# Optimal Contraction Theorem for Exploration–Exploitation Tradeoff in Search and Optimization

Jie Chen, *Member, IEEE*, Bin Xin, *Student Member, IEEE*,
Zhihong Peng, Lihua Dou, and Juan Zhang

***Abstract*—Global optimization process can often be divided into two subprocesses: exploration and exploitation. The tradeoff between exploration and exploitation (*T:Er&Ei*) is crucial in search and optimization, having a great effect on global optimization performance, e.g., accuracy and convergence speed of optimization algorithms. In this paper, definitions of exploration and exploitation are first given based on information correlation among samplings. Then, some general indicators of optimization hardness are presented to characterize problem difficulties. By analyzing a typical contraction-based three-stage optimization process, *Optimal Contraction Theorem* is presented to show that *T:Er&Ei* depends on the optimization hardness of problems to be optimized. *T:Er&Ei* will gradually lean toward exploration as optimization hardness increases. In the case of great optimization hardness, exploration-dominated optimizers outperform exploitation-dominated optimizers. In particular, random sampling will become an outstanding optimizer when optimization hardness reaches a certain degree. Besides, the optimal number of contraction stages increases with optimization hardness. In an optimal contraction way, the whole sampling cost is evenly distributed in all contraction stages, and each contraction takes the same contracting ratio. Furthermore, the characterization of optimization hardness is discussed in detail. The experiments with several typical global optimization algorithms used to optimize three groups of test problems validate the correctness of the conclusions made by *T:Er&Ei* analysis.**

***Index Terms*—Exploitation, exploration, global optimization, optimal contraction theorem, optimization hardness.**

## I. Introduction

The tradeoff between exploration and exploitation (*T:Er&Ei*) has long been a significant topic in evolutionary computation and optimization. This topic exists in many fields, including machine learning [1]–[7] and cognition [8], [9], modeling and prediction [10], [11], search and optimization [12], [13], and many other cases in which uncertainty exists. In learning algorithms, *T:Er&Ei* is a sticking point, having a great impact on effective and efficient algorithm design [3]. A learning algorithm makes decisions to maximize its rewards—exploitation—but lacks the knowledge about the reward generating process. Thus, occasionally the algorithm might introduce explorations, which improves the knowledge about the reward generating process but not necessarily maximizes the current reward. This problem is the same for search and optimization, particularly for metaheuristics [14], while the latter is usually a more complicated case. For all optimization methods, *T:Er&Ei* is crucial. An effective *T:Er&Ei* helps to reduce computational cost and implement efficient optimization. Early researches on *T:Er&Ei* can be sorted into two categories. The first groups investigated methodologies which relate *T:Er&Ei* to bandit problem. In this problem, a gambler has to decide which arm of a $K$-slot machine to pull to maximize his total reward in a series of trials. The exploration/exploitation tradeoff is reflected on one hand by the necessity of trying all alternatives, and on the other hand by the regret suffered when trying an alternative which is not optimal. Particularly, Holland [12] proposed a least loss criterion to solve two-armed bandit problem as a reference for *T:Er&Ei* in genetic algorithms. A few scholars [15]–[18] contributed to this research and even extended this problem to multiarmed bandit problems. Statistics is the most powerful tool in analyzing these problems, and many strategies of controlling *T:Er&Ei* is based on statistical parameters which can reflect exploration and exploitation to some extent [4]–[6]. The direct control of *T:Er&Ei* belongs to the second category of researches.

*T:Er&Ei* in search and optimization originates from heuristics, particularly those population-based evolutionary algorithms in which diversity is an explicit exhibition of *T:Er&Ei*. It dominates the computational cost and convergence quality of such optimization algorithms. Excessive exploitation will depress diversity and induce premature convergence. Excessive exploration will result in slow convergence. Thus, it is crucial to strike a proper *T:Er&Ei*. The effect of diversity on the premature convergence about genetic algorithms was discussed in [19]. Many strategies of controlling diversity were proposed for various evolutionary algorithms in the past decade [20]–[26]. Particularly, the diversity of solutions in multiobjective optimization is prominent as its goal is usually to find the distributed *Pareto* front in the objective space [13], [27], [28].

In practice, it is common to adjust the tradeoff dynamically by controlling some parameters or indices related to it [4]–[6], [29], [30]. These efforts are devoted to constructing effective and efficient optimization algorithms. However, no explanation of the relationship between *T:Er&Ei* and practical problems has been given in these direct control methods. This actuality is mainly due to the diversity of practical problems and detailed algorithms. Another important reason for the difficulty of describing this relationship is the absence of a formal and uniform definition of exploration and exploitation.

The main objective of this paper is to reveal the relationship between *T:Er&Ei* and some characteristics of problems to be optimized in order to identify the ultimate factor which determines the best *T:Er&Ei*. The rest of this paper is organized as follows. In Section II, definitions of exploration and exploitation are given, according to which the characters of currently prevailing optimization algorithms on exploration and exploitation are briefly analyzed. In Section III, several new concepts about the optimization landscape such as *optimal field* and *optimal supreme cap* are proposed. In this section, we also propose the *Optimal Contraction Theorem* (*OCT*) which can *quantitatively* explain the relationship between *T:Er&Ei* and optimization problems. In Section IV, we make some experimental analysis of the *OCT* for pulse functions and several common test functions. Effects of exploration and exploitation on the search process in optimization algorithms are analyzed in this section. Conclusions are made in Section V.

## II. Algorithm Characterization Based on the Definitions of Exploration and Exploitation

A latent viewpoint interprets exploration and exploitation as global search and local search, respectively [13], [29], [31], [32]. This, indeed, reflects one aspect of their characters; howbeit, it is not reliable to distinguish exploration and exploitation only by search scope, because there is no undisputed threshold to define the scale of "local". An unambiguous but meaningless measure is to strictly restrict exploration to keep within the whole search space, and any search

Manuscript received November 23, 2007; revised September 10, 2008. First published February 6, 2009; current version published April 17, 2009. This work was supported by the Open Issues Foundation under Grant 20060104 of the Key Laboratory of Complex Systems and Intelligent Science, Institute of Automation, Chinese Academy of Sciences. This paper was recommended by Associate Editor A. Bargiela.

The authors are with the Department of Automatic Control, School of Information Science and Technology, and the Laboratory of Complex System Intelligent Control and Decision, Ministry of Education, Beijing Institute of Technology, Beijing 100081, China (e-mail: brucebin@bit.edu.cn; chenjie@bit.edu.cn).

Digital Object Identifier 10.1109/TSMCA.2009.2012436

within a shrunk space will be considered as exploitation. In learning algorithms, exploration and exploitation correspond to the acquisition and utilization of knowledge, respectively [3]. Some scholars think that exploration and exploitation can be distinguished by the intensity of randomness [6]. Here, we regard exploration and exploitation as two kinds of behavior in acquiring information about unknown problems. For unification, the information acquisition process is treated as a sampling process which gradually turns a "black-box" model for optimization problems into a "white-box."

*Definition 1—exploration (er):* A sampling behavior is exploration iff its sampling point is generated independent of the information acquired by historical sampling points. Mathematically, a sampling $X_n$ at time $n$ is exploration iff $P(X_n = X|X_{n-1}, \ldots, X_0) = P(X_n = X)$.

*Definition 2—exploitation (ei):* A sampling behavior is exploitation iff the generation of its sampling point depends upon the information acquired by historical sampling points. Mathematically, a sampling $X_n$ at time $n$ is exploitation iff $P(X_n = X|X_{n-1}, \ldots, X_0) \neq P(X_n = X)$.

In the definition of exploitation, the utilization of historical information about the function to be optimized is emphasized. Typically, random sampling (*RS*) without preference for any region in search space is completely explorative, because sampling points generated by this method are independent of each other. In contrast, the steepest ascent optimizers (*SAO*) use gradient-based information at a point to generate the next sampling point, so they are completely exploitative. In detail, there are various forms of exploitation in the implementation of optimization algorithms. For example, simulated annealing (*SA*) [33] and genetic algorithms with proportional selection (*GA:PS*) [12] use differences between the function values at different points (named as chromosome in *GA*) to adjust the sampling distribution. In particle swarm optimizer (*PSO*) [34], differential evolution (*DE*) [35], ant colony optimizer (*ACO*) [36], estimation of distribution algorithms (*EDA*) [37], and genetic algorithms with ranking selection (*GA:RS*) [38], the sampling is controlled according to the order in function landscapes. In other words, these algorithms only care about whether a point is superior to another or not, regardless of their quantitative differences. The algorithms previously mentioned except *RS* utilize historical information in a normal manner, as they tend to concentrate search on those regions in which better solutions may hopefully exist. It is not unusual that abnormal exploitation (*AnEi*) exists in some algorithms [39]–[41]. Repulsion [40], [41] and anticonvergence [42] typically belong to *AnEi*. In essence, they are the same. Obviously, the effect of *AnEi* on optimization is similar to that of *Er*. Both *AnEi* and *Er* contribute to increasing the probability of escaping from local optima. With a proper structure, *AnEi* helps to explore the undiscovered promising regions in search space. Therefore, in a sense, *AnEi* can be regarded as a special exploration. Particularly, the crossover and mutation operators in *GA* belong to *Ei* as they use the genic materials from parent chromosomes to produce child chromosomes, but it is not easy to determine whether they are normal exploitations (*NEi*) or not. This is because they cannot ensure the preservation of excellent genes without proper operation positions. A normal selection operator, such as proportional selection and ranking selection, tends to preserve most of the excellent genes in the whole population. Note that, for clarity, the term *Ei* mentioned later is *NEi*. In general, the role of *Er* in the search process is to acquire information, and that of *Ei* is to utilize the information.

*Er* and *Ei* coexist in most optimization algorithms, particularly in global optimizers. Therefore, these optimizers cannot be classified as *Er* or *Ei* directly. However, the *Er/Ei* ratios in different optimizers appear to vary. As two special cases, *RS* with the preservation of optimal solutions can be directly classified as *Er*, and *SAO* as *Ei*. However, this is not the case in *SA, GA, PSO, ACO*, and many other optimizers. Intuitively, *SA, GA*, and *ACO* are more explorative than *PSO* since all of them base their optimization on the adjustment of the probability distributions of candidate solutions, which can be characterized by Markov chains [43]–[45]. In contrast, *PSO* is an exploitation-dominated optimizer as indicated by its iteration expressions [34]. Without two random numbers introduced into its velocity expression, it will become a deterministic optimizer [46]. The explorative randomness attached to PSO's evolutionary framework just brings more occasions of escaping from the local optima.

The combination of *Er* and *Ei* varies in different optimizers. In *TRUST* [40], *Er* and *Ei* are executed alternately. Each time an *SAO* is executed to find a local optimum, a "tunneling" process follows to further improve the temporal optimum. However, there is no distinct order between *Er* and *Ei* in *PSO* and *GA*. Generally, they are mingled with each other. Since there are so many different optimizers available for us to choose in practical optimization, a logical question emerges: What kind of *Er/Ei* ratio and combination pattern is the best for various function landscapes? In fact, we are not unfamiliar to this question, as it is often discussed what algorithm is the best one for a particular problem. The *No Free Lunch* theorem [47] indicates that no optimizers can keep optimal when optimizing arbitrary problems; however, it does not give a direct answer. The answer to this question depends on the landscape of objective functions. For simplicity, without loss of generality, we focus on the unconstrained maximization of functions in bounded space with finite dimensions in the form of $\max_{\mathbf{x} \in \mathcal{B}} f(\mathbf{x})$, where $\mathcal{B}$ is an $n$-dimensional *Borel* field, $\mathbf{x}$ is a vector corresponding to a point in $\mathcal{B}$, and $f(\mathbf{x})$ denotes the function value corresponding to $\mathbf{x}$.

