Applied Soft Computing 12 (2012) 2828–2839

Contents lists available at SciVerse ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

# An intuitive distance-based explanation of opposition-based sampling

Shahryar Rahnamayan[a,*], G. Gary Wang[b,1], Mario Ventresca[c,d]

[a] Faculty of Engineering and Applied Science, University of Ontario Institute of Technology (UOIT), 2000 Simcoe Street North, Oshawa, Ontario, Canada L1H 7K4
[b] School of Engineering Science, Simon Fraser University, 250-13450 102 Avenue Surrey, BC, Canada V3T 0A3
[c] Center for Pathogen Evolution, Department of Zoology, University of Cambridge, Downing St., Cambridge CB2 3EJ, UK
[d] Department of Mechanical and Industrial Engineering, 5 King's College Road, Toronto, ON, Canada M5S 3G8

ARTICLE INFO

*Article history:*
Received 1 October 2009
Received in revised form 2 November 2011
Accepted 18 March 2012
Available online 30 April 2012

*Keywords:*
Opposition-based learning
Opposite point
Sampling
Opposition-based optimization
Opposition-based soft computing

ABSTRACT

The impact of the opposition concept can be observed in many areas around us. This concept has sometimes been called by different names, such as, opposite particles in physics, complement of an event in probability, absolute or relative complement in set theory, and theses and antitheses in dialectic. Recently, opposition-based learning (OBL) was proposed and has been utilized in different soft computing areas. The main idea behind OBL is the simultaneous consideration of a candidate and its corresponding opposite candidate in order to achieve a better approximation for the current solution. OBL has been employed to introduce opposition-based optimization, opposition-based reinforcement learning, and opposition-based neural networks, as some examples among others. This work proposes an Euclidean distance-to-optimal solution proof that shows intuitively why considering the opposite of a candidate solution is more beneficial than another random solution. The proposed intuitive view is generalized to *N*-dimensional search spaces for black-box problems.

## 1. Introduction

Opposition-based learning (OBL) was introduced by Tizhoosh in 2005 [18]. The main idea behind OBL is the simultaneous consideration of an estimate and its corresponding opposite estimate (i.e., guess and opposite guess) in order to achieve a better approximation for the current candidate solution. Later, by considering opposite individuals during opposition-based population initialization and generation jumping, OBL was employed to introduce opposition-based differential evolution (ODE) [3,4,7,8,14,17]. Comparative studies have confirmed that ODE performs better than DE in terms of convergence speed. A self-adaptive ODE was introduced in [11]. A comprehensive survey of in differential evolution are provided in [5,6]. By replacing quasi-opposite numbers with opposite numbers in ODE, quasi-oppositional DE (QODE) [10,12] was proposed. Both ODE and QODE used a constant generation jumping rate, variable jumping rates were investigated for ODE in [13]. A decreasing jumping rate presented better performance than a fixed one; which means opposition-based generation jumping is more beneficial during exploration than during exploitation. A self-adaptive ODE with population size reduction was employed to tackle large scale problems [2] [37]. As some applications for ODE among others, ODE with a small population size (Micro-ODE) was utilized for image thresholding [16]; results confirmed that the Micro-ODE converges to optimal solution faster than Micro-DE. An adaptive ODE applied to tuning of a Chess program [39]. Similarly, by considering opposite states and opposite actions, opposition-based reinforcement learning (ORL) was proposed [19,20,24–26] and showed that ORL outperforms its parent algorithm (RL). ORL was applied to prostate ultrasound image segmentation [33] and management of water resources [34]. Furthermore, opposition-based neural networks were introduced by considering opposite transfer functions and opposite weights [27,28,30]. Opposition-based simulated annealing (OSA) was proposed based on opposite neighbors [29]. OSA showed improvement in accuracy and convergence rate over traditional SA. By introducing opposite particles, Particle Swarm Algorithms were accelerated and opposition-based PSO was introduced [38,40,45–48]. Opposition-based ant colony (OACO) algorithms were proposed by introducing opposite(anti)-pheromone [35,36]. Population-based incremental learning (PBIL) has also been greatly enhanced by considering opposite samples [31]. Performance of the harmony search [32] and biogeography-based optimization [22,23] were improved by OBL. All of these algorithms have tried to enhance searching or learning in different fields of soft computing and they were experimentally verified

