to improve the global convergence performance of numerical optimization. In the proposed search scheme, the candidate solutions of multiple individuals are updated parallelly, where a primary individual corresponds to point search for global optima, while the auxiliary individuals are used to find appropriate search regions and perturbations are added to make the candidate solutions escape from local minima. Moreover, the search information is shared among all the individuals. The numerical simulation examples show that the proposed algorithm has better global convergence performance than the conventional local search methods.*
Keywords: Nonlinear optimization, Gradient-based algorithm, Multi-point search, Global convergence

1. **Introduction.** Optimization is an essential issue in all engineering disciplines since many engineering problems are treated as mathematical parameter or structure optimization of loss functions. For example, in system identification [1, 2, 3, 4], signal processing [5, 6, 7], control systems [8, 9, 10], and recently notable machine learning and data-driven applications [11, 12], the system performance is often evaluated by quadratic criteria of loss functions to evaluate error signal, mathematical model structure and parameters, and then system design is executed through searching criteria's minima.

Following categories of optimization techniques are usually applied in practical applications. a) Direct search algorithms, such as simplex search method [13], and Hooke-Jeeves method [14], are the simplest choices and easily to be implemented for smooth loss functions. However, the direct search algorithms usually have slow convergence, and are hard to be applied for complicated problems. b) Gradient-based algorithms apply gradients given by analytic derivatives of loss functions with respect to the parameter vector, or by approximation of finite difference techniques to determine the parameters' changing direction [15, 16]. The existing gradient-based algorithms include linear optimization methods such as least squares (LS), nonlinear optimization methods distinguished by different choices for scaling and rotation of the gradients. The well-known gradient-based algorithms are steepest decent method (SD), Newton's method, Gauss-Newton method, quasi-Newton method, conjugate gradient method, and Levenberg-Marquardt method [1, 17]. They are

DOI: 10.24507/icicel.17.10.1125

the most common and important nonlinear optimization techniques; however, their solution may fall down into a local minimum close to the point where the optimization starts. As a result, the gradient-based algorithms are local searching schemes, which cannot guarantee global optima [1]. c) Nonlinear global optimization techniques try to find the global optima from random starts in simulated annealing (SA), chaotic method [18], or a population of individual candidates are parallelly searched in evolutionary algorithms (EA), genetic algorithms (GA), particle swarm optimization (PSO) [3, 19, 20, 21, 22], and ant colony optimization (ACO) [23]. d) Machine learning algorithms try to extract information by combination of statistical techniques and optimization methods [11]. However, the main drawback in global optimization methods and machine learning is the huge computational demand, hence some trade-off is required between the searching range for global convergence and the computational complexity, while a potential trade-off is to combine global search techniques with local approaches [1].

A multi-start local search approach has been considered as the combination of global and local search techniques, and then the number of start points and the associated convergence probability are discussed [24]. Furthermore, a multi-start based Levenberg-Marquardt method has been applied in system identification of a closed-loop system operated in a feedback controller, where the numerical condition of data matrices is so poor that it is fragile to the interference and noise terms [25]. With aid of multi-start, the performance of global convergence has been largely improved, whereas sometimes the conventional gradient-based methods still converge to local minima. On the other hand, notice that in multi-start methods, the search process from each start is commonly conducted independently, and less information on search results is shared with the searching procedure started from other points. Inspired by the idea of information sharing with the other individuals in PSO, a parallel search scheme using multi-point rather than multi-start is proposed in this paper to improve both the convergence performance and efficiency of the local search process. The highlight of this work is that among the individuals, an individual corresponds to point search while the others are used to find appropriate search regions to make the candidate solutions escape from local minima, and the region information is shared among all the individuals.

The rest of the paper is organized as follows. In the next section, the main preliminaries for local and global optimization are reviewed. In Section 3, the multi-point based optimization algorithm is proposed, and the search procedure is illustrated; furthermore, the application in system identification is discussed in Section 4. The numerical examples are illustrated, and some comparison with the other optimization algorithms to demonstrate the effectiveness of the proposed algorithm are given in Section 5. Finally, Section 6 summarizes the conclusion and the future research work to deal with more complicated problems.

2. **Preliminaries.** The common local and global optimization methods are reviewed in this section.

