

Hybrid Optimization Using DIRECT, GA, and SQP for Global Exploration

Satoru HIWA, Tomoyuki HIROYASU, Mitsunori MIKI

Abstract—As there are many good optimization algorithms each with its own characteristics, it is very difficult to choose the best method for optimization problems. Thus, it is important to select and apply the appropriate algorithms according to the complexities of the problem. However, it is difficult to solve very complicated problems with only a single algorithm, and a hybrid optimization approach, which combines multiple optimization algorithms, is necessary. To develop an efficient hybrid optimization algorithm, it is necessary to determine how the optimization process is performed. This paper focuses on the balance between local and broad searches. Multiple optimization methods are controlled to derive both the optimum point and the information of the landscape. To achieve the proposed optimization strategy, three distinguished optimization algorithms are introduced: DIRECT (DIviding RECTangles), GAs (Genetic Algorithms), and SQP (Sequential Quadratic Programming). To integrate these three algorithms, each algorithm, especially DIRECT, was modified and developed. This paper describes a new hybrid method using these three algorithms. The performance of the proposed hybrid algorithm was examined through numerical experiments. From these experiments, not only the optimum point but also the information of the landscape was determined. The information of the landscape verified the reliability of optimization results.

I. INTRODUCTION

Various efficient optimization algorithms have been developed and used in many applications with good results. Sequential Quadratic Programming (SQP) [1], Genetic Algorithms (GAs) [2], and Simulated Annealing (SA) [3] are well-known optimization algorithms. SQP is one of the best algorithms to solve constrained nonlinear optimization problems. Heuristics, such as GAs and SA, are also efficient in solving complex multimodal optimization problems. Although these algorithms have been applied to several types of real-world problems, their performance is significantly influenced by the complexity of the given problems. For example, SQP show efficient performance for unimodal functions, while GAs suffer from poor convergence. On the other hand, GAs are effective even when the target problem has several local optima.

To solve real-world optimization problems, it is important to select the appropriate optimization algorithms according to the complexity of the problem. However, it may be difficult to solve such problems with only a single algorithm when the complexities of the problems are high. As one method

to solve these difficulties, a hybrid optimization approach is often used [4] [5]. Hybrid optimization algorithms consist of multiple optimization algorithms with different features and performances, and should provide high performance that cannot be achieved with only a single algorithm.

In this paper, the necessity of the hybrid optimization approach is first discussed, and then a description of how to design hybrid optimization algorithms is given.

II. HYBRID OPTIMIZATION APPROACH

A. Strategy for Controlling the Multiple Optimization Algorithms

The most important things in developing an efficient hybrid optimization algorithm are how the optimization process is performed and what types of solution are required after optimization. For example, the best solution may be required with a reasonable convergence speed, or the promising area or candidate solutions may be obtained after the optimization process even if the execution time is not realistic. The desired solutions may vary depending on the user, and it is difficult to fulfill all demands.

Therefore, it is necessary to define the search strategy or the required solutions before designing a hybrid optimization algorithm. This is referred to here as the "optimization strategy." The optimization strategy determines the algorithms that should be hybridized, and also provides control policies for the multiple algorithms.

B. Optimization Strategy for Global Exploration

Based on the discussion in the previous section, the effective optimization strategy for designing an efficient hybrid algorithm is discussed in this section. First, we focus on the problem in application of the optimization algorithm to real-world problems.

Generally, when we solve an optimization problem, the landscape of the target problem is unknown. In this case, various optimization algorithms are applied and the landscape of the solution space is investigated through trial and error. Otherwise, application of strong global optimization methods, such as GAs, is expected.

Although GAs are very powerful and are called "global" optimization algorithms, they are not guaranteed to search the variable space uniformly and equally because their search is probabilistic. That is, as GAs do not necessarily cover the entire search space, unexplored areas may remain in the search results. Thus, there is significant doubt whether the solutions obtained by GAs represent the global optima. In

S. Hiwa is with the Graduate School of Engineering, Doshisha University 610-0321 Kyoto, Japan (email: hiwa1983@gmail.com).

T. Hiroyasu is with the Department of Engineering, Doshisha University 610-0321 Kyoto, Japan (email: tomo@is.doshisha.ac.jp).

M. Miki is with the Department of Engineering, Doshisha University 610-0321 Kyoto, Japan (email: mmiki@mail.doshisha.ac.jp).

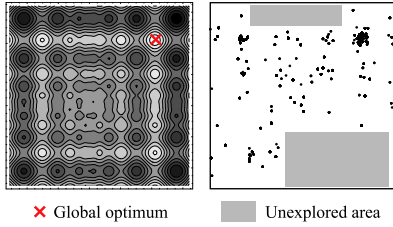


Fig. 1. Unexplored areas in the GA search.

Figure 1, the figure on the right shows the GA search in two-dimensional space, while that on the left shows the contour plot of the target problem. From the figure, although GA obtains the global optima, unexplored areas remain after GA search.

To determine whether the optimum solution exists in the unexplored area, there is no choice but to search within this area. Otherwise, we have to believe the results provided by the GA.