[*] Corresponding author. Tel.: +1 905 721 8668x3843.
*E-mail addresses:* shahryar.rahnamayan@uoit.ca (S. Rahnamayan), gary_wang@sfu.ca (G.G. Wang), mario.ventresca@utoronto.ca (M. Ventresca).
[1] Tel.: +1 778 782 8495.

[2] It uses opposition concept implicitly by changing the sign of F and so searching in the opposite direction.

1568-4946/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2012.03.034

by benchmark functions; a majority of these algorithms and also other opposition-based works have been explained in [9].

Among all proposed opposition-based algorithms, the ODE is the most well-known and promising one. Neri et al. divided modern DE-based algorithms into the following two categories [6]: (1) DE with an integrated extra component, and (2) DE with a modified structure. The first group includes the algorithms with DE framework and an extra component, such as, local searchers and/or additional operators. The second group contains types of DE-based algorithms which modify the main structure of the canonical DE. According to this classification, the same authors considered the ODE in the second category [6] with other recently proposed enhanced DE variants, such as, Self Adaptive Control Parameters [56–59], Global–Local Search DE [60,61], and Self Adaptive Coordination of Multiple Mutation Rules [62–64]. DE suffers from its limited amount of the exploratory moves (due to its limited mutation and crossover combinations) which can be improved by embedding alternative moves [49]. Furthermore, the limited amount of the moves can cause an undesirable search process stagnation; the situation which diversity of the population is still high but it does not converge to a solution [50]. The successful extra moves can be achieved by two ways: (a) increasing the exploitative pressure and/or (b) utilizing some randomization scheme [6]. In this light, the ODE uses the first approach by proposing a new operator (i.e., opposition-based generation jumping); it checks unexplored areas of the decision space by utilizing the mentioned alternative moves [6]. These additional moves improve DE's exploration performance and also reduce the chance of stagnation by injecting fitter opposite individuals during the generation jumping. The risk of stagnation is higher when the dimension of the problem increases [6,65–69,58]. Probably that is why ODE even performs better on large-scale problems [17]. In literature, there are other similar works which have introduced extra moves to improve DE's efficiency [6]; introducing a trigonometric mutation [51], an adaptive local search [52], memetic DE [53,54], and scale factor local search [55] are some examples among others in this direction. Weber et al. introduced a scale factor inheritance mechanism in distributed DE [49], in their proposed mechanism, they have utilized different scale factors for each sub-population to increase the exploration power of the original DE. In their algorithm, the negative value for the scale factor means the search in the opposite directions, which is similar to the ODE mechanism that considers the opposite individuals during the dynamic generation jumping.

All OBL-based approaches reported promising results but there was a fundamental question which should be answered properly: Intuitively, why is the opposite of the current point more advantageous than a second pure random point (opposition vs. randomness)? This question has just been addressed in [2] where the mathematical proof and experimental verification confirmed each other and proved numerically how much better an opposite point is when compared to a uniformly generated random point. However, the proposed proof suffers from two shortcomings: (1) only one dimensional search spaces were considered, and (2) the proof is not able to provide an intuitive explanation for the observed results. In [9], the authors mentioned that *"Due to our imperfect understanding of interplay between opposite entities, this work will most likely be a preliminary investigation. Hence, more comprehensive elaborations with a solid mathematical understanding of opposition remains a subject of future research."* In order to address this issue, we propose the current mathematical proof which (1) is much simpler than the previous proof proposed by the first author [1,2], and (2) is generalized to higher dimensions, and (3) explains intuitively the philosophy behind the opposition concept from the perspective of distance to the optimal solution.