## III. Optimal Contraction Theorem

As a preparation for the analysis of optimization process, we extract some useful features of optimization landscapes to characterize the optimization hardness of objective functions.

*Definition 3—optimal field (of):* A subspace $\mathcal{D}^*$ in the whole search space $\mathcal{B}$ is optimal field iff both of the following conditions are satisfied.

1) Arbitrary point in $\mathcal{D}^*$, denoted by $\mathbf{x}$, satisfies $f(\mathbf{x}) > f(\mathbf{x}')$, where $\mathbf{x}'$ is a local optimum which is only inferior to global optima among all local optima.
2) Arbitrary point in $\mathcal{B}$ satisfying the inequality $f(\mathbf{x}) > f(\mathbf{x}')$ is involved in $\mathcal{D}^*$.

*OF* can be denoted by the level set $\{\mathbf{x}|\forall \mathbf{x} \in \mathcal{B} : f(\mathbf{x}) > f(\mathbf{x}')\}$. Note that global optima are special local ones. *OF* does not contain any local optima except global optima. If there is only one global optimum in a function landscape, the section of this landscape corresponding to optimal field is unimodal. A unimodal landscape is steepest ascent optimizable. An illustration for optimal field and other concepts as follows is shown in Fig. 1.

*Definition 4—strict optimality ratio (sor):* The strict optimality ratio of a function landscape is the ratio of the measure of its optimal field [denoted by $\sigma(\mathcal{D}^*)$] to that of the whole search space [denoted by $\sigma(\mathcal{B})$].

*Definition 5—optimal supreme cap (osc):* For each isolated global optimum (denoted by $\mathbf{x}^*$), its optimal supreme cap $C^*(\mathbf{x}^*)$ is the section of function landscape that includes all points for which there exists a steepest ascent path to reach the global optimum. Particularly, if many global optima gather together to form a continuous field, they will be regarded as an ensemble, and the unique *OSC* corresponding to the ensemble contains all points that can reach at least one optimum in the cap along a steepest ascent path. *OSC* is similar to the concept of the attraction basin of local optima for minimization problems.

*Definition 6—optimality ratio (or):* The optimality ratio of a function landscape is the ratio of the measure of its *OSC* [denoted by

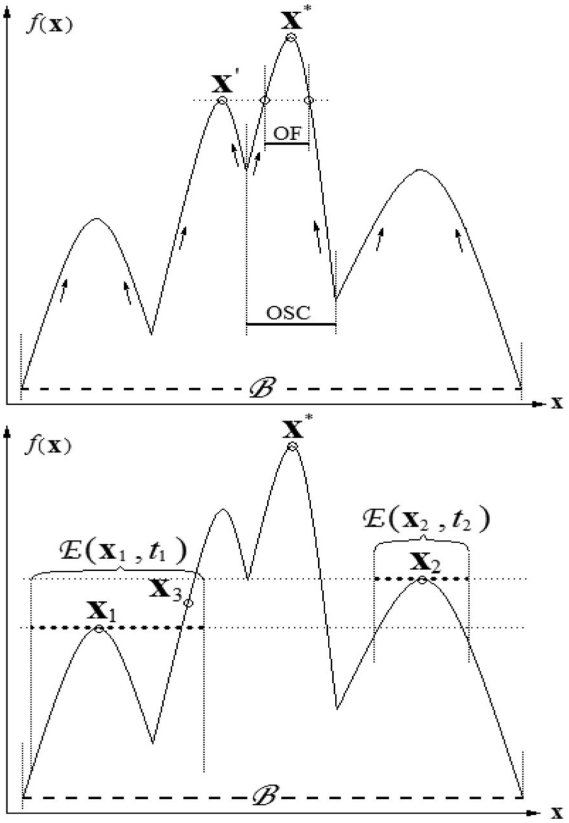

Fig. 1. Illustration for optimal field (*OF*), optimal supreme cap (*OSC*), and $p$-inducive point. $\mathbf{x}'$ is a local optimum which is only inferior to the global optimum $\mathbf{x}^*$. $\mathbf{x}_1$ and $\mathbf{x}_2$ are local optima, too. When the search reaches $\mathbf{x}_1$ at time $t_1$, it can move from $\mathbf{x}_1$ only to the points in $\mathcal{E}(\mathbf{x}_1, t_1)$ which is the reachable region next time. It is easy to see that $\mathcal{E}(\mathbf{x}_1, t_1) \cap \mathcal{R}(\mathbf{x}_1) \neq \Phi$ where $\mathcal{R}(\mathbf{x}_1)$ is the level set corresponding to $\mathbf{x}_1$ and defined as $\{\mathbf{x} | \forall \mathbf{x} \in \mathcal{B} : f(\mathbf{x}) > f(\mathbf{x}_1)\}$. For example, as shown above, $\mathbf{x}_3 \in \mathcal{E}(\mathbf{x}_1, t_1) \cap \mathcal{R}(\mathbf{x}_1)$, that is to say, it is possible that a point better than $\mathbf{x}_1$ could be found. If the improvement of the detected objective value can continue and the global optimum $\mathbf{x}^*$ is found at last in a probability not less than $p$, then $\mathbf{x}_1$ can be regarded as a $p$-inducive point. In contrast, when the search reaches $\mathbf{x}_2$ at time $t_2$, the detected objective value cannot be further improved due to $\mathcal{E}(\mathbf{x}_2, t_2) \cap \mathcal{R}(\mathbf{x}_2) = \Phi$. Therefore, it can be confirmed that $\mathbf{x}_2$ is not a $p$-inducive point. All $p$-inducive points form the $p$-inducive field (*piF*).

$\sigma(C^*)$] to that of the whole search space [denoted by $\sigma(\mathcal{B})$]. Note that $C^* = \sum C^*(\mathbf{x}), (\mathbf{x} \in O^*)$, where $O^*$ is the set of all global optima. Obviously, *OF* $\subseteq$ *OSC* and *SOR* $\leq$ *OR*. Besides, *OF* = *OSC* and *SOR* = *OR* iff the whole landscape is unimodal.

For exploitation-preferred optimizers, e.g., *TRUST, OR* is a reasonable indicator of optimization hardness, because the steepest ascend optimizer (*SAO*) embedded in them can find a global optimum once the search enters *OSC*. However, for greedy search that only favors the currently best point (denoted by $\mathbf{x}^\circ$), even if the search enters *OSC*, a point outside *OSC* that is superior to $\mathbf{x}^\circ$ will be able to attract the search to break away from *OSC*. Only if the search enters *OF* could a global optimum be definitely found by an *SAO*. Thus, *SOR* is a reasonable characterization of the optimization hardness of function landscape with greedy search. As validated by the experiments in Section IV, *OR* and *SOR* are suitable for those landscapes without obvious macrocharacters such as "needle-in-haystack" landscapes.

In order to make a general characterization of optimization hardness, we propose three novel concepts as follows.

*Definition 7—p-inductive field (pif):* For an optimizer, when its search reaches a certain point (denoted by $\mathbf{x}_0$) at time $t_0$, and if it can move from $\mathbf{x}_0$ to a global optimum after certain steps with a probability not less than $p$, then $\mathbf{x}_0$ is called *p-inductive point*. The probability $p$ depends on not only landscapes but also optimizers. An optimizer has to move from $\mathbf{x}_0$ to a better point at first before it could reach a global optimum. Thus, the improvement probability at $\mathbf{x}_0$ becomes a predeterminate factor, which can in the sense of uniformity be expressed by

$$p(\mathbf{x}_0, t_0) = \sigma\left(\mathcal{E}(\mathbf{x}_0, t_0) \cap \mathcal{IR}(\mathbf{x}_0)\right) \big/ \sigma\left(\mathcal{E}(\mathbf{x}_0, t_0)\right)$$

where $\mathcal{E}(\mathbf{x}_0, t_0)$ is the set of the points to which the optimizer may move from $\mathbf{x}_0$, and $\mathcal{R}(\mathbf{x}_0)$ is a level set expressed by $\{\mathbf{x} | \forall \mathbf{x} \in \mathcal{B} : f(\mathbf{x}) > f(\mathbf{x}_0)\}$. Particularly, if $p(\mathbf{x}_0, t_0) = 0$, that is $\mathcal{E}(\mathbf{x}_0, t_0) \cap \mathcal{R}(\mathbf{x}_0) = \Phi$, then it is impossible to reach a global optimum from $\mathbf{x}_0$. All *p-inducive points* constitute the so-called *p-inductive field.*

*Definition 8—p-inductive optimality ratio (pior):* For an optimizer and a function landscape, the $p$-inductive optimality ratio corresponding to them is the ratio of the measure of *piF* [denoted by $\sigma(\text{piF})$] to that of the whole search space [denoted by $\sigma(\mathcal{B})$]. Generally, *OSC* $\subset$ *piF* and *OR* < *piOR*.

Relatively speaking, for the characterization of optimization hardness, *SOR* and *OR* are static, and *piOR* is dynamic. The analysis of *SOR* and *OR* is easier than that of *piOR* for complicated landscapes or optimizers.

*Definition 9—optimization feature factor (off):* Optimization hardness indicator is crucial to *T:Er&Ei*, and it dominates *T:Er&Ei* to a great extent. For unification, we call the indicator *optimization feature factor* (*OFF*) to represent *SOR, OR, piOR*, or other reasonable measures.

In essence, *OFF* provides a quantitative characterization of the distribution of an "easy" local region within the whole search space. Here, the word "easy" means that it is easy for a specific optimizer to find global optima once the search enters the local region. For example, unimodal problems are often regarded as a kind of easy problems for most optimizers. Their landscapes may become a local part of the landscape of another problem whose global optima may be contained in the local region. This is the same case for other "easy" problems. Therefore, *OFF* can be regarded as a generalized measure of the distribution of the "easy" local region which contains global optima in the whole search space. The optimal field (*OF*) and the optimal supreme cap (*OSC*) are just special cases of such an "easy" local region. In this sense, the characterization of optimization hardness in the form of *OFF* depends on its opposite—the characterization of landscapes which are easy for specific optimizers to optimize.

When search space is very large particularly in high-dimensional function optimization, it is impossible to take an exhaustive search due to unbearable time consumption. Therefore, the contraction of search scope is necessary to exclude those regions which are unlikely to contain global optima. In fact, the direct effect of normal exploitation in search and optimization is just the contraction of search scope. For many convergent optimizers, search scope shrinks to one point or a point set at last. Although shrinkage and dilation may coexist in the search process, what we discuss here is mainly a continuous shrinkage process in which every contracted space includes *OF, OSC*, or other easy local regions with a certain probability. The objective function considered here is pulse function with its expression as follows:

$$f(\mathbf{x}) = \prod_{i=1}^{n} \left(A_i \cdot [\text{Sign}(x_i - a_i) - \text{Sign}(x_i - b_i)]\right)$$

$$\mathbf{x} = [x_1, x_2, \ldots, x_n]^{\mathrm{T}} \in \prod_{i=1}^{n} (c_i, d_i),$$

$$A_i > 0,\ c_i < a_i < b_i < d_i \text{ for all } i \qquad (1)$$

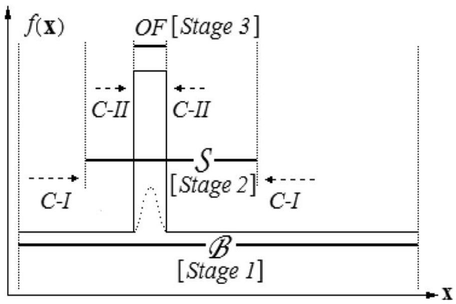

Fig. 2. Illustration for three-stage search. $C - I$ and $C - II$ represent the first contraction and the second contraction, respectively. $C - I$ occurs after the first stage and the search scope is reduced from $\mathcal{B}$ to a lesser local space $S$, as indicated by the arrows above the symbols "$C - I$." $C - II$ occurs after the second stage, and the search enters *OF* at this time. For the pulse function described by (1), the third stage is unnecessary since each point in *OF* is a global optimum. However, if there is only one global optimum in *OF* as shown by the dashed curve in the pulse, then the exploitative search at the final stage like gradient-based search is usually indispensable.

where $\mathrm{Sign}(x) = +1, 0, -1$, respectively for $x > 0$, $x = 0$, and $x < 0$

$$f(\mathbf{x}) = \begin{cases} 2^n \prod_{i=1}^{n} A_i, & \text{if } a_i < x_i < b_i \text{ for all } i \\ 2^{n-m} \prod_{i=1}^{n} A_i, & \text{if } \exists i : \mathrm{num}(i : x_i = a_i \text{ or } b_i) = m \\ & \text{and } a_i < x_i < b_i \text{ for all other } i \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $\mathrm{num}(i : C)$ denotes the number of $i$ that satisfies the condition $C$.