To solve these problems, we propose an optimization strategy for global exploration, to search the variable space uniformly and equally. That is, areas that cannot be searched by probabilistic algorithms, such as GAs, are covered by other algorithms with features different from GAs. In this approach, it is expected that we can obtain the landscape information, and evaluate the reliabilities of the obtained solutions after optimization.

C. Optimization Algorithms Used for the Proposed Strategy

Based on the discussion in the previous section, it is difficult to perform a search based on the proposed strategy using only GAs. Therefore, an optimization algorithm that can search the variable space more uniformly and equally is needed. However, such a broad search may lead to the requirement of a large number of function evaluations. Therefore, an algorithm that can efficiently explore the entire search space is required. In this research, the DIRECT optimization algorithm [6] is used for more global exploration. DIRECT performs a global search of the variable space while identifying promising areas.

Although DIRECT can search the variable space uniformly and equally, its features lead to convergence degradation. On the other hand, GAs are faster than DIRECT with regard to convergence speed. Therefore, if these algorithms are hybridized while preserving the characteristic features of each algorithm, an efficient hybrid optimization algorithm will be developed.

Moreover, as GAs have trouble finding an exact solution after reaching a global region because its search are probabilistic, Sequential Quadratic Programming (SQP) is also hybridized with these two algorithms. SQP is among the best algorithms to solve nonlinear constrained convex optimization problems, using gradient information. Details of these three algorithms (referred to here as "sub-algorithms") are described in the next section.

III. OPTIMIZATION ALGORITHMS

A. DIRECT Algorithm

DIRECT (DIViding RECTangles) is a global optimization algorithm to solve optimization problems with bound constraints. The DIRECT algorithm is a modification of Lipschitzian optimization. Classical Lipschitzian Optimization requires an appropriate setting for the Lipschitz constant. DIRECT does not require estimation of the Lipschitz constant, and searches for optimum solutions using all possible constants [7] [8].

DIRECT divides the hyper-rectangles (referred to here as "boxes"), and samples their center points. The DIRECT algorithm is given as follows:

DIRECT algorithm

- 1) Normalize the search space to the unit hyper-cube with center point c_1 , and evaluate $f(c_1)$.
- 2) Until the terminal criterion is satisfied,
 - a) Identify the set S of potentially optimal boxes.
 - b) For each box $j \in S$
 - i) Sample new points, and evaluate the function value at the new points.
 - ii) Divide the box j .

Each operation is described in the next sections.

1) *Initialization and Division of the Hyper-cube*: DIRECT begins the search by transforming the domain of the target problem into the unit hyper-cube:

$$\bar{\Omega} = \{x \in R^n : 0 \leq x_i \leq 1\} \quad (1)$$

Then, the center point of the hyper-cube c_1 is sampled. Next, DIRECT divides this space by evaluating the function values at the points $c_1 \pm \delta \bar{e}_i$ ($i = 1, \dots, n$), where δ is one-third the side-length of the hyper-cube, and \bar{e}_i is the i th Euclidean base-vector. That is, a hyper-cube is divided into three hyper-rectangles in each dimension.

The sequence of the dimensions to be divided is determined by w_i , which is shown in Equation (2), and the first division is performed in the dimension with the smallest w_i .

$$w_i = \min\{f(c_1 + \delta e_i), f(c_1 - \delta e_i)\} \quad (i = 1, \dots, n) \quad (2)$$

This operation is repeated for all dimensions on the box with the point c_1 , choosing the next dimension with the next smallest w_i . Figure 2 illustrates the search space after the initial divisions. The numbers in the figure on the left of Figure 2 show the function values at each point. In this case, $w_1 = 60.0$ and $w_2 = 150.1$; therefore, the first division is performed along the direction of x_1 .

2) *Division of the Hyper-rectangles*: DIRECT divides the hyper-rectangles by performing division only in the dimensions with the longest side length of the hyper-rectangles. The sequence of the dimensions to be divided is determined by w_j :

$$w_j = \min\{f(c_i + \delta_i e_j), f(c_i - \delta_i e_j)\}, \quad j \in I \quad (3)$$

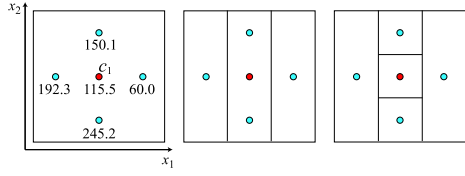


Fig. 2. Design variable space after the first division.

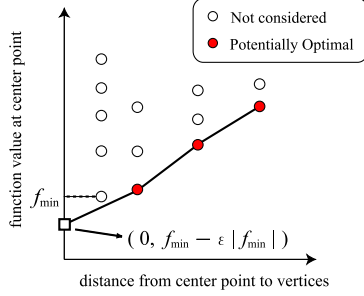


Fig. 3. Selection of hyper-rectangles to be divided.

where I is set of the dimensions with the longest side length, and δ_i is one-third the length of the longest side of the hyper-rectangle i . DIRECT performs division for all dimensions in I .