This paper is organized as follows: the concept of opposition is described in Section 2. Preliminary definitions and assumptions for the proposed mathematical proof are given in Section 3. The proposed mathematical proof is explained in Section 4. The results are compared and analyzed in Section 5 and finally the paper is concluded in Section 7.

## 2. The concept of opposition

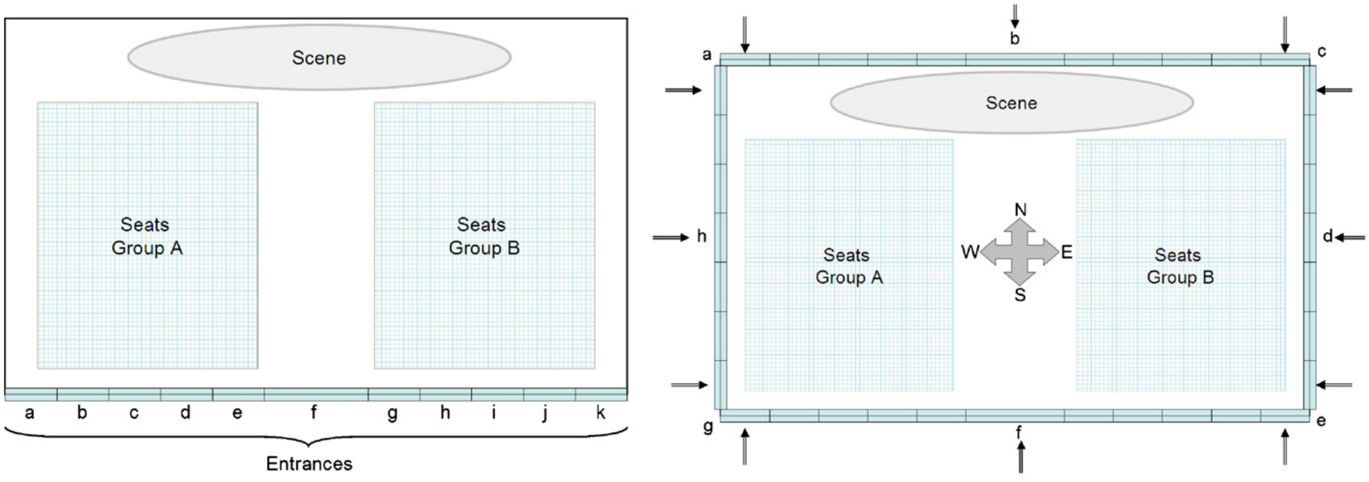
The footprints of the opposition concept can be observed in many areas around us. This concept has sometimes been labeled with different names. Opposite particles in physics, antonyms in languages, complement of an event in probability, antithetic variables in simulation, opposite proverbs in culture, absolute or relative complement in set theory, subject and object in philosophy of science, opposition parties in politics, theses and antitheses in dialectic, and dualism in religions and philosophies are just some examples among others to mention.

The Yin-Yang symbol in ancient Chinese philosophy is probably the oldest opposition concept which was expressed by human kind. Black and white represent yin (receptive, feminine, dark, passive force) and yang (creative, masculine, bright, active force), respectively. This symbol reflects the twisted duality of all things in nature, namely, receptive versus creative, feminine versus masculine, dark versus bright, and finally passive versus active forces. Even Greek classical elements to explain patterns in nature mention the opposition concept, namely, fire (hot and dry) versus water (cold and wet), earth (cold and dry) versus air (hot and wet). Cold, hot, wet, dry present the pair-wised opposite characteristics of these four elements.

It seems that without using the opposition concept, the explanation of different entities around us is hard and maybe even impossible. In order to explain an entity or a situation we sometimes explain its opposite instead. In fact, opposition often manifests itself in a balance between completely different entities. For instance, the east, west, south, and north cannot be defined alone. The same is valid for cold and hot and many other examples. Extreme opposites constitute our upper and lower boundaries. Imagination of the infinity is vague, but when we consider the limited, it then becomes more imaginable because its opposite is definable.