The pulse function is also called *needle-in-a-haystack* problem for the counterpart in the research on genetic algorithms [38]. There are three reasons for us to use the pulse function as the representative objective function in our research on *T:Er&Ei*. First, any functions can be accurately or approximately represented as a combination of different pulse functions. The analysis on the optimization of pulse functions is a foundation for the research on the optimization of general functions. Second, from the viewpoint of locating *OF, OSC, piF*, or other local search spaces which benefit a specific optimizer, the primary cost of optimizing an arbitrary function is equal to that of optimizing a certain pulse function which shares some similarity with the arbitrary function. This is validated by the experiment on equivalent optimization hardness in Section IV. Therefore, the conclusions below are not limited to the pulse function. Finally, it is easy to get an unambiguous optimization feature factor of pulse functions for the quantitative analysis later.

The optimal field of the pulse function can be expressed by $\{\mathbf{x} | a_i < x_i < b_i, i = 1, \ldots, n\}$. Its *OFF* (*OR* or *SOR*) can be expressed by $\prod_{i=1}^{n}(b_i - a_i)/(d_i - c_i)$. It is obvious that concentrative search in any local space beyond *OF* is blind and easily leads to premature convergence. Therefore, *Er* should be prior to *NEi* to locate *OF, OSC*, or *piF* and lay the foundation for *NEi*. According to this principle, the whole search process is divided into three stages. At the first stage, the whole search space is explored and, finally, contracted to a lesser space that covers *OF* with a certain probability. At the second stage, *Er* continues in the lesser space, and finally, the search enters *OF*. At the final stage, *NEi* comes into effect, and a global optimum is found in *OF* at last. The final stage is common in most optimizers. In a sense, the final search is similar to the optimization of a *quasi-concave* landscape or a landscape with *quasi-concave envelop* in which abundant useful information can be normally exploited. Relatively speaking, the search cost at this stage is inevitable, and many local optimizers can be used at this stage. An illustration for the three-stage search is shown in Fig. 2.

For many optimizers such as *GA* [48] and *PSO* [49], the exploration at the first stage, usually in the form of random initialization, has a great impact on their following exploitative behaviors and optimization results. Since any concentrative search beyond *OF* easily leads to premature convergence, uniform random sampling is preferred in the explorative stages. Furthermore, we focus on the cost of locating *OF* that is the cost of the first two stages. For brevity, denote *OFF* that is $\prod_{i=1}^{n}(b_i - a_i)/(d_i - c_i)$ by $\eta$. In this case, the cost of the final stage, that is the cost of exploitation, is zero since each point in *OF* is globally optimal. Therefore, *NEi* is needless here.

After $m$ times of sampling and evaluating at the first stage, the probability that the search enters a contracted space, denoted by $\mathcal{S}$, can be expressed as $P_1 = 1 - (1 - 1/M_1)^m$ where $M_1 = \sigma(\mathcal{B})/\sigma(\mathcal{S})$ is the contracting ratio of the first contraction. Note that the contracted space is expected to contain *OF*. After $n$ times of sampling and evaluating in $\mathcal{S}$ at the second stage, the probability that the search enters *OF* can be represented as $P_2 = 1 - (1 - M_1\eta)^n$. The contracting ratio of the second contraction is $M_2 = (M_1\eta)^{-1}$. Thus, through the first two stages, the probability that the search enters *OF* is $P = P_1 \cdot P_2 = [1 - (1 - 1/M_1)^m][1 - (1 - M_1\eta)^n]$. At the third stage, the search will continue in *OF* for locating a global optimum. The cost of locating *OF* can be given by $T = m + n$. Since this cost is generated at the explorative stages, it can be regarded as the cost of exploration. Thus, one can get the following cost function:

$$\begin{aligned} T(M_1, n) &= n + \ln\{1 - P/[1 - (1 - M_1\eta)^n]\} \\ &\quad / \ln(1 - 1/M_1) \\ \mathcal{D}_T &= \{(M_1, n) \mid 1 \le M_1 \le 1/\eta, \\ &\qquad n \ge \ln(1 - P)/\ln(1 - M_1\eta)\} \end{aligned} \tag{3}$$

where $\mathcal{D}_T$ is the definition domain of $T(M_1, n)$, $\ln(\cdot)$ is natural logarithm, and $P$ can be regarded as the confidence level of entering *OF*. Naturally, the following minimization problem comes forth:

$$\min T(M_1, n), (M_1, n) \in \mathcal{D}_T. \tag{4}$$

The detailed analysis of this optimization problem is presented in Appendix A. One of the most important analysis results is that the optimal contracting ratios of the two contractions are equal, that is to say $M_1^* = M_2^*$. Furthermore, a generalized $(k + 1)$-stage contraction process is considered. Similar to the aforementioned three-stage contraction process, the final stage, in this case, is also exploitative. Since the anterior $k$-stage contraction subprocess can be viewed as $(k - 1)$ continuous two-stage contraction subprocesses, any two neighboring contractions should have the same contracting ratio in the best case. In other words, the optimal contraction pattern should be $M_1^* = M_2^* = \cdots = M_k^*$. Besides, $\prod_{i=1}^{k} M_i^* = 1/\eta$. Therefore, it can easily be derived that $M_i^* = (1/\eta)^{1/k}$ for $\forall i \in \{1, 2, \ldots, k\}$. In this case, the minimal cost of locating *OF* for the $(k + 1)$-stage contraction process is $T(k) = k \ln(1 - P^{1/k})/\ln(1 - \eta^{1/k})$. The analysis of $T(k)$ is presented in Appendix B.

The following theorem sums up the results of the analysis regarding the cost functions $T(M_1, n)$ and $T(k)$.

*Theorem 1—optimal contraction theorem:*

1) $OFF(\downarrow) \Rightarrow k^*(\uparrow)$

$$k^* = \arg\min\left\{T\left(\lceil -\log_2(\eta)\rceil + 1\right), T\left(\lceil -\log_2(\eta)\rceil\right)\right\}$$

where $k^*$ is the optimal number of contraction stages (the final exploitative stage not included), and $\lceil x \rceil$ denotes the biggest integer which is not greater than $x$. More stages are preferred as optimization hardness increases.

2) $(3-\sqrt{5})/2 \leq OFF < 1 : k^* = 1$

In the special case where $(3-\sqrt{5})/2 \leq OFF < 1$, only one contraction stage is needed for locating OF. This corresponds to a very easy optimization case for specific optimizers.

3) $OFF(\downarrow) \Rightarrow \min T(\uparrow) : \min T = T(k^*)$

The minimal cost of locating OF or other "easy" regions for specific optimizers increases as OFF is reduced, which means that OFF is an indicator of optimization hardness. (Strictly, the indicator should be $1 - OFF$.) The proportion of exploration in the whole samplings increases with the optimization hardness of objective functions.

4) $(OFF < 1 \,\&\, P \to 1) \Rightarrow (\min T \to \infty)$

For stochastic contraction, strict global optimization cannot be guaranteed if $OFF \neq 1$. In the extreme case where $OFF = 0$, global optimization is impossible for any optimizers.

5) $M_1^* = M_2^* = \cdots = M_{k^*}^* = (1/\eta)^{1/k^*}, \qquad T_1 = T_2 = \cdots = T_{k^*} = \ln(1 - P^{1/k^*})/\ln(1-\eta^{1/k^*})$

In the best case, all contractions have the same contracting ratio which is determined by optimization feature factor. Besides, all stages have the same cost. In conclusion, the uniform contraction pattern is optimal.

For most of the prevailing optimizers, the contraction process is discontinuous, and dilation may alternate with shrinkage. Generally, dilation is regarded as an explorative behavior. In this case, the contraction process can be represented by

$$CP = \prod_{i=1}^{N} \left[ \left( \prod_{k=1}^{w(i)} W_{i,k} \right) \bigg/ \left( \prod_{j=1}^{m(i)} M_{i,j} \right) \right] \tag{5}$$

where the whole process is divided into $N$ stages and each stage includes a continuous shrinkage subprocess denoted by $M_{i,j}\{j = 1, \ldots, m(i)\}$ and a continuous dilation subprocess denoted by $W_{i,k}\{k = 1, \ldots, w(i)\}$. $M_{i,j}$ and $W_{i,k}$ are, the shrinkage ratio of the $j$th shrinkage and the dilation ratio of the $k$th dilation at the $i$th stage, respectively. Both of them are real numbers not less than one. Convergent optimizers will converge to one point or a point set including finite elements, and their measures in the *Euclidean* space are both zero. Therefore, $\lim_{N\to\infty} CP = 0$ and shrinkage dominates the contraction process. Accordingly, the *Optimal Contraction Theorem* remains valid for the generalized case.

In the aforementioned process of locating *OF*, contraction is in essence a kind of momentary exploitation that is based on the explorative sampling before contraction. Therefore, strictly speaking, the anterior $k$-stage contraction subprocess also has certain degrees of exploitation. However, the way of contraction is not given above. Typically, a viable way is to exclude inferior evaluated points and even the small local regions that, respectively, include these points [50]–[52]. Such exclusion-based exploitative behavior appears prudential and conservative. For comparison, an intensified exploitative behavior emphasizes particularly on superior evaluated points, and more sampling points will be generated in the local regions which include these superior points. Generally speaking, the exclusion-based exploitation is slower in exploiting local landscape information, but it is beneficial to prevent premature convergence and locate *OF* when optimization hardness is great. As a particular exclusion-based exploitation, exhaustive search is usually considered to be inefficient when search space is very huge. However, exhaustive search is the unique method that can guarantee global optimization within the whole range of optimization hardness except for $OFF = 0$. When optimization hardness is extremely great, the exhaustive search without repetitive samplings becomes "efficient", and an enormous cost is inevitable.

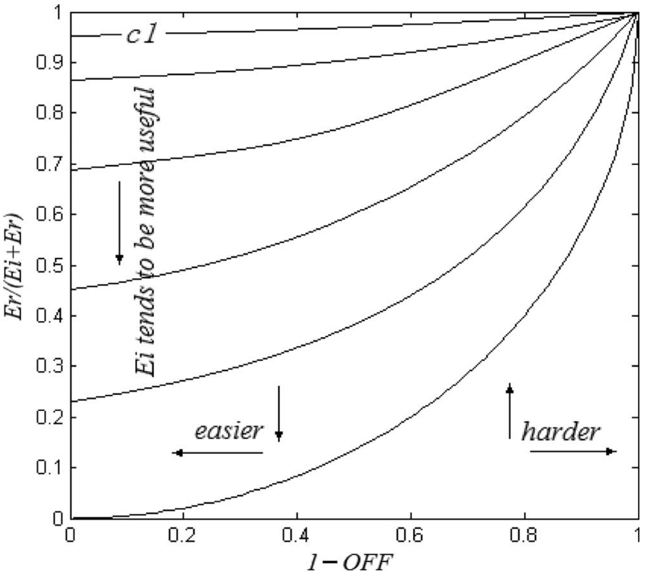

Fig. 3. $1 - OFF$ is a direct quantitative characterization of optimization hardness. *Er/(Ei + Er)* denotes the portion of efficient exploration in sampling, which reflects *T:Er&Ei*. The starting point of each curve, which corresponds to $OFF = 1$, reflects the portion of efficient exploration in locating global optima within *OF, OSC, piF*, or other local regions which benefit specific optimizers to achieve global optimization.