3) *Potentially Optimal Boxes*: DIRECT divides all of the boxes that satisfy the definition of potentially optimal:

Definition (potentially optimal boxes)

Let $\epsilon > 0$ be a positive constant and let f_{min} be the current best function value. A box j is potentially optimal if there exists some $\hat{K} > 0$ such that:

$$\begin{aligned} f(c_j) - \hat{K}d_j &\leq f(c_i) - \hat{K}d_i, \forall i, \text{ and} \\ f(c_j) - \hat{K}d_j &\leq f_{min} - \epsilon|f_{min}| \end{aligned} \quad (4)$$

In Equation (4), c_j is the center point of the box j , and d_j defines a measure for this hyper-rectangle. Jones et al. [6] used the distance from the center point c_j to its vertices. Jones also recommended the value $\epsilon = 1.0 \times 10^{-4}$. This definition is illustrated in Figure 3.

In Figure 3, the horizontal axis represents the d in Equation (4), and the vertical axis shows $f(c)$. From Figure 3, it can be seen that the potentially optimal boxes lie on the bottom, right hand part of the convex hull of the all boxes in the graph.

Moreover, the hyper-rectangles with f_{min} are not always potentially optimal. That is, ϵ controls the local and global search. The DIRECT search is performed by repeating the above operations. Several iterations of the DIRECT search are shown in Figure 4.

4) *Features of the DIRECT search*: To determine the characteristics of the DIRECT search, DIRECT was applied to the following three benchmark functions—Rosenbrock, Rastrigin and Schwefel function:

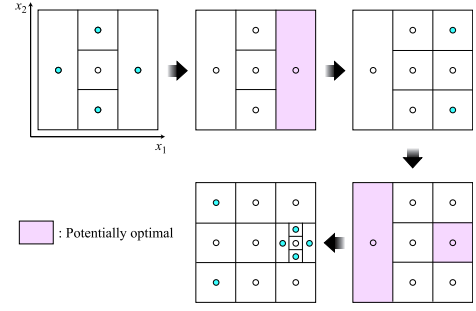


Fig. 4. Several iterations of the DIRECT search.

- Rosenbrock function

$$\begin{aligned} f_1(\vec{x}) &= \sum_{i=2}^n \{100(x_1 - x_i^2)^2 + (1 - x_i)^2\} \\ &(-2.048 \leq x_i \leq 2.048) \\ \min(f(x)) &= F(1, 1, \dots, 1) = 0 \end{aligned} \quad (5)$$

The Rosenbrock function is unimodal and has correlation among its design variables.

- Rastrigin function

$$\begin{aligned} f_2(\vec{x}) &= 10n + \sum_{i=1}^n \{x_i^2 - 10 \cos(2\pi x_i)\} \\ &(-4.12 \leq x_i \leq 6.12) \\ \min(f(x)) &= F(0, 0, \dots, 0) = 0 \end{aligned} \quad (6)$$

The Rastrigin function has lattice-shaped local optima around the global optimum, and there is no correlation among design variables.

- Schwefel function

$$\begin{aligned} f_3(\vec{x}) &= 418.98 \cdot n - \sum_{i=1}^n x_i \sin(\sqrt{|x_i|}) \\ &(-512 \leq x_i \leq 640) \\ \min(f(x)) &= F(420.97, \dots, 420.97) = 0 \end{aligned} \quad (7)$$

The Schwefel function consists of a number of peaks and valleys. It has some local optima far from the global optimum where many search algorithms are trapped. There is no correlation among its design variables.

These functions are two-dimensional. Figure 5 shows the search history of DIRECT. From the figure, it can be seen that DIRECT explores the search space uniformly and equally, and we can roughly grasp the landscapes of the target problems from the results. Therefore, the search history of DIRECT provides the information of the landscape.

B. Genetic Algorithms

Generally, in GAs, the binary representations are used as representation schemes. However, for function optimization, Real-Coded GAs (RCGAs), which use real number vector representation of chromosomes, work well for global optimization of nonlinear functions. In RCGAs, offspring can be generated by dealing directly with the parent distribution

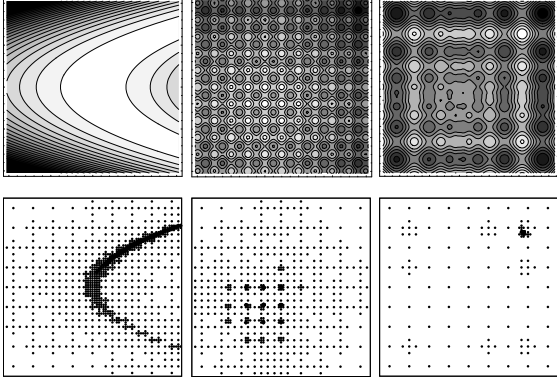


Fig. 5. All points sampled by DIRECT.

in design space. Thus, various crossover operators have been proposed for RCGAs, some of which have been shown to have efficient search ability [9] [10] [11] [12].