2.1. **Information criterion and loss function.** A criterion defines what to be optimized in mathematical description. For example, in system identification the criterion is often a quadratic loss function of error signal

$$f(\boldsymbol{x}) = \frac{1}{2(t_2 - t_1)} \sum_{t=t_1}^{t_2-1} e^2(t) \tag{1}$$

to evaluate the quality of constructed system model, where $e(t)$ is an error signal defined as the difference between the system output and the model output determined by the model parameter vector $\boldsymbol{x}$, and t is the normalized instant, while in some control systems, the error signal may be an error between the desired reference and the plant output,

and the power of control input is often included in the criterion to reduce the energy consumption. Moreover, some constraints may also be added to the criterion function, where optimization becomes constrained problem. Without loss of generality, let the loss function be denoted as $f(\boldsymbol{x})$, where optimization searches a solution $\boldsymbol{x}^*$ such that

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}). \tag{2}$$

In the nonlinear optimization problems, besides the global optima, the nonlinear functions, such as Matlab peaks function, and Goldstein-Price function shown in Figure 1, often have several local minima, whereas in most of applications the global convergence is highly desired.

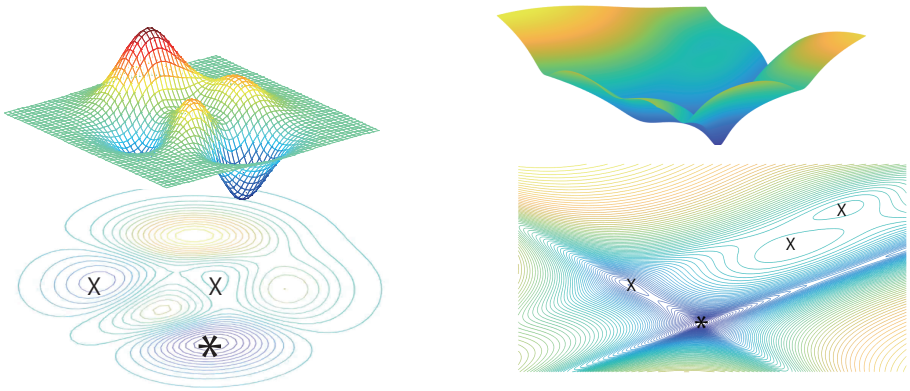

FIGURE 1. Illustration of local minima of Matlab peaks function (left) and Goldstein-Price function (right). $*$: Global minima; $\times$: Local minima.

2.2. **Gradient-based local optimization methods.** In nonlinear optimization, no analytic solution is available, and the numerical solutions are searched by iterative computation. Assume that in the kth iteration, the solution is updated from $\boldsymbol{x}_k$ to $\boldsymbol{x}_{k+1}$, while the gradient is obtained by $\boldsymbol{g}_k = f'(\boldsymbol{x}_k) = \frac{df(\boldsymbol{x})}{d\boldsymbol{x}}\Big|_{\boldsymbol{x}=\boldsymbol{x}_k}$ analytically, or approximated by finite difference techniques [15, 16]. The gradients indicate the updating directions where $\boldsymbol{g}_k \to 0$ at the optima. Then the new solution is updated by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mu_k \boldsymbol{q}_k, \ \ \boldsymbol{q}_k = \boldsymbol{R}_k \boldsymbol{g}_k, \tag{3}$$

where μ_k is a step size or a learning rate for the iteration computation, and $\boldsymbol{R}_k$ is a rotating and scaling matrix with respect to the gradient $\boldsymbol{g}_k$. $\boldsymbol{R}_k$ has different choices in the gradient-based local optimization methods. The simplest choice is $\boldsymbol{R}_k = \boldsymbol{I}$ in SD method, whereas it is defined as $\boldsymbol{R}_k = \boldsymbol{H}_k^{-1}$, where $\boldsymbol{H}_k$ is Hessian matrix calculated by the 2nd order derivative of $\boldsymbol{H}_k = f''(\boldsymbol{x}_k)$ in Newton's method, or the approximation

$$\boldsymbol{H}_k = \frac{1}{t_2 - t_1} \sum_{t=t_1}^{t_2-1} \frac{de(t)}{d\boldsymbol{x}} \left(\frac{de(t)}{d\boldsymbol{x}} \right)^T + \alpha \boldsymbol{I} \tag{4}$$

at $\boldsymbol{x} = \boldsymbol{x}_k$ corresponding to quadratic loss function of the mean square error $e^2(t)$ without a regularization term, i.e., $\alpha = 0$ in Gauss-Newton method, and with the positive regularization term $\alpha \neq 0$ in Levenberg-Marquardt method.