As *OFF* decreases and optimization hardness increases, the amount and proportion of exploration in sampling usually increases, and the best sampling behavior approaches that of random sampling gradually. Particularly, when a pulse function is optimized with finite resolution and only one discrete point lies in the pulse, no exploitative behavior is rational and the whole sampling process should be explorative. If the number of discrete points corresponding to the applied resolution is very large, the optimization hardness of this pulse function is great. In such discrete case, the measure of search space is actually the number of points involved in the discrete space. In any case, *T:Er&Ei depends on optimization hardness*. Their correspondence relation is roughly shown in Fig. 3. For pulse functions, once search enters *OF*, at least one global optimal point will be found, and no further exploitation is needed if it is not required to find all optimal solutions. Thus, the correspondence relation curve for different pulse functions approaches the curve $c1$ in Fig. 3. In this *T:Er&Ei* spectrum, each point corresponds to a tradeoff point for the optimization of some functions. Particularly, the tradeoff points for quasi-concave functions (or wide pulse functions) and needle-in-haystack functions (or narrow pulse functions) lie to the left side and the right side, respectively. For example, the optimization of the *Sphere* function (see Experiment III in Section IV) corresponds to a certain point near the lower left-hand corner of the *T:Er&Ei* spectrum. This is because there is much useful information in its landscape and the exploitative gradient-based search will greatly benefit from this.

For the optimal contraction theorem, a reasonable definition of *OFF* is crucial. *OR* and *SOR* are just suitable for pulse functions and their analogs [such as the function shown at the top of Fig. 4(a)]. As shown in Fig. 4(b), adding some "noise" with smaller magnitude beyond *OF* into the pulse function will not obviously affect its optimization hardness for many optimization methods, which will be validated by the experiments in Section IV. A simple variation is shown below the pulse function in Fig. 4(b). Different from pulse function, exploitation is necessary for this variation after the search enters *OF*. This is because a successful locating of *OF* does not imply the discovery of any global optimal point. However, these function landscapes have similar optimization hardness. For many optimizers, e.g., *PSO, OR*,

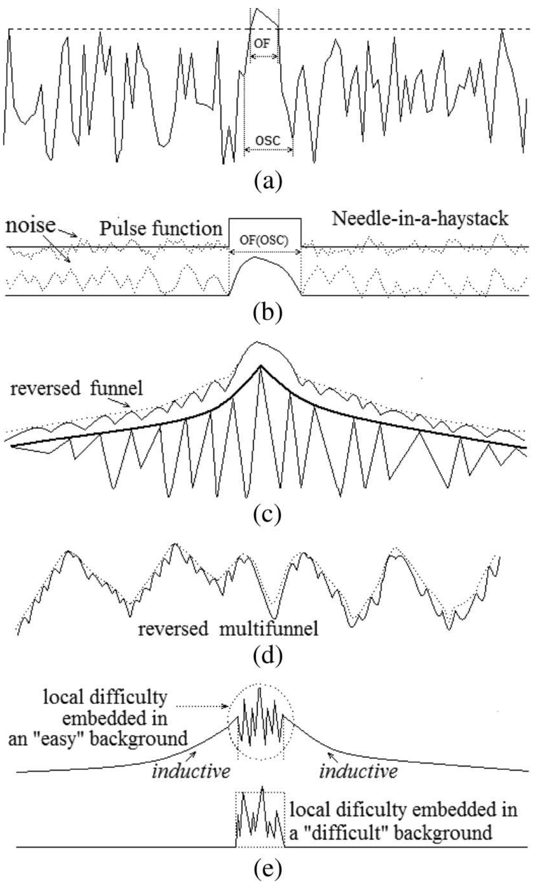

Fig. 4. Characterization of optimization hardness.

and *SOR* are not exact indicators of optimization hardness of the reserved funnel function shown in Fig. 4(c), because these optimizers can easily optimize the *funnellike* function [49]. In this case, the optimization hardness should be explained by new promising concepts like *piOR*.

The curve plotted by a thicker solid line in the middle of Fig. 4(c) represents a typical quasi-concave function landscape which is easy for most optimizers. It can be optimized by *SAO* with guaranteed global convergence. Below this curve is a relatively complicated landscape. Its fluctuation is drastic, which seems to make the rugged landscape difficult to optimize. However, the distribution of its local optima is regular, and all of them lie on the above quasi-concave curve which can be regarded as the envelope of this fluctuant curve. Besides, compared with the whole search scope, the distance between neighboring local optima is not too long. All these features are the same with the reversed funnel function in Fig. 4(c). In contrast to the curve in Fig. 4(a), they contribute to reducing the optimization hardness that the fluctuations introduce into these landscapes. If the distance between the global optimal point and its neighboring local optima is extended to some extent, the corresponding landscapes will relatively become difficult to optimize. This is because the improvement probability at the local optima around the global optimal point will be reduced in this case, and many optimizers tend to be trapped in these local optima. The concept of *p-inducive* point above is useful to explain this phenomenon. In addition, multifunnel landscapes are usually considered as quite difficult cases [49], [53]. A reversed multifunnel landscape is shown in Fig. 4(d). Its optimization hardness can be roughly denoted by $1 - \sigma(\mathcal{F})/\sigma(\mathcal{B})$, where $\sigma(\mathcal{B})$ is the measure of the whole search space and $\sigma(\mathcal{F})$ is that of the funnel including the global optimal point. In a sense, $\sigma(\mathcal{F})/\sigma(\mathcal{B})$ can be taken as an alternative of intrinsic and static *OFF*. The top envelop of this landscape is plotted with a dashed curve, and its optimization hardness is close to that of the reversed multifunnel landscape.

The two curves in Fig. 4(e) are more complicated as their optimization hardness levels are embedded. The upper curve has local difficulty embedded in an "easy" background. The quasi-concave background helps to induce optimizers to approach *OF* so that exploitative samplings will rapidly concentrate on the regions near *OF*. Furthermore, explorative samplings are expected to be implemented on a rugged local landscape. Accordingly, a rough but reasonable representation of the optimization hardness for this landscape is $1 - \sigma(OF)/\sigma(\mathcal{L})$ or $1 - \sigma(OSC)/\sigma(\mathcal{L})$ where $\mathcal{L}$ is the local region covered by a dashed circle in Fig. 4(e). For the bottom curve, the points in the local region (denoted by $\mathcal{E}$) covered by a dashed square have closer relations because all of these points are superior to the points beyond this region. Once search enters this region, more samplings will be attracted into it, which may help to increase the probability of finding *OF* or *OSC*. Thus, the optimization hardness of this landscape, denoted by OH, can be estimated as $1 - \sigma(\mathcal{E})/\sigma(\mathcal{B}) < OH < 1 - \sigma(OF)/\sigma(\mathcal{B})$ or alternatively, $1 - \sigma(\mathcal{E})/\sigma(\mathcal{B}) < OH < 1 - \sigma(OSC)/\sigma(\mathcal{B})$. More general landscapes could be the combinations of the above landscapes.

Note that the optimization hardness analysis of a landscape for different optimizers may be implemented in different spaces. Generally, different operators correspond to different landscapes [54]. This mainly affects the neighborhood in search space, which may change optimization hardness. In addition, the improvement probability at the same point in a landscape for different optimizers is usually different. The optimization hardness analysis above is static and intrinsic for any landscapes if only one optimal point or optimal function value is required. If more or all optimal points are required, the distribution of optima should be considered. In addition, as mentioned in [55], a detailed problem hardness analysis for different optimizers may be dynamic.

In practice, there may exist rather difficult optimization problems particularly in the field of biology, life sciences and chemistry [56], [57]. From the viewpoint of optimizing arbitrary functions, both deterministic and stochastic optimizers are stochastic since the optimization results and costs of optimizing various functions may be distinct. Hence, unless the level of optimization hardness can be determined or optimization problems are not difficult enough, optimal optimization methods can never be constructed. It has been proven that no omnipotent method can best solve all optimization problems with invariant structures and parameters [47]. Some special strategies such as interdimension cooperation [58] could reduce the computational complexity of optimizing some problems. This is because they utilize some particularities of special problems (e.g., separable problems), which can be regarded as an implicit form of exploitation. For interdimension cooperation, if the independent optimization process in each dimension cannot benefit subsequent independent search to approach global optimal points, this strategy will become ineffective. However, it is very useful for some particular problems, although separable problems as a special class suitable for this strategy to deal with are not suggested as a quite reasonable test benchmark [59]. As indicated by Macready and Wolpert [15], incorporating domain knowledge into constructing effective optimization strategies is vital. It can in a manner reduce optimization hardness or give an indication about *T:Er&Ei*.

## IV. Numerical Experiments, Analysis, and Discussion

In order to verify the aforementioned conclusions, six typical optimizers are selected to make the following experiments. They are *PSO,*

TABLE I
PARAMETER SETTINGS FOR SIX DIFFERENT PULSE FUNCTIONS

| Parameters | $f_{\mathrm{p}1}$ | $f_{\mathrm{p}2}$ | $f_{\mathrm{p}3}$ | $f_{\mathrm{p}4}$ | $f_{\mathrm{p}5}$ | $f_{\mathrm{p}6}$ |
|---|---|---|---|---|---|---|
| *a* | 11.23 | 65.81 | 43.49 | 67.32 | 55.28 | 55.280 |
| *b* | 61.23 | 85.81 | 53.49 | 68.32 | 55.38 | 55.281 |
| *OFF* | 0.50 | 0.20 | 0.10 | 1e-2 | 1e-3 | 1e-5 |

*GA, SA, RS, MSHC*, and *BB*, respectively. *MSHC* is the abbreviation of multistart stochastic hill climbing. *BB* is the abbreviation of branch and bound. All these optimizers have certain capability of global optimization. *BB* is an exclusion-based deterministic optimizer. *RS* is a complete stochastic optimizer without exploitation. *PSO, GA*, and *SA* are exploitative stochastic optimizers. *MSHC* can be regarded as a combination of *RS* and hill climbing which is an exploitative deterministic local optimizer. The experiments here aim at validating correlative conclusions rather than comparing the performances of these optimizers. For convenience without loss of generality, low-dimensional functions are chosen to test these optimizers. One hundred tests are taken for each optimizer and each test function. For ease of comparison, any optimizer will be terminated if it enters the level set determined by a predefined threshold or its evaluation number exceeds a predefined maximal value $(10^5)$. There are two criteria for comparison. The first one is the success ratio of locating *OF* or finding an objective value better than the predefined threshold. The second one is the average number of function evaluations in successful tests.