Simplex Crossover (SPX) [12] [13] [14] is one of the most efficient crossover operators for RCGAs. In n -dimensional space, SPX generates offspring in a simplex, a polyhedron formed by $n+1$ parents. As SPX is robust for the correlation among design variables or the rotation of the coordinate system, RCGA using the SPX operator was used for the proposed hybrid optimization algorithm. Details of the SPX method are as follows:

SPX algorithm

- 1) Select $n+1$ parents $\vec{p}_0, \dots, \vec{p}_n$ from the population by random sampling.
- 2) Calculate their center of mass \vec{g} as:

$$\vec{g} = \frac{1}{n+1} \sum_{i=0}^n \vec{p}_i \quad (8)$$

- 3) Calculate \vec{x}_k and \vec{c}_k by:

$$\begin{aligned} \vec{x}_k &= \vec{g} + \epsilon(\vec{p}_k - \vec{g}) \\ \vec{c}_k &= r_{k-1}(\vec{x}_{k-1} - \vec{x}_k + \vec{c}_{k-1}) \\ r_k &= (u(0,1))^{\frac{1}{k+1}} \end{aligned} \quad (9)$$

where ϵ is the expansion rate ($\epsilon > 0$), a control parameter of SPX. $u(0,1)$ is uniform random number $\in [0, 1]$.

- 4) Generate offspring \vec{c} by:

$$\vec{c} = \vec{x}_n + \vec{c}_n \quad (10)$$

Figure 6 illustrates the offspring generation by SPX. SPX generates offspring distributed uniformly on the range illustrated in Figure 6, where ϵ is the expansion rate and a positive parameter of SPX. The expansion rate has a marked effect on the search of SPX, and the efficient value of the expansion rate $\epsilon = \sqrt{n+2}$ is recommended [12].

C. SQP

Sequential Quadratic Programming (SQP) is one of the most efficient gradient-based algorithms for constrained non-

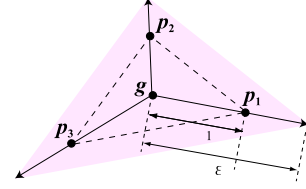


Fig. 6. Offspring generation by SPX.

linear optimization problems. Here, the open source software ADVENTURE_Opt module, developed as part of the ADVENTURE project [15], was used.

IV. VERIFICATION OF THE HYBRID OPTIMIZATION ALGORITHM

In this section, one example of the hybrid optimization approach using sub-algorithms—DIRECT, GA, and SQP—is proposed, and studies to investigate its performance are described.

A. Hybrid Optimization Algorithm

In this section, we propose a hybrid optimization algorithm that achieves the proposed strategy, and examine the effective control of sub-algorithms. First, in the proposed hybrid optimization algorithm, the purposes of the three sub-algorithms are summarized as follows:

DIRECT

- To search the variable space uniformly and equally.
- To identify the promising area and narrow down the search area.

GA

- To intensify the search in the promising area and improve the accuracy of solutions.

SQP

- Fine-tuning to determine the optimal solution.

As DIRECT can globally explore the entire search space, it is used to achieve the strategy—to explore the design space uniformly and equally. Moreover, DIRECT defines the “potentially optimal box” that is considered to be promising. Therefore, we assume that DIRECT can also be used to identify the promising area.

GA is used for more locally intensified searches than DIRECT, and improves the overall search performance. GA begins the search by utilizing the center points of the potentially optimal boxes as their individuals. By this, GA can intensify the search in the promising area found by DIRECT.

Although SQP is not efficient for multimodal functions, rapid convergence to an optimum solution is obtained using the gradient information for unimodal functions. Therefore, SQP is used to fine-tune the solutions obtained by DIRECT and GA. SQP begins from the best point found so far, and improves the best solution.

The procedure of the proposed hybrid optimization algorithm is as follows:

Hybrid Optimization Algorithm

- 1) Perform the search by DIRECT until the terminal criterion is satisfied.
- 2) Identify the potentially optimal boxes when the DIRECT search was terminated.
- 3) Execute GA until the terminal criterion is satisfied. In this, the center points of the potentially optimal boxes are utilized as individuals in GA.
- 4) Execute SQP from the elite individual in GA.

In the proposed algorithm, the number of individuals equals the number of potentially optimal boxes in DIRECT, because GA utilizes the center point of the potentially optimal boxes. However, the number of potentially optimal boxes increases with an increase in the dimensions and iterations. Therefore, DIRECT must divide a large number of boxes, and so its performance becomes poor for high-dimensional problems [16]. In addition, as GA utilizes the center points, an increase in the number of the potentially optimal boxes leads to an increase in the number of the individuals of GA. We assume that the number of individuals is determined and fixed according to the complexity of the problem. Thus, in the proposed hybrid algorithm, the number of potentially optimal boxes should be adjusted according to the number of individuals.

Therefore, in Step 3), if the number of potentially optimal boxes is not sufficient for the GA search, randomly generated individuals are added. Otherwise, if the number of potentially optimal boxes is greater than the number of individuals, a certain number of potentially optimal boxes should be selected for the individuals of GA. Thus, to select the appropriate number of boxes, box selection rules are needed.