On the other hand, in high-dimension problems, the computational complexity is reduced by an approximation of Broyden-Fletcher-Golfarb-Shanno (BFGS) formula

$$\boldsymbol{R}_{k+1} = \left(\boldsymbol{I} - \frac{\Delta \boldsymbol{x}_k \Delta \boldsymbol{g}_k^T}{\Delta \boldsymbol{x}_k^T \Delta \boldsymbol{g}_k} \right) \boldsymbol{R}_k \left(\boldsymbol{I} - \frac{\Delta \boldsymbol{x}_k \Delta \boldsymbol{g}_k^T}{\Delta \boldsymbol{x}_k^T \Delta \boldsymbol{g}_k} \right)^T + \frac{\Delta \boldsymbol{x}_k \Delta \boldsymbol{x}_k^T}{\Delta \boldsymbol{x}_k^T \Delta \boldsymbol{g}_k}, \tag{5}$$

in qausi-Newton method, where $\Delta \boldsymbol{x}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$, $\Delta \boldsymbol{g}_k = \boldsymbol{g}_{k+1} - \boldsymbol{g}_k$ [1, 15, 17]. Moreover, the conjugate gradient method reduces the computational complexity, where the conjugate gradient $\boldsymbol{q}_k$ is a direction of descent with momentum given by

$$\boldsymbol{q}_k = \boldsymbol{g}_k - \frac{\boldsymbol{g}_k^T \boldsymbol{g}_k}{\boldsymbol{g}_{k-1}^T \boldsymbol{g}_{k-1}} \boldsymbol{q}_{k-1}. \tag{6}$$

More details of gradient-based optimization can be found in optimization textbooks.

As mentioned before, the gradient-based methods update the solutions following the direction associated with the gradient $\boldsymbol{g}_k$. When $\|\boldsymbol{g}_k\|_2 < \varepsilon_g$, where $\|\cdot\|_2$ is an L^2 norm, ε_g is a small positive number that indicates the threshold of local optima, the solution is considered as that it converges to the minimum in the search region around the start point's neighborhood, and hence the gradient-based methods are the local search techniques, where the global convergence cannot be guaranteed. If the global minima are strongly desired, some global optimization techniques should be considered.

2.3. **Global optimization methods.** For global convergence, most of the global optimization methods attempt to cover the regions of optima's neighborhood as many as possible. Multi-start, perturbation and multi-individual schemes are commonly used in global optimization.

SA method, and the methods in [24, 25] use multi-start scheme, where the search procedures are repeated from different start points. It is expected that at least one of the start points is within the neighborhood of global minima, clearly the more start points, the better global convergence performance is, but the heavier computational load is required.

On the other hand, to make the solution escape from the local minima, the perturbation scheme adds some perturbation into the candidate solution update in (3) [18]. The additive perturbation level and frequency should be the trade-off between the ability to escape from the local minima and the convergence speed.

Unlike multi-start scheme, the multi-individual schemes such as GA, PSO, ACO update a set of solution candidates parallelly. Different from the gradient-based methods that update the solutions following gradient direction, the global optimization methods introduce probability operation into the search procedure to make the search region not be restricted to local minima; however, it often deteriorates the computational efficiency. Therefore, it is desired to combine the global optimization techniques with gradient for better global convergence as well as lower computational complexity [1, 21]. Inspired by this idea, a multi-point search scheme is investigated where the point search for global optima is combined with region research to improve the probability of global convergence.

3. **Multi-Point Search Based Optimization Algorithm.** In the new multi-point search based optimization, a set of candidate individuals are updated parallelly. The individuals are expected to cover the convergence regions of the optima as many as possible, each individual's local searching uses the gradient to reduce the computational complexity, and information sharing among the individuals may avoid the redundant computation.

3.1. **Initial candidate solutions.** Assume that M individuals are used in the multi-point search based optimization and let the iteration number be $k = 0$. The individuals' initial values $\boldsymbol{x}_0^{(m)}$, $m = 1, 2, \ldots, M$ are chosen randomly. Find the m^*th individual whose value of loss function is the smallest one among $f\left(\boldsymbol{x}_0^{(m)}\right)$ for $m = 1, 2, \ldots, M$, then exchange the 1st with m^*th individuals, and hence $\boldsymbol{x}_0^{(1)}$ has the smallest value of loss function among the M candidates. Let the 1st individual be the primary candidate whose solution is corresponding to point search for global optimum, while the others be the auxiliary ones corresponding to region search to avoid local minima.