Both *PSO* and *GA* hold a 20-sized population. The inertia factor in *PSO* is set to 0.729, and its two acceleration factors are both set to 1.494 [46]. *GA* uses 30-bit binary codes, proportional selection, single-point crossover with probability 0.6, and uniform mutation with probability 0.01. The descending of temperature in *SA* is proportional, and its proportion factor is set to 0.99. Temperature descending occurs only when the number of state acceptance at the temperature exceeds a predefined number. The hill-climbing step in *MSHC* is set to $\alpha|\vec{d}\cdot\vec{b}|$. $\vec{d}$ is a unit vector representing the current steepest ascending direction. $\vec{b}$ is a vector which has the maximal vector module in search space. $\alpha$ is the step factor initially set to 0.05. $|\vec{d}\cdot\vec{b}|$ denotes the absolute value of the inner product of these two vectors. Starting points are chosen randomly. *BB* uses midpoint sampling for both updating the currently known best value of the objective function and identifying unwanted regions. The principles for branching and deleting are similar to those used in [51]. The bounding of *BB* is based on the *Lipschitz* constant. Note that there is no *Lipschitz* constant for pulse functions and their variations, and a nominal "*Lipschitz*" constant such as ten is used in *BB* for the optimization of these functions.

*Experiment I—Optimization of 1-D Pulse Functions With Different OFFs:* The pulse functions used here have the same form as follows:

$$f_{\mathrm{p}}(x) = \mathrm{Sign}(x-a) - \mathrm{Sign}(x-b), \qquad \text{with } 0 < x < 100$$

where $0 < a < b < 100$.

The six optimizers previously mentioned are used to maximize the pulse functions in Table I. The performances of these optimizers are shown in Table II.

*Remark:* As indicated in Table II, when optimization hardness reaches certain degree, exploration-dominated algorithms like *RS* will become the best. When pulses are wide enough, the performance of exploitative algorithms such as *PSO* and *GA* are similar to that of *RS*. In fact, it is often the initial random samplings or explorative operators in these algorithms that contribute to the locating of optima. For pulse functions, exploitation is almost useless, which explains why the success ratio of the exploitation-dominated optimizer *PSO* trails off with the increase of optimization hardness. For the same reason, *GA, SA*, and *MSHC*, which depend on exploitation more or less, pay out an extra cost for locating optimal field in contrast with *RS*. When optimization hardness is not too great, the performance of *BB* with midpoint sampling is excellent. In fact, the branching and deleting (excluding) subprocess of *BB* is an exclusion-based contraction process, which avoids repeatedly sampling any region in search space. Usually, *BB* chooses the region, whose upper bound is the biggest, as its first processing object. Therefore, the performance of *BB* depends on the accuracy and validity of its estimation of the upper bound for each region. In an ideal case, the estimated upper bound is expected to be the minimal upper bound. Generally, the bounding subprocess of *BB* requires some global information such as the *Lipschitz* constant about objective functions. However, as optimization hardness increases, it becomes difficult to provide such information, and the estimation loses its validity or becomes impractical gradually. In this case, the estimation-based processing order may result in slow or premature convergence. In addition, as indicated by Table II, the cost of each optimizer increases with optimization hardness, which accords with the *Optimal Contraction Theorem.*

*Experiment II—Validation of Equivalent Optimization Hardness About Various Functions:* In order to validate the equivalent difficulty of different functions, the following two functions are used as test problems. The two functions are expected to have the equivalent optimization hardness as the function $f_{\mathrm{p}}(x)$ since all of them have the same *OFF*.

$$f_{\mathrm{np}}(x) = A\cdot[2 - f_{\mathrm{p}}(x)]\cdot r + f_{\mathrm{p}}(x), \quad \text{with } 0 < x < 100$$

where $0 < A < 1$ and $r$ is a random number generated in the interval (0, 1)

$$f_{\mathrm{dp}}(x) = [2 - f_{\mathrm{p}}(x)]\cdot\frac{(x-50)^2}{5000} + f_{\mathrm{p}}(x), \qquad \text{with } 0 < x < 100.$$

$f_{\mathrm{np}}(x)$ is a noisy time-variant function. $f_{\mathrm{dp}}(x)$ is a deceptive function which has two strong nonoptimal attractors at boundaries. The curves for $f_{\mathrm{p},2}(x)$, $f_{\mathrm{np},2}(x)$, and $f_{\mathrm{dp},2}(x)$ which have the same *OFF* are shown in Fig. 5. The experimental results for the maximization of the two functions are shown in Table III.

*Remark:* As indicated by Table III, for different optimizers, the optimization cost of the same function may be different. In fact, the minimal cost of optimizing a problem depends not only on the problem itself but also on optimizers. This can be deciphered by the definition of *OFF*. The optimization hardness mentioned here is measured by the minimal cost. As shown in Tables II and III, with *BB* only considered as a reference, the minimal cost of optimizing each function with the above optimizers is the cost of *RS*. Therefore, the cost of *RS* is chosen as the foundation for problem difficulty analysis. It is easy to see that the minimal optimization costs of the functions $f_{\mathrm{np},i}$, $f_{\mathrm{dp},i}$, and $f_{\mathrm{p},i}$, for the same $i$, are almost equal. Therefore, the three functions have equivalent optimization hardness. This experiment shows that some feature factor such as *OFF* determines the difficulty of optimizing a function. For an optimizer, distinct functions may have the same rank of optimization hardness. In addition, all optimizers except *RS* are sensitive to the deceptive function $f_{\mathrm{dp},5}$. Normal exploitation is badly misguided in the optimization of $f_{\mathrm{dp},5}$. Noise may be beneficial in the optimization of $f_{\mathrm{np}}$ as it may bring useful gradientlike information. In contrast with other optimizers, *SA* is obviously sensitive to the magnitude of noise.

TABLE II
SUCCESS RATIO AND AVERAGE COST AT SUCCESS OF DIFFERENT OPTIMIZERS IN OPTIMIZING PULSE FUNCTIONS

| Optimizers | $f_{p,1}$ | $f_{p,2}$ | $f_{p,3}$ | $f_{p,4}$ | $f_{p,5}$ | $f_{p,6}$ |
|---|---|---|---|---|---|---|
| *PSO* | (1.00, 2.09) | (1.00, 4.76) | (1.00, 8.93) | (0.64, 52.1) | (0.19, 151) | (0.00, -) |
| *GA* | (1.00, 1.80) | (0.99, 7.92) | (1.00, 8.58) | (0.77, 1521) | (0.59, 9358) | (0.11, 39412) |
| *SA* | (1.00, 8.12) | (1.00, 18.8) | (1.00, 45.5) | (1.00, 293) | (0.54, 999) | (0.01, 1240) |
| *RS* | (1.00, 2.13) | (1.00, 4.30) | (1.00, 10.4) | (1.00, 106) | (1.00, 969) | (0.66, 40428) |
| *MSHC* | (1.00, 3.82) | (1.00, 11.5) | (1.00, 24.5) | (1.00, 321) | (1.00, 3296) | (0.31, 49160) |
| *BB* | (1.00, 1.00) | (1.00, 2.00) | (1.00, 8.00) | (1.00, 64.0) | (1.00, 586) | (1.00, 75139) |

*The first part of each data pair in Table II is the success ratio of locating *OF* within 100,000 function evaluations. The second part is the average number of function evaluations for successful locatings. Since *BB* makes use of global information, it is only regarded for fairness as a reference for other optimizers.

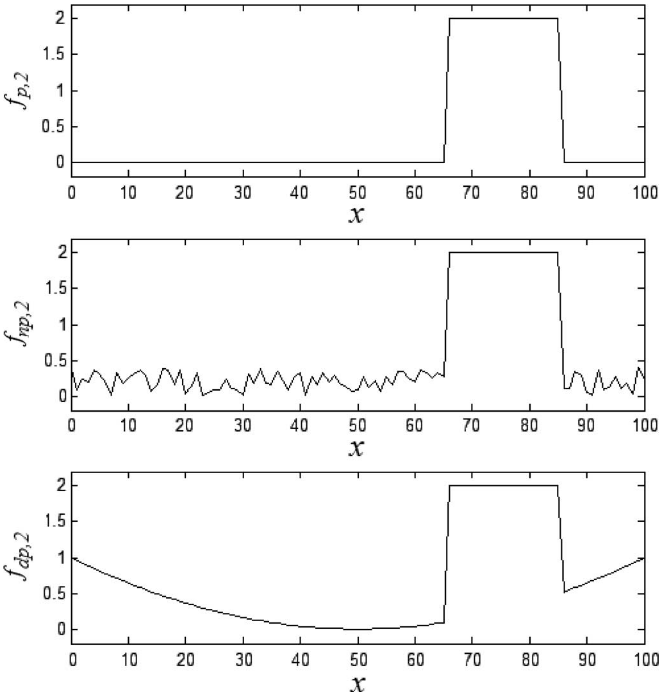

Fig. 5. Curves for $f_{\mathrm{p},2}(x)$, $f_{\mathrm{np},2}(x)$, and $f_{\mathrm{dp},2}(x)$.

*Experiment III—2-D Benchmark Function Tests:*

$$f_{\mathrm{b},1}(x_1,x_2) = x_1^2 + x_2^2, \qquad \text{with } -100 < x_1, x_2 < 100$$

$$f_{\mathrm{b},2}(x_1,x_2) = \frac{1}{4000}\sum_{i=1}^{2}(x_i-100)^2 - \prod_{i=1}^{2}\cos\left(\frac{x_i-100}{\sqrt{i}}\right)+1, \qquad \text{with } -600 < x_1, x_2 < 600$$

$$f_{\mathrm{b},3}(x_1,x_2) = 100\left(x_2-x_1^2\right)^2 + (x_1-1)^2, \qquad \text{with } -2.048 < x_1, x_2 < 2.048$$

$$f_{\mathrm{b},4}(x_1,x_2) = 0.5 + \frac{\sin^2\sqrt{x_1^2+x_2^2}-0.5}{\left[1+0.001\left(x_1^2+x_2^2\right)\right]^2}, \qquad \text{with } -100 < x_1, x_2 < 100$$

$$f_{\mathrm{b},5}(x_1,x_2) = -\sum_{i=1}^{2} x_i \cdot \sin\left(\sqrt{|x_i|}\right), \qquad \text{with } -500 < x_1, x_2 < 500.$$

The above benchmark functions are 2-D *Sphere, Griewank, Rosenbrock, Schaffer*, and *Schwefel* functions, respectively [46], [49]. $f_{\mathrm{b},1}$, $f_{\mathrm{b},2}$, $f_{\mathrm{b},3}$, and $f_{\mathrm{b},4}$ are single-funnel functions which have obvious global features [49]. Particularly, $f_{\mathrm{b},1}$ is unimodal. In contrast, $f_{\mathrm{b},5}$ is a multifunnel function which is considered to be relatively difficult for *PSO* to optimize [49]. The *OF*-based *OFF*s corresponding to the five functions are 1, $4.61e-8$, 1, $7.65e-7$, and $3.22e-3$, respectively.

Six optimizers are expected to minimize these test functions. For optimizers used for maximization, if they are required to minimize an objective function, the simplest approach is to maximize the reverse of the function. Different from pulse functions, when an optimizer locates the optimal field of these functions successfully, it does not mean that the optimizer has discovered a global optimal point. Some gradient information will become an indispensable guide for exploitative optimizers to find a global optimal point. In this case, the disadvantage of random sampling in optimization accuracy will emerge since it has no exploitation. Thus, for fairness, both the performance of approaching global optima and that of locating optimal field about six optimizers are considered. The experimental results for these two performance comparisons are shown in Tables IV and V, respectively.