One of the shortcomings of DIRECT is the lack of an obvious terminal criterion [8] [16] [17]. Although Jones' original DIRECT uses the iteration limit as the termination rule, it is unsuitable for the proposed hybrid optimization algorithm, in which DIRECT should be stopped after completing the global exploration of the search space. If the iteration limit was set, we would have to appropriately adjust the limit to perform the global exploration. Therefore, it is necessary to define the effective terminal conditions for the proposed hybrid algorithm.

Thus, some modifications with respect to the box selection rules and terminal rules were made to DIRECT in this study. Details of the modifications and terminal criterion are described in the next section.

It is also necessary to define the terminal criterion of GA. Generally, in RCGA, the search is terminated by the number of function evaluations or the function value threshold prescribed. Thus, a new terminal condition is required as in the case of DIRECT. The new terminal criterion of GA is described in Section IV-C.

B. Modifications to the DIRECT Algorithm

1) *Box Selection from the Potentially Optimal Set:* Here, the selection mechanism of the boxes chosen for division is proposed. By selecting a certain number of boxes from

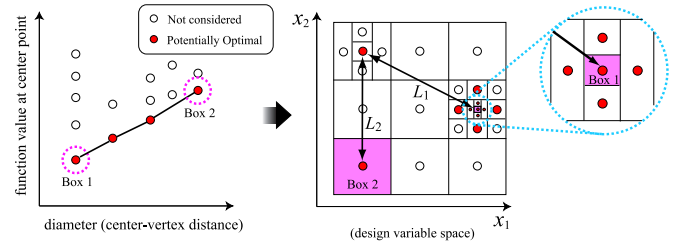


Fig. 7. Box selection algorithm

the set of potentially optimal boxes, the number of boxes partitioned at each iteration can be reduced.

However, inadequate selection of the boxes breaks the novel concepts of DIRECT. For example, if boxes with smaller diameter (center-vertex distance) are chosen for division, the DIRECT search is biased more toward local improvement [18], while the selection of boxes with larger diameter biases the search toward exhaustive search.

Therefore, box selection should be made without breaking the original search characteristics. Typical implementations of DIRECT balance local and global searches by selecting both smaller and largest boxes as potentially optimal. Thus, we propose the following box selection algorithm:

Box selection algorithm

- 1) Identify the set S of potentially optimal boxes.
- 2) If the number of boxes in S is larger than the prescribed parameter $N_{reduced}$,
 - a) Select the best box j_{min} in the boxes with the smallest diameter and the best box j_{max} among the boxes with the largest diameter from S , and add them to the reduced potentially optimal set $S_{reduced}$.
 - b) Remove j_{min} and j_{max} from S .
 - c) for each box $j \in S$,
 - i) Calculate the distance l_1 between j and j_{min} in design variable space.
 - ii) Calculate the distance l_2 between j and j_{max} in design variable space.
 - iii) $L_j = l_1 + l_2$
- 3) Sort S by L_j in descending order, and select $N_{reduced}$ boxes with larger L_j .
- 4) Add $N_{reduced}$ boxes selected in 3) to $S_{reduced}$.

Here, the boxes with the best function value in each diameter are referred to as the "best box." $N_{reduced}$ is the number of the selected boxes. Figure 7 illustrates the selection procedures. In this mechanism, the smallest and largest boxes in the potentially optimal set are always selected. Moreover, the boxes near the smallest and the largest are discarded by calculating the distance L_j in design variable space from them. In this way, the search characteristics of the original DIRECT are preserved while reducing the number of the potentially optimal boxes, without biasing the search toward local or global search.

The number of selected boxes $N_{reduced}$ is the control parameter for reduction level. If the number of potentially optimal boxes is smaller than $N_{reduced}$, the selection algorithm is not applied. As DIRECT is switched to GA, the number of potentially optimal boxes at the end of the DIRECT search corresponds to the number of individuals in GA. Thus, $N_{reduced}$ should be determined based on the number of individuals in GA.

2) *Terminal Criterion*: To perform efficient switching to GA, a new terminal criterion is proposed. In the proposed hybrid optimization algorithm, only a certain depth of the design space exploration is required because DIRECT is not used to obtain the global optima, but is used only for global exploration of the solution space.

Therefore, we utilize a new terminal criterion—the longest side length of the best potentially optimal box. The “best potentially optimal box” is that with the best function value in the potentially optimal set. In this criterion, DIRECT is terminated when the longest side length of the best potentially optimal box is less than the prescribed tolerance value. We can easily set the tolerance and terminate the DIRECT search at the required level of exploration, because the longest side length of the box represents the degree of exploration.

C. New Terminal Criterion for GA

In the proposed hybrid optimization algorithm, as the local search is performed by SQP, it is not necessary for GA to make local improvements. Similar to DIRECT, only a certain depth of design space exploration is required. Thus, we define the “spread of individuals in design variable space” and use this as the terminal criterion.