Add the candidate solutions $\boldsymbol{x}_0^{(1)}, \ldots, \boldsymbol{x}_0^{(M)}$ into the candidate history record $\boldsymbol{X}$.

3.2. **Gradient for local searching.** In the kth iteration, the candidate solutions are updated from $\boldsymbol{x}_k^{(m)}$ to $\boldsymbol{x}_{k+1}^{(m)}$. It is expected to utilize a local search algorithm with high convergence rate for the primary individual to find the optima quickly, while the algorithm with considerable low computational complexity for auxiliary individuals. For example, the candidate solution of primary individual may be updated by Newton's method if the loss function has the 2nd order derivative in the convex region, or Gauss-Newton method for a quadratic loss function. As for the auxiliary candidates, quasi-Newton method, conjugate method or steepest decent method can be used.

3.3. **Information sharing.** Let the individual number m^* be given by

$$m^* = \arg\min_m f\left(\boldsymbol{x}_{k+1}^{(m)}\right). \tag{7}$$

If $m^* \neq 1$, exchange the 1st individual and the m^*th. Then, among the exchanged candidate solutions, $\boldsymbol{x}_{k+1}^{(1)}$ is the powerful candidate for the next iteration.

Furthermore, if the values $\boldsymbol{x}_{k+1}^{(2)}, \ldots, \boldsymbol{x}_{k+1}^{(M)}$ of auxiliary candidates are very close, i.e., if the solution of the mth auxiliary candidate fulfills

$$\left\|\boldsymbol{x}_{k+1}^{(m)} - \boldsymbol{X}\right\|_2 < \varepsilon_x, \tag{8}$$

where ε_x is a very small positive number, $\boldsymbol{x}_{k+1}^{(m)}$ is regarded as falling into the same local minimum recorded in $\boldsymbol{X}$, and should be regenerated in the perturbation step to escape from the local minima. Otherwise, add the candidate solutions into record $\boldsymbol{X}$.

3.4. **Perturbation of auxiliary candidates.** Perturbations are added to the auxiliary candidates under the following conditions.

1) When the mth auxiliary individual's gradient $\left\|\boldsymbol{g}_k^{(m)}\right\|_2 < \varepsilon_g$, the solution can be regarded as that $\boldsymbol{x}_{k+1}^{(m)}$ converges to the local minimum, and then $\boldsymbol{x}_{k+1}^{(m)}$ is replaced by $\boldsymbol{x}_{k+1}^{(m)} + \boldsymbol{p}_{k+1}^{(m)}$, where $\boldsymbol{p}_{k+1}^{(m)}$ is a random perturbation vector to escape from the local minimum.

2) The gradient direction has the wrong direction in the neighborhood of candidate solution, i.e., if no step size μ_k satisfying

$$f\left(\boldsymbol{x}_k^{(m)} - \mu_k \boldsymbol{q}_k^{(m)}\right) < f\left(\boldsymbol{x}_k^{(m)}\right) \quad \text{for } \varepsilon_\mu < \mu_k \leq 1 \tag{9}$$

exists, the loss function of the mth auxiliary cannot be decreased further, and then the perturbation term $\boldsymbol{p}_{k+1}^{(m)}$ is added to $\boldsymbol{x}_{k+1}^{(m)}$ to start the local searching from a new point.

3) Redundant candidates detected in information sharing are added the perturbation terms to avoid wastefully repeating search of the same local minima.

3.5. **Procedure of optimization.** The procedure of the multi-point based optimization algorithm is summarized in Algorithm 1.

It is seen that the larger M is, the higher probability of global convergence and computational complexity will be; therefore, the choice of individual number M should be a trade-off between the convergence probability and computation load in the large scale problems. In the next section, Algorithm 1's application to numerical optimization in system identification problem is illustrated.

4. **Application in System Identification.** Assume that the physical process can be described by a linear discrete-time model [2]

$$A\left(z^{-1}\right)y(t) = \frac{B\left(z^{-1}\right)}{F\left(z^{-1}\right)}u(t) + \frac{C\left(z^{-1}\right)}{D\left(z^{-1}\right)}w(t), \tag{10}$$

where $u(t)$ and $y(t)$ are the sampled input and output, $w(t)$ is a white stochastic noise, respectively, and t is a normalized instant. Here $A\left(z^{-1}\right) \sim F\left(z^{-1}\right)$ are the polynomials of

Algorithm 1 Algorithm of multi-point based optimization

Set small positive numbers ε_g, ε_x, ε_μ and the maximal iteration number $K_{\max}$.