*Remark:* There is much useful gradient or gradientlike information in the optimization of these functions, which benefits exploitative optimizers. This is one reason for that the exploitation-dominated optimizer *PSO* outperforms the other optimizers except *BB* in this experiment. However, *PSO* is easily trapped at the local optimal points of the *Schwefel* function $(f_{\mathrm{b},5})$. This is because the optimization of the *Schwefel* function depends on a preferable ability of locating *OF*. Furthermore, such ability depends on exploration which is just the weakness of *PSO*. Nevertheless, due to excellent exploitation performance, *PSO* can achieve accurate optimization rapidly once it locates the *OF* of the *Schwefel* function. In fact, for *PSO*, the *OFF*s of the above five functions are not based on *OF*, because *PSO* is not a purely explorative optimizer like *RS*. Thus, the *OF*-based *OFF* cannot reflect the optimization hardness of these functions for *PSO*. It is reasonable to use *piOR* as *OFF* in this case. The *piOR*s of the *Griewank* function and the *Schaffer* function for *PSO* are much larger than that of the *Schwefel* function. This explains why it is relatively difficult for *PSO* to optimize multifunnel functions [49].

In contrast with *PSO, RS* keeps its superiority in locating the *OF* of the *Schwefel* function $(f_{\mathrm{b},5})$, but it lacks the ability of accurate optimization. For *RS*, the success ratio of locating the *OF* of the *Griewank* function is the lowest, which accords with the indication of *OFF* (*SOR*).

For binary-coded *GA*, the *OF*-based *OFF* that is *SOR* is also invalid since the three operators in *GA* do not operate in the *Euclidean* space [54]. A simple example for the irrationality of *SOR* for *GA* is shown in Fig. 6.

In order to characterize the optimization hardness of problems for *GA*, the neighborhood of the points in search space must be redefined.

TABLE III
VALIDATION OF EQUIVALENT OPTIMIZATION HARDNESS

| Optimizers | $f_{np,2}$ ($A$=0.2) | $f_{dp,2}$ | $f_{np,4}$ ($A$=0.2) | $f_{dp,4}$ | $f_{np,5}$($A$=0.2) | $f_{np,5}$ ($A$=0.9) | $f_{dp,5}$ |
|---|---|---|---|---|---|---|---|
| *PSO* | (1.00,4.94) | (1.00,4.45) | (0.87,785) | (0.83,236) | (0.88,3554) | (0.88,3565) | (0.38,1827) |
| *GA* | (0.99,5.00) | (1.00,5.77) | (0.69,1341) | (0.64,2968) | (0.46,11074) | (0.52,9450) | (0.14,27422) |
| *SA* | (1.00,22.7) | (1.00,20.1) | (1.00,326) | (1.00,329) | (1.00,988) | (0.70,1485) | (0.46,1229) |
| *RS* | (1.00,5.10) | (1.00,5.07) | (1.00,107) | (1.00,106) | (1.00,978) | (1.00,974) | (1.00,979) |
| *MSHC* | (1.00,12.2) | (1.00,16.4) | (1.00,145) | (1.00,235) | (1.00,1342) | (1.00,1513) | (1.00,4130) |
| *BB* | (1.00,2.00) | (1.00,2.00) | (1.00, 91.7) | (1.00,96) | (1.00,762) | (1.00,776) | (1.00, 970) |

* The functions $f_{np,i}$ and $f_{dp,i}$ correspond to the function $f_{p,i}$ in Table I. For the same $i$, the functions $f_{np,i}$, $f_{dp,i}$ and $f_{p,i}$ are expected to have equivalent optimization hardness.

TABLE IV
SUCCESS RATIO AND AVERAGE COST AT SUCCESS OF DIFFERENT OPTIMIZERS IN OPTIMIZING FIVE BENCHMARK FUNCTIONS

| Optimizers | $f_{b,1}$ ($f_T$=1e-3) | $f_{b,2}$ ($f_T$=1e-4) | $f_{b,3}$($f_T$=1e-3) | $f_{b,4}$($f_T$=1e-4) | $f_{b,5}$($f_T$=-837.9) |
|---|---|---|---|---|---|
| *PSO* | (1.00, 735) | (0.76, 8806) | (1.00, 933) | (0.65, 9558) | (0.41, 982) |
| *GA* | (1.00, 3527) | (0.12, 46487) | (0.18, 20065) | (0.02, 34540) | (0.04, 11705) |
| *SA* | (0.00, -) | (0.00, -) | (0.02, 1516) | (0.13, 17159) | (0.68, 4560) |
| *RS* | (0.04, 59786) | (0.00, -) | (0.81, 37879) | (0.00, -) | (0.16, 41835) |
| *MSHC* | (1.00, 1711) | (0.00, -) | (0.56, 42918) | (0.64, 40267) | (1.00, 6597) |
| *BB* | (1.00, 317) | (1.00, 1185) | (1.00, 33) | (0.00, -) | (1.00, 396) |

*All optimizers will be terminated if their currently discovered optimal values are less than a predefined threshold denoted by $f_T$ in Table IV.

TABLE V
SUCCESS RATIO AND AVERAGE COST AT SUCCESS OF DIFFERENT OPTIMIZERS IN LOCATING THE OPTIMAL FIELDS OF THE ABOVE FUNCTIONS

| Optimizers | $f_{b,1}$ ($f_T$=1e-1) | $f_{b,2}$($f_T$=7.4e-3) | $f_{b,3}$($f_T$=1e-1) | $f_{b,4}$($f_T$=9.7e-3) | $f_{b,5}$($f_T$=-720) |
|---|---|---|---|---|---|
| *PSO* | (1.00, 413) | (1.00, 2893) | (1.00, 150) | (0.73, 8257) | (0.42, 168) |
| *GA* | (1.00, 471) | (0.31, 42432) | (0.46, 7071) | (0.47, 45254) | (0.46, 271) |
| *SA* | (0.42, 3420) | (0.00, -) | (1.00, 1076) | (0.30, 8152) | (1.00, 577) |
| *RS* | (0.56, 37456) | (0.00, -) | (1.00, 564) | (0.08, 30242) | (1.00, 309) |
| *MSHC* | (1.00, 215) | (0.00, -) | (1.00, 1029) | (1.00, 5821) | (1.00, 4086) |
| *BB* | (1.00, 33) | (1.00, 749) | (1.00, 4) | (0.00,-) | (1.00, 24) |

*Due to minimization, the optimal field in this experiment should be expressed as $\{\mathbf{x} \mid \forall \mathbf{x} \in \mathcal{B} : f(\mathbf{x}) < f(\mathbf{x}')\}$ where $\mathcal{B}$ is the whole search space, and $\mathbf{x}'$ is still a local optimal point only inferior to global optima. The threshold $f_T$ for the functions $f_{b,2}$, $f_{b,4}$ and $f_{b,5}$ in Table V is an approximate value of $f(\mathbf{x}')$ . So, the values of $f_T$ in Table V are greater than the counterparts in Table IV. In contrast, the whole search space for the function $f_{b,1}$ or $f_{b,3}$ optimal field. Accordingly, there is no need to locate their optimal field. Instead, two larger thresholds are selected as a comparison for the optimization of $f_{b,1}$ and $f_{b,3}$.

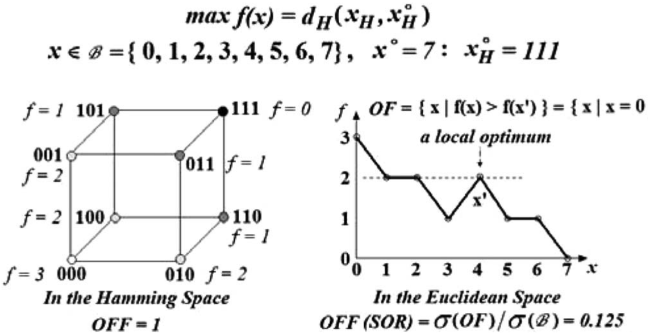

Fig. 6. Example for the irrationality of *SOR* for binary-coded *GA*. $x_H$ is the *Hamming* code for $x$ that is the binary representation of $x$. $d_H(x_H, y_H)$ denotes the *Hamming* distance between $x_H$ and $y_H$, and it is equal to the number of bits on which $x_H$ and $y_H$ have different values. It is easy to see that $x = 0$ is the global optimum of this problem. For *GA*, this is a unimodal monotonic problem which is schema-GA-easy [54]. Therefore, the accurate *OFF* of this problem should be 1. However, the *OF*-based *OFF* (*SOR*) is 0.125. The difference is mainly caused by the different neighborhood structures in the two different spaces—the *Hamming* space and the *Euclidean* space.

Schema theory [38] provides a reference for the optimization hardness analysis about *GA*. However, it is very hard to analyze all schemata about a complicated function defined in the *Euclidean* space and even in the *Hamming* space. Compared with *PSO*, the binary-coded *GA* is unsuitable for the accurate optimization of the functions in this experiment. Similarly, *SA* also depends on a slow stochastic exploitation which is similar to the proportional selection in *GA*. Therefore, it is not unusual that *SA* cannot overwhelm *RS* in this experiment. In fact, *GA, SA*, and *MSHC* on one hand are more exploitative than *RS*. On the other hand, they are more explorative than *PSO*, which can be validated by the optimization of the *Griewank* function.

*MSHC* has both the advantage of *RS* and that of *Hill climbing*, so it performs better than *RS* with respect to accurate optimization. However, it has the disadvantage of multipath redundancy, which makes the processes of hill climbing with different paths approach the same local optimal point. For *MSHC*, the starting point of each hill climbing is provided by random sampling. Therefore, its gradient-based exploitation depends on its exploration. Besides, once a hill-climbing search is started, it will not stop unless it cannot find a better solution. Therefore, the exploitation is preferential in *MSHC*. It seems that *OR* is a suitable choice for the definition of *OFF* for *MSHC*. However, in fact, the neighborhood for *MSHC* is different from that for *RS*, although *MSHC* like *RS* is also built in the *Euclidean* space. This is because the hill climbing in *MSHC* takes a fixed climbing step. A strict definition of *OFF* about *MSHC* must take this into account.

In contrast with the other global optimizers, *BB* performs very well in almost all cases, which reflects the significance of global information [60]. Particularly, it is hard for *BB* to optimize the *Schaffer* function accurately. It can be observed that *BB* approaches the suboptimal local optima of the *Schaffer* function rapidly. At this time, further subdivision of promising regions will ignore the region which contains

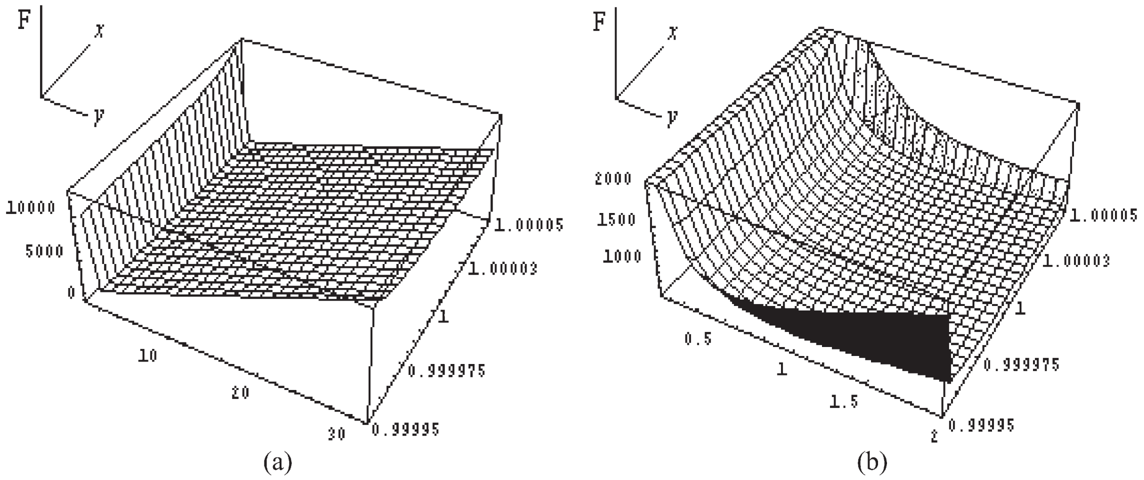

Fig. 7. Landscape of the function $F(x, y)$. (a) It is easy to see the nature of $F(x, y)$ in $y$-axis direction from the whole landscape of $F(x, y)$. (b) Local landscape of $F(x, y)$ can show a rough character of $F(x, y)$ in $x$-axis direction.

the *OF* and thus the global optimum of this function. This is because the midpoint sampling in the region gets an inferior point and thus leads to a poor lower bound for the region. Although *BB* can find this region in theory, it will take a long time due to the processing order of *BB*. For the optimization of the *Sphere* function and the *Rosenbrock* function, a gradient-based local optimizer such as *Newton*'s method is a better choice as its cost is much less. To sum up, the optimal optimizer for a particular problem must utilize the particularity of the problem as much as possible. The tradeoff between exploration and exploitation for specific optimizers depends on the optimization hardness of problems.