The spread of individuals in design variable space corresponds to the distance from the individual with the minimum design variable to that with the maximum:

$$d_i = \max(x_i) - \min(x_i) \quad (i = 1, \dots, n) \quad (11)$$

If d_i is smaller than the threshold in all dimensions, GA is terminated because this means that the population of GA converges. We can easily determine the threshold according to the required level of exploration.

V. NUMERICAL EXPERIMENTS

In this section, we describe application of the proposed hybrid optimization algorithm to the benchmark problem and discuss its efficiency.

A. Experimental Setup

The search of the hybrid optimization algorithm was compared with the search using only GA through the Rosenbrock function, the Rastrigin function and the Schwefel function. The dimensions of the problems were set to 10. The parameters for the terminal criterion of each sub-algorithm were set as follows:

DIRECT

- Stop when the longest side length of the best potentially optimal box is less than 10^{-3} .

- Stop when the function value is less than 10^{-3} .

GA

- Stop when the spread of the individuals in design space is less than $(R_{upper} - R_{lower}) \times 10^{-3}$.
- Stop when the function value is less than 10^{-4} .

SQP

- Stop when the value of $p_k (= x^{(k+1)} - x^{(k)})$ has reached the tolerance value 10^{-3} .
- Stop when 1000 iterations have been reached.

To illustrate the effectiveness of the proposed hybrid optimization approach, the proposed algorithm was compared to the search using only GA in 30 runs. The search using only GA was repeated until the function value was less than 10^{-6} . In addition, if GA could not obtain the optimum in 10^6 function evaluations, the search was terminated.

In DIRECT, the number of selected boxes $N_{reduced}$ is set to 100. That is, the number of individuals in GA becomes the same value.

B. Results and Discussion

Table I shows the function value when each sub-algorithm was terminated, and also shows the function value obtained by the search using only GA. The function value of the hybrid optimization algorithm was equal to that of SQP because SQP improves the best solution found so far. Moreover, Table II also shows the number of function evaluations.

As shown in Table I, for the Rosenbrock function and the Schwefel function, the hybrid optimization algorithm obtained the optimum. In addition, Table II shows that the proposed hybrid algorithm can derive the optimum with lower function evaluations than the GA-only search. SQP was successful in improving the solutions obtained in the GA search, and obtained the global optima with less function evaluations.

On the other hand, for the Rastrigin function, the proposed hybrid algorithm could not obtain the optimum. In the proposed algorithm, GA could not intensify the search in the promising area because the potentially optimal boxes that were identified when the DIRECT search was terminated converged to the local optimum. Moreover, for the Rastrigin function, SQP failed to line search, so that it could not improve the best solution. The Rastrigin function has lattice-shaped local optima around the global optimum, and a local optimum exists near the center point of the search space. Therefore, as DIRECT samples the center point first, it explored near the center point and converged to the local optimum. Thus, the performance of the proposed approach was not efficient for the functions with the local optima near the center of the search space.

However, our purpose was not to obtain the optimum, but to cover the search space and to provide the information of the landscape. Therefore, to determine whether the proposed optimization strategy—to search the design space uniformly and equally—was achieved by the hybrid approach, the histories in design variable space of the DIRECT and GA search in the hybrid algorithm and the search by GA only

TABLE I

FUNCTION VALUE OF THE OBTAINED SOLUTION IN EACH ALGORITHM.

#value		DIRECT	GA	SQP (Hybrid)	GA (single)
f_1	AVG	1.20E+00	8.10E-05	1.54E-05	5.45E-05
	St. Dev.	0.00E+00	1.44E-05	1.41E-05	2.08E-04
f_2	AVG	9.99E+00	3.02E+00	3.02E+00	2.02E+00
	St. Dev.	0.00E+00	2.01E+00	2.01E+00	1.01E+00
f_3	AVG	3.52E-02	1.49E-04	5.05E-08	5.58E+02
	St. Dev.	0.00E+00	1.11E-04	3.44E-08	1.82E+02

AVG: Average of 30 runs

 f_1 : Rosenbrock, f_2 : Rastrigin, f_3 : Schwefel

TABLE II

NUMBER OF FUNCTION EVALUATIONS OF EACH ALGORITHM.

#eval		DIRECT	GA	SQP	Hybrid	GA
f_1	AVG	22479	85953	53	108485	192323
	St. Dev.	0	10361	8	10362	47584
f_2	AVG	475	163107	45	163627	245167
	St. Dev.	0	25626	17	25630	28403
f_3	AVG	13529	111857	46	125432	279703
	St. Dev.	0	6376	12	6378	22402

AVG: Average of 30 runs

 f_1 : Rosenbrock, f_2 : Rastrigin, f_3 : Schwefel

are shown in Figure 8. In these figures, the histories in ten-dimensional variable space for the Schwefel function are projected into two-dimensional space. Although there are 45 plots, only a typical example (x_2, x_5 plane) is shown in Figure 8. Here, the global optima of the Schwefel function is $(x_1, \dots, x_{10}) = (420.97, \dots, 420.97)$.