1: Determine the initial values $\boldsymbol{x}_0^{(1)}, \ldots, \boldsymbol{x}_0^{(m)}$.
2: Find the individual number m^* such that $m^* = \arg\min_m f\left(\boldsymbol{x}_0^{(m)}\right)$. If $m^* > 1$, exchange the values of $\boldsymbol{x}_0^{(1)}$ with $\boldsymbol{x}_0^{(m^*)}$.
3: Add the candidate solutions $\boldsymbol{x}_0^{(m)}$, $m = 1, \ldots, M$ into the history record $\boldsymbol{X}$, and let the iteration number be $k = 0$ to start the iteration.
4: **while** $k < K_{\max}$ **do**
5: Calculate the gradients $\boldsymbol{g}_k^{(m)}$ of M individuals, then calculate the updating vector $\boldsymbol{q}_k^{(m)}$ using the local optimization methods with respect to the primary and auxiliary individuals, determine the step size $\mu_k^{(m)}$.
6: Indicate the mth regeneration index as 1 for the auxiliary individuals if their gradients $\left\|\boldsymbol{g}_k^{(m)}\right\|_2 < \varepsilon_g$, or the step size $\mu_k^{(m)} < \varepsilon_\mu$, otherwise indicate it as 0.
7: Update the candidate solutions $\boldsymbol{x}_{k+1}^{(m)} = \boldsymbol{x}_k^{(m)} - \mu_k^{(m)} \boldsymbol{q}_k^{(m)}$ of primary and auxiliary individuals, respectively.
8: Regenerate the solution whose index is 1 by adding the perturbation term $\boldsymbol{p}_k^{(m)}$ to $\boldsymbol{x}_{k+1}^{(m)}$. Let it be a new start point.
9: Check whether the candidate solutions $\boldsymbol{x}_{k+1}^{(m)}$ of auxiliary individuals are close to any solution in the record $\boldsymbol{X}$. If so, regenerate the solution as a new start point.
10: Find the individual number m^* such that $m^* = \arg\min_m f\left(\boldsymbol{x}_{k+1}^{(m)}\right)$. If $m^* > 1$, exchange the values of $\boldsymbol{x}_{k+1}^{(1)}$ with $\boldsymbol{x}_{k+1}^{(m^*)}$, as well as the gradients $\boldsymbol{g}_k(m)$, rotating and scaling matrixes $\boldsymbol{R}_k$, and $\boldsymbol{q}_k^{(m)}$. Add the candidate solutions into record $\boldsymbol{X}$, then let $k = k + 1$ for the next iteration.
11: **end while**

a backward operator z^{-1}, whereas except $B\left(z^{-1}\right)$, the polynomials are monic. In system identification problems, the modeling issue is to construct a prediction model

$$\hat{y}(t) = \left(1 - \frac{\hat{D}\left(z^{-1}\right)}{\hat{C}\left(z^{-1}\right)} \hat{A}\left(z^{-1}\right)\right) y(t) + \frac{\hat{D}\left(z^{-1}\right)}{\hat{C}\left(z^{-1}\right)} \frac{\hat{B}\left(z^{-1}\right)}{\hat{F}\left(z^{-1}\right)} u(t), \tag{11}$$

by finding the optima $\boldsymbol{x}$ through nonlinear optimization problem of a quadratic loss function defined in (1), where $\boldsymbol{x}$ is the parameter vector of $\hat{A}\left(z^{-1}\right) \sim \hat{F}\left(z^{-1}\right)$ in (11), and the symbol $\hat{}$ indicates the estimates.

The conventional identification algorithms utilize Gauss-Newton method or Levenberg-Marquardt method, where the Hessian matrix is calculated by (4). However, if the information contained in $\{u(t), y(t)\}$ is not sufficient for numerical optimization, and the process is strongly disturbed by the noise term $\frac{C(z^{-1})}{D(z^{-1})} w(t)$, then the surface of loss function is not smooth and there are several local minima, as a result the conventional algorithms are hard to guarantee the global optima, so the proposed algorithm is expected to improve the convergence performance. The effectiveness of the proposed algorithm will be discussed through numerical simulations in the next section.

5. Numerical Examples.

2 numerical examples are considered to demonstrate the convergence performance of the new multi-point based optimization algorithm.