It should be noted that *OFF* only indicates the hardness of locating the "easy" attractive region, e.g., *OF* and *OSC*, for an optimizer. The cost of further optimization within such regions, that is the cost of exploitation, is not included in the optimal contraction theorem. Usually, if such cost dominates the total cost of the whole optimization process, the problem is not very hard. For some special landscapes and optimizers, such cost may become great due to the issue of long convergence path. In this case, other optimizers may become appropriate choices for the optimization of the special landscapes. For different optimizers, the optimization hardness of the same problem may vary. The intrinsic optimization hardness of a problem corresponds to the minimal average cost for all possible optimizers [61].

## V. Conclusion and Prospect

The optimization hardness of problems dominates *T:Er&Ei* and the minimal average cost of search and optimization. If a problem is very hard for an optimizer, the *T:Er&Ei* for this optimizer will lean toward exploration; otherwise, exploitation dominates the *T:Er&Ei*. For different optimizers used to solve the same problem, *T:Er&Ei* may be different because of different neighborhoods and landscapes. According to the *Optimal Contraction Theorem*, it is important to construct a certain optimizer based on the particularity of optimization problems, which reflects the importance of utilizing instructive information to reduce the optimization hardness of problems. No use of such particularity will cause an inevitable extra cost. In practice, many optimization problems have more or less particularity such as convexity and single funnelness which can be utilized. Even for complicated problems, local or global information such as *Lipschitz* constant can be beneficial. The optimal optimizer for a problem is expected to make the best of the useful information about the problem. In a sense, prior knowledge can transform a hard problem into a relatively easy one. In addition, reasonable characterizations of optimization hardness are significant to select or design a desirable optimizer for a specific problem.

Although the *Optimal Contraction Theorem* indicates the relationship between *T:Er&Ei* and the problems to be optimized, it is hard to regulate the *T:Er&Ei* according to an accurate *optimization feature factor* (*OFF*). This is mainly due to the large extra computational cost of directly obtaining an accurate or at least reliable *OFF*. However, it is promising to utilize a dynamic *OFF* which is computed periodically according to part of the accumulated information on objective functions. The *OFF* can be used to regulate the *T:Er&Ei* dynamically. It is significant to do a comparative research on different optimization algorithms based on different *T:Er&Ei* strategies in future.

## Appendix A
## Analysis of the Cost Function $T(M_1, n)$

For convenience of analysis, we transform the two variables of $T(M_1, n)$ and get a more regular function as follows:

$$\begin{aligned} T(M_1, n) &= F(x, y) \\ &= \ln(1 - x \cdot P^{1/2})\}/\ln(1 - y \cdot \eta^{1/2}) \\ &\quad + \ln(1 - P^{1/2}/x)\}/\ln(1 - \eta^{1/2}/y) \\ \mathcal{D}_F &= \left\{(x, y) | P^{1/2} \le x \le P^{-1/2}, \eta^{1/2} \le y \le \eta^{-1/2}\right\} \end{aligned} \tag{A1}$$

where $x = [1 - (1 - M_1\eta)^n]/P^{1/2}$ and $y = M_1\eta^{1/2}$.

It is easy to verify that (1, 1) is an equilibrium point of $F(x, y)$. The point (1, 1) will be locally minimal if the following inequality can be guaranteed:

$$F_{xx}^{(2)}(1,1) \cdot F_{yy}^{(2)}(1,1) - \left[F_{xy}^{(2)}(1,1)\right]^2 = -\frac{4\left\{P\eta + P^{1/2}\left[2\eta + \eta^{1/2}\ln(1-\eta^{1/2})\right] \cdot \ln(1-P^{1/2})\right\}}{(1-P^{1/2})^2(1-\eta^{1/2})^2\ln^4(1-\eta^{1/2})} > 0$$

where $F_{xx}^{(2)}$, $F_{yy}^{(2)}$, and $F_{xy}^{(2)}$ are two-order partial derivatives of $F(x, y)$.

Since what we concern is the case in which $P$ tends to one, the above inequality is equivalent to $1 - \eta^{1/2} - \exp(-2\eta^{1/2}) > 0$. It is easy to prove that the function $g(\eta) = 1 - \eta^{1/2} - \exp(-2\eta^{1/2})$ has only one zero in the interval (0, 1). Therefore, the unique solution to $g(\eta) = 0$, denoted by $\eta^*$, determines the range of $\eta$ restricted by the above inequality. Numerically, $\eta^* \approx 0.892643$.

When $P = 0.9999$ and $\eta = 0.001$, the landscape of $F(x, y)$ is shown in Fig. 7(a) and (b).

As $P$ tends to one, the measure in $x$-axis direction will gradually reduce to zero, and the variable $x$ tends to one. The 2-D function $F(x, y)$ will be degraded into a 1-D function which has the same shape and global optima with the following function:

$$\begin{aligned} G(y) &= -1/\ln(1 - y\eta^{1/2}) - 1/\ln(1 - \eta^{1/2}/y) \\ \mathcal{D}_G &= \{y \,|\, \eta^{1/2} y \eta^{-1/2}\}. \end{aligned} \tag{A2}$$

Then, the minimization problem $\min G(y), y \in \mathcal{D}_G$ is solved. Since $G(y)$ is continuous and smooth in its domain, we can get its optimum by evaluating all local optimal points and boundary points. It can be proved that there are at most three equilibrium points for $G(y)$. One of the three points is $y = 1$ which is a local minimal point and lies between the other two points. Therefore, the potential global minimum of $G(y)$ can only lie at $y = 1$ or its boundaries ($y = \eta^{1/2}$ or $y = \eta^{-1/2}$). Note that $G(\eta^{1/2}) = G(\eta^{-1/2})$.

If $G(1) \geq G(\eta^{1/2})$, the range of $\eta$ can be given by $(3 - \sqrt{5})/2 \leq \eta \leq 1$ and the minimum of $G(y)$ is $G(\eta^{1/2})$. In other words, $y = \eta^{1/2}$ or $\eta^{-1/2}$ is the solution to the minimization of $G(y)$. Correspondingly, for the minimization of $T(M_1, n)$ with $P \to 1$, the optimal contracting ratio is $M_1^* = 1$ or $1/\eta$, which means that the aforementioned three-stage contraction process is degraded into a two-stage contraction process. In this case, one contraction is enough to achieve the locating of *OF*. In this sense, $M_1^* = 1/\eta$ is the unique optimal contracting ratio because the search enters *OF* after the first stage, and the cost of locating *OF* which corresponds to $M_1^* = 1/\eta$ is reduced to $T|_{M=1/\eta} = m < m + n = T|_{M=1}$.

If $G(1) < G(\eta^{1/2})$, the range of $\eta$ can be given by $0 < \eta < (3 - \sqrt{5})/2$, and the minimum of $G(y)$ is $G(1)$. $M_1^* = 1/\eta^{1/2}$ is the unique optimal contracting ratio. In this case, $M_2^* = (M_1^* \eta)^{-1} = 1/\eta^{1/2} = M_1^*$. Besides, it is easy to verify that $m = n = \ln(1 - P^{1/2})/\ln(1 - \eta^{1/2})$ which means that the cost of the first stage $(m)$ is equal to that of the second stage $(n)$. Therefore, the minimal cost of locating *OF* is $2\ln(1 - P^{1/2})/\ln(1 - \eta^{1/2})$. As $P$ tends to one, the minimal cost tends to infinity, which means the locating of *OF* cannot be determinately achieved. This is because the contraction results in the incompleteness of the information about objective functions and makes the probability of omitting *OF* nonzero. Therefore, strict global optimization is impossible for stochastic contraction process.

## Appendix B
## Analysis of the Generalized Cost Function $T(k)$

Relax $T(k)$ to $T(z) = z\ln(1 - P^{1/z})/\ln(1 - \eta^{1/z})$ by permitting the variable $z$ to be any real number satisfying $z \geq 1$. In order to obtain the optimal number of contraction stages, we analyze the equilibrium point of $T(z)$ by solving the zero of its one-order derivative $T^{(1)}(z)$. The equation $T^{(1)}(z) = 0$ is equivalent to the following equation:

$$\frac{\eta^{1/z}\ln(\eta^{1/z})}{(1 - \eta^{1/z})\ln(1 - \eta^{1/z})} - \frac{P^{1/z}\ln(P^{1/z})}{(1 - P^{1/z})\ln(1 - P^{1/z})} = 1. \tag{B1}$$

As $P$ tends to one, the second term in the above equation $[P^{1/z}\ln(P^{1/z})]/[(1 - P^{1/z})\ln(1 - P^{1/z})]$ tends to 0. Thus, (B1) equivalently becomes $\eta^{1/z}\ln(\eta^{1/z}) = (1 - \eta^{1/z})\ln(1 - \eta^{1/z})$. Since the curve $h(t) = t\ln(t)$ intersects $h(1 - t)$ only at $t = 0.5$ in the interval (0, 1), the equilibrium point of $T(z)$ is determined by $\eta^{1/z} = 1 - \eta^{1/z}$, giving $z = -\log_2(\eta)$. Therefore, the optimal number of contraction stages denoted by $k^*$ is $\lceil -\log_2(\eta) \rceil$ or $\lceil -\log_2(\eta) \rceil + 1$ where $\lceil x \rceil$ denotes the biggest integer which is not greater than $x$. Note that $z \geq 1$ requires $\eta \leq 0.5$. Given different values of $\eta$ with $P$ fixed at 0.9, the curve of $T(k)$ is shown in Fig. 8.

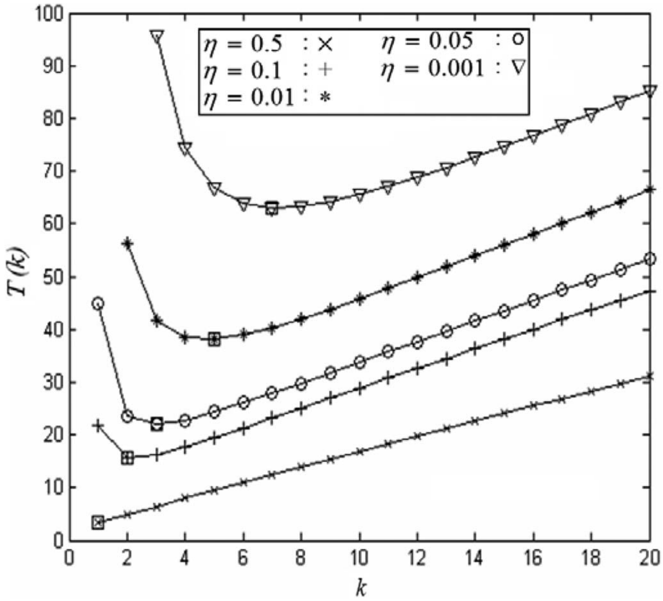

Fig. 8. Optimal stage number increases when $\eta$ is reduced. Note that each point surrounded by a square is optimal among all points lying on the same curve.