Figure 8 shows that the hybrid algorithm performs global exploration, and detects not only the global optimum, but also the local optimum. On the other hand, the GA-only search failed to reach the global optimum and could not detect any local optima. That is, in the hybrid algorithm, DIRECT covered the unexplored area where GA could not explore. From the experimental results, the effectiveness of the proposed hybrid optimization approach was demonstrated.

C. Usage of the Search Results of Hybrid Optimization

To show that the proposed hybrid optimization approach provides the information of the landscape, we discuss the usage of the search results of the proposed algorithm.

If the proposed algorithm provides the information of the landscape, the landscape can be roughly understood from the search results. To verify this, we attempted to fit the search results by DIRECT and GAs in the proposed algorithm on each benchmark function to the following polynomial function:

$$f(x_1, x_2) = \sum_{i=0}^4 \sum_{j=0}^i a_{ij} x_1^{i-j} x_2^j \quad (12)$$

The parameter a_{ij} is determined by least-squares fitting. We assume that the landscape of the problem can be roughly approximated by the fitting function. Fitting is made on two design variables (x_1, x_2).

Figure 9 shows the results of fitting. For the Rosenbrock function, it can be seen that the landscape approximated from

the results of the hybrid optimization represents the long narrow ridge of Rosenbrock. Especially, for the Schwefel function, the multimodal landscape can be grasped from the hybrid optimization result. For the Rastrigin function, although the hybrid algorithm could not obtain the optimum, the approximated landscape provides the rough information of the entire search space. On the other hand, the landscape approximated from the GA results was shallow, and was almost unimodal on all functions. Especially for the Schwefel function, it was obvious that GA could not provide the information of the local optima near the bounds. These observations indicate that GA cannot provide information for the entire space. These results showed that the proposed hybrid optimization approach provides information of the entire search space, and we can roughly grasp the landscape of the problem. In this way, we can easily evaluate the reliability of the obtained solution.

VI. CONCLUSIONS

This paper described how to design hybrid optimization algorithms and also proposed a new hybrid optimization algorithm. In constructing hybrid optimization algorithms, it is important to define the optimization strategy—how the optimization process should be performed, or what types of solution are required. One of the major contributions of this research is the proposal of an optimization strategy—to search the design space uniformly and equally. From this strategy, after the optimum solution is derived, the user can roughly grasp the landscape of the target problem, and also verify the reliability of the optimization results.

Moreover, based on the proposed strategy, a hybrid optimization algorithm using DIRECT, GA, and SQP was proposed and its effectiveness was investigated. To integrate these algorithms, DIRECT was modified to reduce the number of partitions. The termination criteria of each algorithm were also discussed. Through numerical experiments, the proposed hybrid algorithm was shown to have efficient performance, and to provide the information of the landscape.

As future work, to verify its efficiency, it will be necessary to apply the proposed algorithm to various benchmark functions and real-world problems.

REFERENCES

- [1] K. Schittkowski. NLQPL: A FORTRAN-Subroutine Solving Constrained Nonlinear Programming Problems. *Annals of Operations Research*, 5:485–500, 1985.
- [2] David E. Goldberg. Genetic Algorithms in Search. *Optimization, and Machine Learning*, 1989.
- [3] M. P. Vecchi S. Kirkpatrick, C. D. Gelatt. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, 1983.
- [4] Kurt A. Hacker, John Eddy, and Kemper E. Lewis. Efficient Global Optimization Using Hybrid Genetic Algorithms. *Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2002.
- [5] R. Chelouah and P. Siarry. Genetic and Nelder-Mead algorithms hybridized for a more accurate global optimization of continuous multimimima functions. *European Journal of Operational Research*, 6:191–213, 2000.
- [6] C.D. Perttunen Jones, D.R. and B.E. Stuckman. Lipschitzian optimization without the Lipschitz constant. *Journal of Optimization Theory and Applications*, 79(1):157–181, 1993.

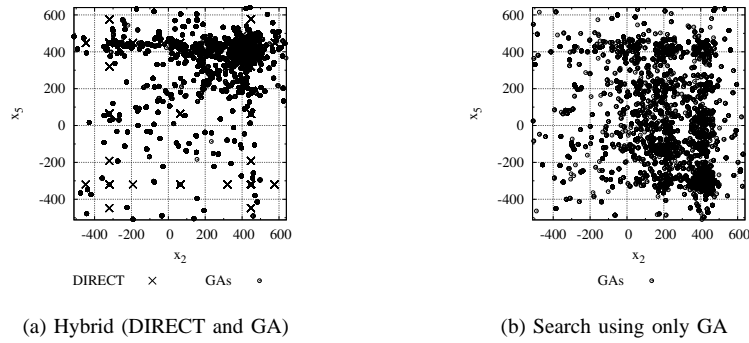


Fig. 8. Search history projected into x_2, x_5 plane.