5.1. Case of a benchmark function.

The optimization problem of a benchmark Goldstein-Price function is considered first. It has 2 variables x_1, x_2 as follows:

$$f(x_1, x_2) = \left(1 + (x_1 + x_2 + 1)^2 \left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right) \left(30 + (2x_1 - 3x_2)^2 \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)\right). \quad (12)$$

The schematic contour is shown in right of Figure 1. It is seen that besides a global minimum at $(0, -1)$, there are several local minima such as $(-0.6, -0.4)$, $(0.8, 0.2)$, $(0.6, 0.4)$, and $(1.2, -0.2)$. Hence, the gradient-based methods cannot guarantee the convergence of global optima. The results by Newton's method are shown in left of Figure 2. The initial values of start points are chosen randomly in the range of $-1.75 \leq x_1, x_2 \leq 1.75$, and the simulation is independently executed for 100 runs, whereas only 10 runs converge at the global minimum $(0, -1)$. It implies that the probability of global convergence obtained by Newton's method is only about 0.1 in this problem.

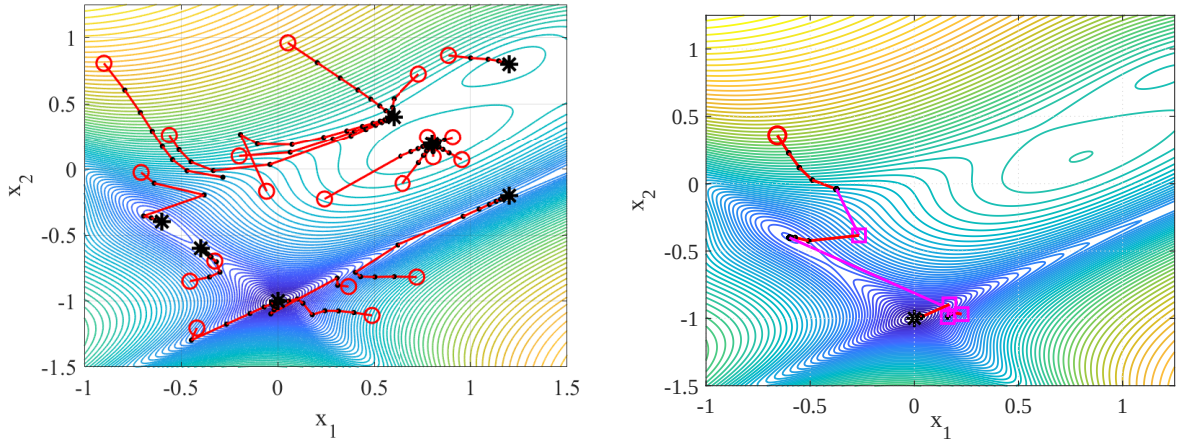

FIGURE 2. Optimization result of function $f(x_1, x_2)$. Left: Local minima obtained by Newton's method with random start points in 20 runs; Right: result in 1 run of multi-point based algorithm. $\circ$: start point; $\square$: perturbation point; $*$: local minima.

4-point based optimization algorithm is used for this problem. Newton's method is chosen in the local searching of primary individual, while quasi-Newton method is chosen for the auxiliary ones. The trajectory of primary's solution in 1 simulation run is shown in right of Figure 2. It is seen that the perturbation term is added to the solution 4 times to make the candidate solution of primary individual escape from the local minima, and then the global optimum $(0, -1)$ can be obtained with high probability.

The average iteration number k to obtain the global optimum $(0, -1)$ in 100 simulation runs with respect to individual number M from 2 to 100 is summarized in Table 1. It is seen that the least iteration number of global convergence decreases with increasing the individual number M, while the value of $k \log M$ is almost the same in this example, which implies that an appropriate choice of M depends on the requirement of convergence and tolerance of computational load.