Sometimes we apply the opposition concept in our regular life unconsciously. Let us look at a simple example (see Fig. 1). Suppose police officers want to arrest a suspect in a theater hall arranged to have two seating groups ($A$ and $B$) and a number of entrance doors ($a-k$) on one side of the hall (Fig. 1(a)). The seat position of the target person is unknown (just like the position of the optimal solution in a black-box optimization problem) and only two officers are available. If the first officer selects the door $a$, which door will likely be selected by the second officer? What happens if the first officer selects the door $b$? In order to increase their chances of successfully arresting the suspect the officers will arrange themselves such that they cover the most exist possible (assuming an officer can cover a reasonable distance about their position).

Let us consider the same example but this time the theater hall has doors on all four sides, as shown in Fig. 1(b). Now, let us increase the number of officers (like individuals in a population-based optimization method) and repeat the same questions. When officer one selects the door $h$, the second officer will selects $d$. Why are the other doors not selected instead? The third officer will now select door $b$, and the fourth will likely choose door $f$. It seems that even when we increase the number of officers the opposition pattern for covering the doors is still followed. These are the officers' intuitive decisions in different situations, and perhaps they are unaware of the concept of the opposition but they apply it in order to cover the search space more efficiently.



(a) Theater Hall with entrance doors on one side.

(b) Theater Hall with entrance doors on all four sides.

**Fig. 1.** Theater hall example.

As mentioned above, the opposition concept has been utilized in different fields of soft computing in order to enhance evolutionary algorithms (e.g., DE, PBIL and PSO), neural networks (NN), reinforcement learning (RL), ant colony algorithms (ACA), simulated annealing (SA), etc. According to our point of view, the opposition concept introduces a new scheme which can be utilized in a wide range of scientific areas. To mention an example, opposition-based sampling can be employed by methods which tackle expensive design optimization problems, such as the mode pursing sampling method (MPS) [41,42].

## 3. Preliminaries and intuition

In this section we outline the assumptions and definitions required for the subsequent mathematical proof in the following section.

### 3.1. Preliminaries

**Definition 1.** Let $a < b \in \mathbb{R}$ and $x \in [a, b]$ be selected according to a uniform distribution. The opposite of $x$ will be denoted as $\breve{x}$ and calculated by $\breve{x} = a + b - x$.

This definition can be extended to higher dimensions by applying the same formula to each dimension [18,21].

**Definition 2.** Let $P := \mathbf{X}^N \in \mathbb{R}$ be an arbitrary point in $N$-dimensional space such that $x_i \in [a_i, b_i]$ for $a_i < b_i \in \mathbb{R}$ and $i = 1, \ldots, N$. The opposite of $P$, is denoted as $\breve{P}$ and is calculated as:

$$\breve{P} = \breve{X}^N = \breve{x}_{i=1,\ldots,N} = a_i + b_i - x_i \tag{1}$$

The top plot in Fig. 2 illustrates $x$ and its opposite $\breve{x}$ in interval $[a, b]$. As is seen, $x$ and $\breve{x}$ are located at equal distances from the interval's center ($|(a+b)/2 - x| = |\breve{x} - (a+b)/2|$) and the interval's boundaries ($|x - a| = |b - \breve{x}|$) as well. The remaining two plots consider dimensions of size 2 and 3.