## Acknowledgment

The authors would like to thank the Editor-in-Chief, Associate Editor, and the anonymous referees for their constructive suggestions and comments on the earlier version of this paper. The authors would also like to thank Dr. J. Hosseni of Marquette University, Milwaukee, WI, for his contribution to the improvement of this paper's quality.

## References

[1] A. G. Pipe, T. C. Fogarty, and A. Winfield, "Balancing exploration with exploitation-solving Mazes with real numbered search spaces," in *Proc. 1st IEEE Int. Conf. Comput. Intell.*, Orlando, FL, 1994, pp. 485–489.

[2] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[3] G. Yen, F. Yang, T. Hickey, and M. Goldstein, "Coordination of exploration and exploitation in a dynamic environment," in *Proc. Int. Joint Conf. Neural Netw.*, Washington DC, 2001, pp. 1014–1018.

[4] S. Ishii, W. Yoshida, and J. Yoshimoto, "Control of exploration and exploitation meta-parameter in reinforcement learning," *Neural Netw.*, vol. 15, no. 4–6, pp. 665–687, Jun./Jul. 2002.

[5] P. Auer, "Using confidence bounds for exploitation-exploration tradeoffs," *J. Mach. Learn. Res.*, vol. 3, pp. 397–422, 2002.

[6] A. Pchelkin, "Efficient exploration in reinforcement learning based on utile suffix memory," *INFORMATICA*, vol. 8, no. 2, pp. 267–283, 2003.

[7] A. McMahon, D. Scott, and D. W. Browne, "An autonomous explore/exploit strategy," in *Proc. Workshop Genet. Evol. Comput.*, Washington DC, 2005, pp. 103–108.

[8] M. Usher, J. D. Cohen, D. Servan-Schreiber, J. Rajkowski, and G. Aston-Jones, "The role of locus coeruleus in the regulation of cognitive performance," *Science*, vol. 283, no. 5401, pp. 549–554, Jan. 1999.

[9] S. M. McClure, M. S. Gilzenrat, and J. D. Cohen, "An exploration-exploitation model based on norepinepherine and dopamine activity," in *Advances in Neural Information Processing Systems*, vol. 18. Cambridge, MA: MIT Press, 2005.

[10] G. I. Hawe and J. K. Sykulski, "Balancing exploration and exploitation using Kriging surrogate models in electromagnetic design optimization," in *Proc. 12th Bienial IEEE Conf. Electromagn. Field Comput.*, Miami, FL, 2006, p. 229.

[11] Y. Zhang, W. Xu, and J. Callan, "Exploration and exploitation in adaptive filtering based on Bayesian active learning," in *Proc. 20th Int. Conf. Mach. Learn.*, Washington DC, 2003, pp. 21–24.

[12] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. Michigan Press, 1975.

[13] B. B. Li, L. Wang, and B. Liu, "An effective PSO-based hybrid algorithm for multiobjective permutation flow shop scheduling," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 4, pp. 818–831, Jul. 2008.

[14] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM Comput. Surv.*, vol. 35, no. 3, pp. 268–308, Sep. 2003.

[15] W. G. Macready and D. H. Wolpert, "Bandit problems and the exploration/exploitation tradeoff," *IEEE Trans. Evol. Comput.*, vol. 2, no. 1, pp. 2–22, Apr. 1998.

[16] P. Auer, N. C. Bianchi, Y. Freund, and R. E. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM J. Comput.*, vol. 32, no. 1, pp. 48–77, 2002.

[17] S. Mannor and J. N. Tsitsiklis, "The sample complexity of exploration in the multi-armed bandit problem," *J. Mach. Learn. Res.*, vol. 5, pp. 623–648, 2004.

[18] P. Auer, N. C. Bianchi, and P. Fischer, "Finite time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, no. 2/3, pp. 235–256, May/Jun. 2002.

[19] Y. Leung, Y. Gao, and Z. B. Xu, "Degree of population diversity—A perspective on premature convergence in genetic algorithms and its Markov chain analysis," *IEEE Trans. Neural Netw.*, vol. 8, no. 5, pp. 1165–1176, Sep. 1997.

[20] H. B. Amor and A. Rettinger, "Intelligent exploration for genetic algorithms using self-organizing maps in evolutionary computation," in *Proc. Conf. Genet. Evol. Comput.*, Washington DC, 2005, pp. 1531–1538.

[21] M. Bhattacharya, "Exploiting landscape information to avoid premature convergence in evolutionary search," in *Proc. IEEE Congr. Evol. Comput.*, Vancouver, BC, Canada, 2006, pp. 560–564.

[22] R. K. Ursem, "Diversity-guided evolutionary algorithms," in *Proc. Parallel Problem Solving From Nature VII*, 2002, pp. 462–471.

[23] M. Hutter and S. Legg, "Fitness uniform optimization," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 568–589, Oct. 2006.

[24] E. Alba and B. Dorronsoro, "The exploration/exploitation tradeoff in dynamic cellular genetic algorithms," *IEEE Trans. Evol. Comput.*, vol. 9, no. 2, pp. 126–142, Apr. 2005.

[25] B. Yuan and M. Gallagher, "On the importance of diversity maintenance in estimation of distribution algorithms," in *Proc. Conf. Genet. Evol. Comput.*, Washington DC, 2005, pp. 719–726.

[26] I. C. Trelea, "The particle swarm optimization algorithm: Convergence analysis and parameter selection," *Inf. Process. Lett.*, vol. 85, no. 6, pp. 317–325, Mar. 2003.

[27] P. A. N. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 174–188, Apr. 2003.

[28] A. Toffolo and E. Benini, "Genetic diversity as an objective in multi-objective evolutionary algorithms," *Evol. Comput.*, vol. 11, no. 2, pp. 151–167, 2003.

[29] M. J. Sasena, P. Papalambros, and P. Goovaerts, "Exploration of metamodeling sampling criteria for constrained global optimization," *Eng. Optim.*, vol. 34, no. 3, pp. 263–278, Jan. 2002.

[30] A. Sobester, S. J. Leary, and A. J. Keane, "On the design of optimization strategies based on global response surface approximation models," *J. Glob. Optim.*, vol. 33, no. 1, pp. 31–59, Sep. 2005.

[31] A. Prudius and S. Andradóttir, "Simulation optimization using balanced explorative and exploitative search," in *Proc. Winter Simul. Conf.*, Washington DC, 2004, pp. 545–549.

[32] A. Auger, M. Schoenauer, and O. Teytaud, "Local and global order 3/2 convergence of a surrogate evolutionary algorithm," in *Proc. Conf. Genet. Evol. Comput.*, Washington DC, 2005, pp. 857–864.

[33] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983.

[34] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Perth, Australia, 1995, pp. 1942–1948.

[35] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Glob. Optim.*, vol. 11, no. 4, pp. 341–359, Dec. 1997.

[36] E. Bonabeau, M. Dorigo, and G. Theraulaz, "Inspiration for optimization from social insect behavior," *Nature*, vol. 406, no. 6791, pp. 39–42, Jul. 2000.

[37] M. Pelikan, D. E. Goldberg, and E. Cantu-Paz, "Linkage problem, distribution estimation and Bayesian networks," *Evol. Comput.*, vol. 8, no. 3, pp. 311–340, Sep. 2000.

[38] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[39] T. Kuo and S. Y. Hwang, "A genetic algorithm with disruptive selection," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 26, no. 2, pp. 299–307, Apr. 1996.

[40] V. Barhen, V. Protopopescu, and D. Reister, "TRUST: A deterministic algorithm for global optimization," *Science*, vol. 276, no. 5315, pp. 1094–1097, May 1997.

[41] J. Riget and J. S. Vesterstrøm, "Diversity-guided particle swarm optimizer—The ARPSO," Dept. Comput. Sci., Univ. Aarhus, Aarhus, Denmark, EVALifa Tech. Rep. No. 2002-02, 2002.

[42] T. Blackwell and J. Branke, "Multiswarms, exclusion, and anti-convergence in dynamic environments," *IEEE Trans. Evol. Comput.*, vol. 10, no. 4, pp. 459–472, Aug. 2006.

[43] M. Lundy and A. Mees, "Convergence of an annealing algorithm," *Math. Program.*, vol. 34, no. 1, pp. 111–124, Jan. 1986.

[44] G. Rudolph, "Convergence analysis of canonical genetic algorithm," *IEEE Trans. Neural Netw.*, vol. 5, no. 1, pp. 96–101, Jan. 1994.

[45] T. Stützle and M. Dorigo, "A short convergence proof for a class of ant colony optimization algorithms," *IEEE Trans. Evol. Comput.*, vol. 6, no. 4, pp. 358–365, Aug. 2002.

[46] M. Clerc and J. Kennedy, "The particle swarm—Explosion, stability and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, vol. 6, no. 1, pp. 58–73, Feb. 2002.

[47] D. H. Wolpert and W. G. Macready, "No free lunch theorem for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.

[48] H. Maaranen, K. Miettinen, and A. Penttinen, "On initial populations of a genetic algorithm for continuous optimization problems," *J. Glob. Optim.*, vol. 37, no. 3, pp. 405–436, Mar. 2007.

[49] A. M. Sutton, D. Whitley, M. Lunacek, and A. Howe, "PSO and multi-funnel landscapes: How cooperation might limit exploration," in *Proc. 8th Annu. Conf. Genet. Evol. Comput.*, Seattle, WA, 2006, pp. 75–82.

[50] Z. B. Xu, J. S. Zhang, and Y. W. Leung, "A general CDC formulation for specializing the cell exclusion algorithms of finding all zeros of vector functions," *Appl. Math. Comput.*, vol. 86, no. 2, pp. 235–259, Oct. 1997.

[51] M. Sun and A. W. Johnson, "Interval branch and bound with local sampling for constrained global optimization," *J. Glob. Optim.*, vol. 33, no. 1, pp. 61–82, Sep. 2005.

[52] X. Yang and M. Sun, "Theoretical convergence analysis of a general division-deletion algorithm for solving global search problems," *J. Glob. Optim.*, vol. 37, no. 1, pp. 27–45, Jan. 2007.

[53] R. F. Marcia, J. C. Mitchell, and J. B. Rosen, "Multi-funnel optimization using Gaussian underestimation," *J. Glob. Optim.*, vol. 39, no. 1, pp. 39–48, Sep. 2007.

[54] T. Jones, "One operator, one landscape," Santa Fe Inst., Santa Fe, NM, Tech. Rep. No.95-02-025, 1995.

[55] B. Naudts and L. Kallel, "A comparison of predictive measures of problem difficulty in evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 4, no. 1, pp. 1–15, Apr. 2000.

[56] D. J. Wales and H. A. Scheraga, "Global optimization of clusters, crystals and biomolecules," *Science*, vol. 285, no. 5432, pp. 1368–1372, Aug. 1999.

[57] J. J. Gray, S. Moughon, C. Wang, O. Schueler-Furman, B. Kuhlman, C. A. Rohl, and D. Baker, "Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations," *J. Mol. Biol.*, vol. 331, no. 1, pp. 281–299, Aug. 2003.

[58] F. van den Bergh and A. P. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 225–239, Jun. 2004.

[59] D. Whitley, K. Mathias, S. Rana, and J. Dzubera, "Building better test functions," in *Proc. 6th Int. Conf. Genet. Algorithms*, Pittsburgh, PA, 1995, pp. 239–247.

[60] C. P. Stephens and W. Baritompa, "Global optimization requires global information," *J. Optim. Theory Appl.*, vol. 96, no. 3, pp. 575–588, Mar. 1998.

[61] W. G. Macready and D. H. Wolpert, "What makes an optimization problem hard?" *Complexity*, vol. 1, no. 5, pp. 40–46, 1996.