TABLE 1. Global convergence performance of multi-point based optimization algorithm

M	2	3	4	5	8	12	20	30	60	100
k	79.5	52.7	40.0	36.8	27.3	22.9	19.5	16.9	13.7	12.4
$k \log M$	55.1	55.2	55.4	59.2	56.8	57.0	58.4	57.4	56.1	57.0

5.2. **Case of system identification for dynamic model in closed-loop.** Consider a 3rd order linear process operated by a 2nd order digital controller in closed-loop, where the control interval is 0.18 second, while the input $u(t)$ and output $y(t)$ are sampled at interval of 0.06 second. Assume that the true $A\big(z^{-1}\big)$, $B\big(z^{-1}\big)$ and $F\big(z^{-1}\big)$ are given by

$$A\big(z^{-1}\big) = 1,\ B\big(z^{-1}\big) = 0.1483z^{-1} - 0.1477z^{-2} + 0.0638z^{-3},$$
$$F\big(z^{-1}\big) = 1 - 2.2123z^{-1} + 2.0478z^{-2} - 0.7095z^{-3},$$

and $\frac{C(z^{-1})}{D(z^{-1})}w(t)$ is a 2nd order auto-regression moving average (ARMA) stochastic process, $w(t)$ is a white stochastic noise with finite variance. Due to the feedback controller, the data of $\{u(t), y(t)\}$ are not sufficiently strong for the optimization problem, and both $u(t)$ and $y(t)$ are correlated with the noise term $w(t)$, and then the numerical optimization is an ill-conditioned problem, where the parameter estimation of $\hat{F}\big(z^{-1}\big)$ is very sensitive to the noise term and initial values at start point.

Let the coefficients of $\hat{F}\big(z^{-1}\big)$ with respect to power of backward operator z^{-1} be denoted as x_1, x_2, x_3, the coefficients of $\hat{B}\big(z^{-1}\big)$ as x_4, x_5, x_6, respectively, and $\boldsymbol{x}$ be the parameter vector of $x_1 \sim x_6$. It is seen that all the parameters in $\boldsymbol{x}$, especially the model poles associated parameters $x_1 \sim x_3$ of $\hat{F}\big(z^{-1}\big)$, influence the performance of prediction model in (11), so the square error $\|\boldsymbol{x}_{\text{True}} - \boldsymbol{x}\|_2^2$ rather than the error evaluation of a single specified parameter is used to evaluate the accuracy of $\boldsymbol{x}$ in this example, where $\boldsymbol{x}_{\text{True}}$ is the vector of true parameters.

8-point based optimization algorithm is used in the simulation, where the Levenberg-Marquardt method with regularization terms in (4) is used for local searching. The estimated parameters by the proposed algorithms and the conventional Levenberg-Marquardt method using single individual are summarized in Table 2, where the primary candidate $\boldsymbol{x} = \boldsymbol{x}_k^{(1)}$ in the proposed algorithm has much smaller error than that of the conventional Levenberg-Marquardt method using single individual. Since $\boldsymbol{x}_k^{(1)}$ is close to $\boldsymbol{x}_{\text{True}}$, the prediction model in (11) can be used to predict the true system output. Though x_4 and x_6 in the conventional method are also close to the true ones, the large error of x_1, x_2 and x_3 makes the poles of $\hat{F}\big(z^{-1}\big)$ be far away from that of $F\big(z^{-1}\big)$; as a result, large error arises in the prediction $\hat{y}(k)$ using the estimates obtained by the conventional single individual method.

TABLE 2. Parameters obtained by 8-point based optimization algorithm

Parameter	x_1	x_2	x_3	x_4	x_5	x_6	$\|\boldsymbol{x}_{\text{True}} - \boldsymbol{x}\|_2^2$
$\boldsymbol{x}_{\text{True}}$	−2.2123	2.0478	−0.7095	0.1483	−0.1477	0.0638	——
8-point algorithm	−2.1683	1.9849	−0.6476	0.1322	−0.1552	0.0400	0.0106
Conventinal method	−2.4713	2.4423	−0.9517	0.1358	−0.1925	0.0626	0.2389

6. **Conclusions.** A multi-point search based algorithm is proposed for nonlinear optimization problems in this paper. It uses parallel search technique to update the candidate solutions of a primary and several auxiliary individuals from different start points. The gradients with respect to the parameter vector are utilized to improve the efficiency of point search especially for the primary individual, and region search is performed by auxiliary individuals. Moreover, information sharing helps to detect the local minima and to reduce the redundant computation, and perturbation added to the candidate solutions of auxiliary individuals makes they escape from the local minima. Consequently, the proposed algorithm has better global convergence performance with considerably

low computational complexity. Its effectiveness has been demonstrated in optimization of benchmark function, and a closed-loop identification problem. Some meaningful issues, for example, whether the candidate solution converges to global optima and applications to machine learning algorithms will be considered in the future works.

REFERENCES

[1] O. Nelles, *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*, Springer, 2001.