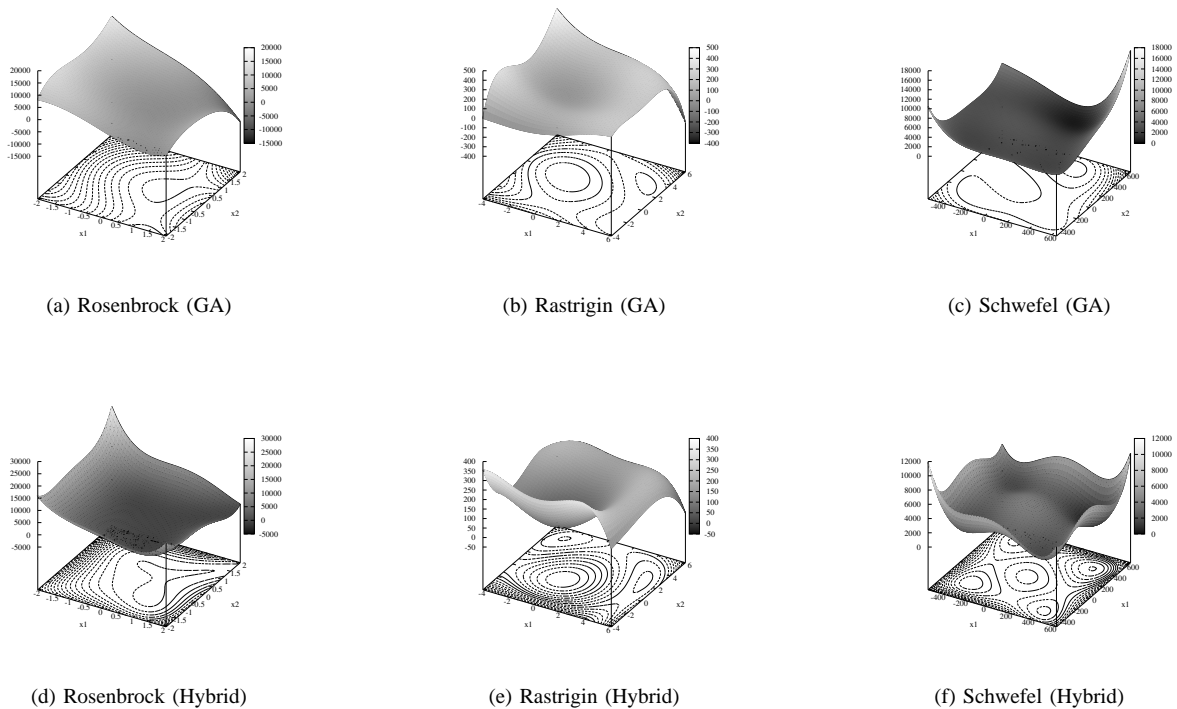


Fig. 9. Results of least-squares fitting

- [7] Mattias Bjorkman and Kenneth Holmstrom. Global Optimization using the DIRECT Algorithm in MATLAB. *Advanced Modeling and Optimization*, 1(2), 1999.
- [8] Steven E. Cox, William E. Hart, Raphael Haftka, and Layne Watson. DIRECT algorithm with Box Penetration for Improved Local Convergence. *Proc. of 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2002.
- [9] L. J. Eshelman and J. D. Schaffer. Real Coded Genetic Algorithms and Interval-Schemata. *Foundations of Genetic Algorithms 2*, pages 187–202, 1993.
- [10] I. Ono and S. Kobayashi. A Real Coded Genetic Algorithm for Function Optimization Using Unimodal Normal Distributed Crossover. *Proc. of the 7th International Conference on Genetic Algorithms*, 1997.
- [11] I. Ono H. Kita and S. Kobayashi. Multi-parental Extension of the Unimodal Normal Distribution Crossover for Real-coded Genetic Algorithm. *Proc. of the 1999 Congress on Evolutionary Computation*, 1999.
- [12] S. Tsutsui T. Higuchi and M. Yamamura. Theoretical Analysis of Simplex Crossover for Real-Coded Genetic Algorithms. *Proc. of PPSN VI*, pages 365–374, 2000.
- [13] M. Yamamura S. Tsutsui and T. Higuchi. Multi-parent recombination with Simplex Crossover in Real-Coded Genetic Algorithms. *Proc. of GECCO 99*, 1999.
- [14] Hisashi Shimosaka, Tomoyuki Hiroyasu, and Mitsunori Miki. Offspring Generation Method Using Delaunay Triangulation for Real-Coded Genetic Algorithms. *Proc. of PPSN IX*, pages 828–838, 2006.
- [15] ADVENTURE Project. <http://adventure.q.t.u-tokyo.ac.jp/>.
- [16] J. He, L. Watson, N. Ramakrishnan, C. Shaffer, A. Verstak, J. Jiang, K. Bae, and W. Tranter. Dynamic Data Structures for a Direct Search Algorithm. *Computational Optimization and Applications*, 23(1):5–25, 2002.
- [17] Hong Zhu and David B. Bogy. Hard disc drive air bearing design: modified DIRECT algorithm and its application to slider air bearing surface optimization. *Tribology International*, 37(2):193–201, 2004.
- [18] J. M. Gablonsky and C. T. Kelley. A locally-biased form of the DIRECT algorithm. *Journal of Global Optimization*, 21:27–37, 2001.