**Definition 3.** The Euclidean distance between two points $X = (x_1, x_2, \ldots, x_N)$ and $Y = (y_1, y_2, \ldots, y_N)$ for $X, Y \in \mathbb{R}^N$ is defined by

$$d(X, Y) = \|X, Y\| = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2}. \tag{2}$$

**Definition 4.** Let $R, X \in S \subset \mathbb{R}^N$ be randomly selected over some bounded subspace $S$ in $\mathbb{R}^N$. Also, let $p_X$, $p_{\breve{X}}$ and $p_R$ be the probability of guess $X, \breve{X}$ and $R$ being closer, with respect to Euclidean distance, to some unknown solution $s \in S$. We assume that $s$ is chosen according to a uniform distribution. The probabilities are denoted as follows:

$$p_X = p(d(X, s) < d(\breve{X}, s) \wedge d(X, s) < d(R, s)), \tag{3}$$

$$p_{\breve{X}} = p(d(\breve{X}, s) < d(X, s) \wedge d(\breve{X}, s) < d(R, s)), \tag{4}$$

$$p_R = p(d(R, s) < d(\breve{X}, s) \wedge d(R, s) < d(X, s)), \tag{5}$$

$$p_X + p_{\breve{X}} + p_R = 1 \tag{6}$$

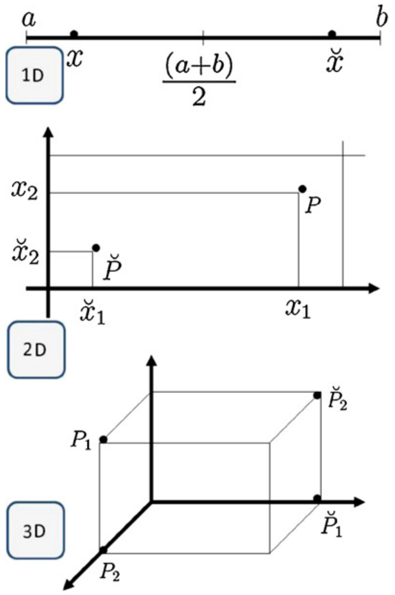


**Fig. 2.** Illustration of a point and its corresponding opposite in one, two, and three dimensional spaces.

**Definition 5.** The probability density function for the uniform distribution is defined as follows:

$$f(x) = \begin{cases} \dfrac{1}{b-a} & a \le x \le b \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

The mean, median, and variance are $(a+b)/2$, $(a+b)/2$, and $(b-a)^2/12$, respectively.

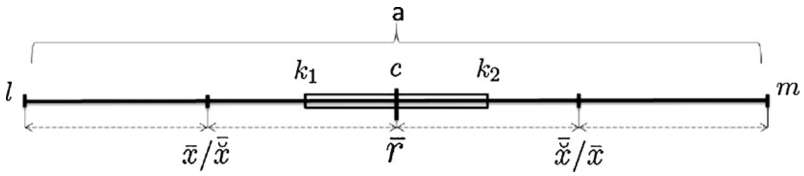


**Fig. 3.** Illustration of the solution's region for 1-D search space in which a random point is closer to the unknown solution than $x$ and $\breve{x}$. $k_1$ and $k_2$ are the centers of intervals $[\bar{x}/\bar{\breve{x}}, \bar{r}]$ and $[\bar{r}, \bar{\breve{x}}/\bar{x}]$.

In the subsequent proof, we make the following assumptions:

1 The problem being solved is black-box and no a priori information is available concerning the location of the optimal solution.
2 Due to (1), a uniform distribution is used to generate random guesses.
3 There exists a unique global optima $s$ within bounded subspace $S \subset \mathbb{R}^N$.
4 The Euclidean distance to $s$ is a sufficient approximation for the quality of a guess within the context of any optimization algorithm.

### 3.2. Distance-based intuition

Before deriving a more comprehensive proof, we provide evidence of the use of opposition (as defined above) from the perspective of paired samples versus according to the method of opposite numbers. Specifically, we consider the expected distance between two values $X, Y \in [a, b]$ chosen at random against $X, \breve{X}$. For joint distribution $f_{XY}(x, y) = 1/L^2$, where $L = b - a$ is the length of the interval being considered, this expected distance can be calculated according to

$$\mathbb{E}[|X-Y|] = \frac{1}{L^2}\int_0^L\int_0^L |x-y|dxdy. \tag{8}$$

The inner integral can be evaluated to

$$\begin{aligned}\int_0^L |x-y|dy &= \int_0^x (x-y)dy + \int_x^L (y-x)dy \\ &= \left[xy - \frac{y^2}{2}\right]_0^x + \left[\frac{y^2}{2} - xy\right]_x^L \\ &= \left(x^2 - \frac{x^2}{2}\right) + \frac{L^2}{2} - xL - \left(\frac{x^2}{2} - x^2\right) \\ &= \frac{L^2}{2} + x^2 - xL.\end{aligned} \tag{9}$$