[2] L. Ljung, *System Identification – Theory for the Users*, Prentice Hall, Englewood Cliffs, NJ, 1999.

[3] T. Hachino, On-line Gaussian process model identification of nonlinear systems using particle swarm optimization, *International Journal of Innovative Computing, Information and Control*, vol.17, no.6, pp.2191-2200, 2021.

[4] F. L. Perez, C. A. Pitz and R. Seara, A two-gain NLMS algorithm for sparse system identification, *Signal Processing*, vol.200, 2022.

[5] S. Haykin, *Adaptive Filter Theory*, 5th Edition, Pearson, 2013.

[6] S. Abrar, A. Zerguine and K. Abed-Meraimm, Adaptive algorithms for blind channel equalization in impulsive noise, *Signal Processing*, vol.201, https://doi.org/10.1016/j.sigpro.2022.108626, 2022.

[7] L. C. Resende, R. M. S. Pimenta, N. N. Siqueira, F. Igreja, D. B. Haddad and M. R. Petraglia, Analysis of the least mean squares algorithm with reusing coefficient vector, *Signal Processing*, vol.202, https://doi.org/10.1016/j.sigpro.2022.108742, 2023.

[8] E. Nobuyama and N. Sebe, *Optimization Theory for Systems and Control*, Corona Publishing, 2015.

[9] O. S. Patil, R. Sun, S. Bhasin and W. E. Dixon, Adaptive control of time-varying parameter systems with asymptotic tracking, *IEEE Trans. Automatic Control*, vol.67, no.9, pp.4809-4815, 2022.

[10] M. Rotulo, C. D. Persis and P. Tesi, Online learning of data-driven controllers for unknown switched linear systems, *Automatica*, vol.145, https://doi.org/10.1016/j.automatica.2022.110519, 2022.

[11] H. Nakagawa, *Machine Learning*, School of Engineering, The University of Tokyo, Maruzen Publishing, 2015.

[12] Y. Bai, W. Chen, J. Chen and W. Guo, Deep learning methods for solving linear inverse problems: Research directions and paradigms, *Signal Processing*, vol.177, 2020.

[13] J. Nelder and R. Mead, A simplex method for function minimization, *Computer Journal*, vol.7, pp.308-313, 1965.

[14] I. Moser, Hooke-Jeeves revisited, *2009 IEEE Congress on Evolutionary Computation*, pp.2670-2676, 2009.

[15] L. E. Scales, *Introduction to Nonlinear Optimization*, Computer and Science Series, Macmillan, London, 1985.

[16] S. Z. Khong, Y. Tan, C. Manzie and D. Nešić, Extremum seeking of dynamical systems via gradient descent and stochasitc approximation methods, *Automatica*, vol.56, pp.44-52, 2015.

[17] E. Aiyoshi, T. Okamoto and K. Yasuda, *Fundamentals of Optimization Approaches*, Morikita Publishing, 2014.

[18] T. Okamoto and H. Hirata, Global optimization using a multipoint type quasi-chaotic optimization method, *Applied Soft Computing*, vol.13, no.2, pp.1247-1264, 2013.

[19] J. Kennedy and R. Eberhart, Particle swarm optimization, *Proc. of IEEE International Conference on Neural Networks*, vol.4, pp.1942-1948, 1995.

[20] Y. Shi and R. Eberhart, A modified particle swarm optimizer, *Proc. of IEEE International Conference on Evolutionary Computation*, pp.69-73, 1998.

[21] M. Noel, A new gradient based particle swarm optimization algorithm for accurate computation of global minimum, *Applied Soft Computing*, vol.12, no.1, pp.353-359, 2012.

[22] M. R. Bonyadi and Z. Michalewicz, Particle swarm optimization for single objective continuous space problems: A review, *Evolutionary Computation*, vol.25, no.1, pp.1-54, 2017.

[23] M. Dorigo and L. M. Gambardella, Ant algorithms for discrete optimization, *Artificial Life*, vol.5, no.2, pp.137-172, 1999.

[24] H. Kanemitsu, H. Konno and N. Takahashi, Global optimization algorithms based multistart local search, *Journal of Hokkaido University of Education (Natural Sciences)*, vol.58, no.2, pp.1-10, 2008.

[25] L. Sun, C. Zhou and A. Sano, New prediction error method based on cyclostationarity for system identification, *ICIC Express Letters*, vol.11, no.1, pp.189-196, 2017.