Substituting back into Eq. (8),

$$\begin{aligned}\mathbb{E}[|X-Y|] &= \frac{1}{L^2}\int_0^L \left(\frac{L^2}{2} + x^2 - xL\right)dx \\ &= \frac{1}{L^2}\left[\frac{xL^2}{2} + \frac{x^3}{3} - \frac{Lx^2}{2}\right]_0^L \\ &= \frac{1}{L^2}\left(\frac{L^3}{2} + \frac{L^3}{3} - \frac{L^3}{2}\right) \\ &= \frac{L}{3}.\end{aligned} \tag{10}$$

Therefore, two uniformly sampled points can be expected to be separated by about 1/3 the size of the interval. This coverage of the search space is somewhat limited considering that the expected value of a random guess will fall at the mid-way point of the interval. If the optimal solution lies within this narrow range, the strategy could be promising. However, in general, we are unsure and assume the optimal is uniformly distributed over the entire interval. One can follow a similar approach as above to show $\mathbb{E}[|X-\breve{X}|] = L/2$, which should be expected from Definition 1. In this situation the interval $L$ is covered much better, and thus we can expect that the distance between a uniformly distributed optimal solution is likely to be closer to $(X, \breve{X})$. In the following section we examine the conjecture that the distance[3] to the optimal solution is lower when using the opposite strategy.

## 4. The main theorem

In this section, $p_X$, $p_{\breve{X}}$, and $p_R$ are mathematically derived. We first show evidence for the claim of opposite superiority using one, two and three dimensional situations. These are then used in the proof which generalizes to $\mathbb{R}^N$.

### 4.1. Investigation

#### 4.1.1. One-dimensional space

Consider the situation presented in Fig. 3, which is bounded by $[l, m]$ and has a center at point $c$. Without loss of generality[4] let $x \in [l, c]$ and $\breve{x} \in [c, m]$. Then, the average values of the guess $\bar{x}$ and opposite guess $\bar{\breve{x}}$ are located midway[5] in the sub-intervals $[l, c]$ and $[c, m]$, respectively. Using the same logic, the average of both random guesses will be located at $\bar{r} = c$. Considering these mean values we can deduce that a uniformly distributed solution $s \in [l, m]$ will, on average, be closer to the independently randomly generated points within the region $[k_1, k_2]$.

Then, the probability of a random point being closer to the optimal solution can be simply calculated:

$$p_{\bar{r}} = \frac{|k_2 - k_1|}{|m-l|} = \frac{2}{8} = 0.25. \tag{11}$$

Moreover, we know $p_x = p_{\breve{x}}$ and that

$$p_x + p_{\breve{x}} + p_{\bar{r}} = 1, \tag{12}$$

and therefore $p_x = 0.375$ and $p_{\breve{x}} = 0.375$. Recall, $p_{\bar{r}}$, $p_x$, and $p_{\breve{x}}$ are in Definition 4.

#### 4.1.2. Two-dimensional space

The two-dimensional situation is very similar to the 1D case. However, here we must now consider four possible mean location of the $(X, \breve{X})$ pairing. These points are represented by gray dots in Fig. 4. The diamond area outlined by a dashed line corresponds to the area where a random guess will outperform the opposite strategy, on average. This area is the 2D version to the 1D interval $[k_1, k_2]$ shown above. The dark black point represents the center of the search space.

The area surrounded by the dotted square shown in Fig. 4 is the only region (on average) in which a random point is closer to the

[3] We focus on the distance to optimal in sample space, which disregards the problem landscape.

[4] In general, $x \in [l, m]$, with $\breve{x}$ being in the lower or upper half of the search space, depending on the location of $x$. For simplicity, generating $x \in [l, c]$ is done.

[5] The mean and median value of a uniform distribution on interval $[a, b]$ is equal to $(a + b)/2$.

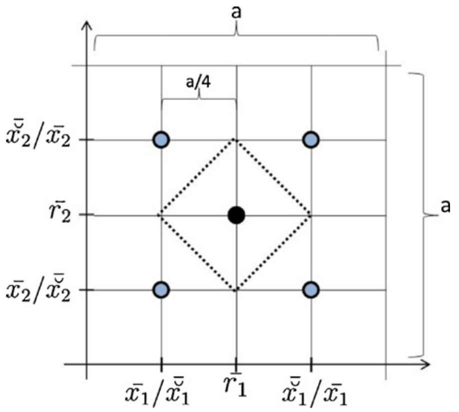

a
a/4
a
$\breve{\bar{x}}_2/\bar{x}_2$
$\bar{r}_2$
$\bar{x}_2/\breve{\bar{x}}_2$
$\bar{x}_1/\breve{\bar{x}}_1$
$\bar{r}_1$
$\breve{\bar{x}}_1/\bar{x}_1$

**Fig. 4.** An illustration of the solution region (the area surrounded by a dotted square) for a 2-D search space. A random point is closer to the unknown solution versus $(x, \breve{x})$ within this area.

solution than $x$ or $\breve{x}$. The area of this region is equal to $(a\sqrt{2}/4)^2$, since the edge length of the dotted square in Fig. 4 is $(a\sqrt{2}/4)$. The probability of the unknown solution falling into this region would therefore be equal to

$$\frac{((a/4)\times\sqrt{2})^2}{a^2}. \tag{13}$$

Further, two possible types of opposite points exist, as shown in Fig. 5.

So, we have

$$p_{\bar{r}} = 2\times\frac{((a/4)\times\sqrt{2})^2}{a^2} = 0.25, \tag{14}$$

and again, $p_x = 0.375$, and $p_{\breve{x}} = 0.375$.

### 4.1.3. Three-dimensional space

This case again is very similar to the two previous ones, see Fig. 6. This time we have eight center points (four diameters and so three different pairs of opposites, resulting in four situations, see Fig. 7). Now, we calculate the volume shown by the dotted cube in a similar manner as above. This results in

$$p_{\bar{r}} = 4\times\frac{((a/4)\times\sqrt{2})^3}{a^3} = 0.25, \tag{15}$$

where the factor 4 represents the number of situations. Thus, $p_x = 0.375$ and $p_{\breve{x}} = 0.375$.

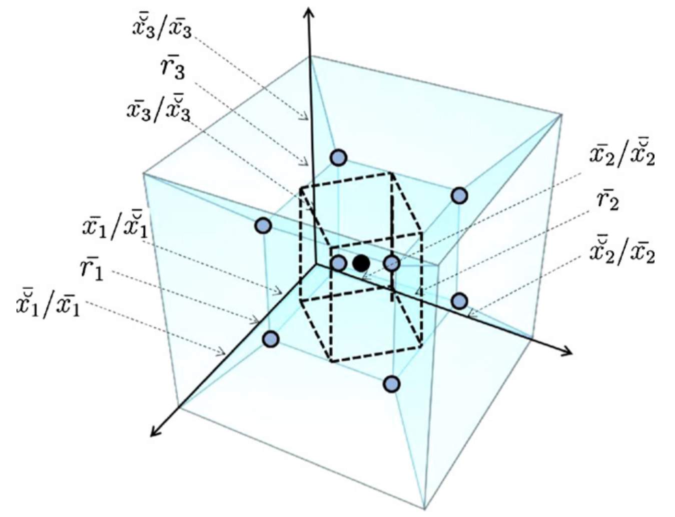

$\breve{\bar{x}}_3/\bar{x}_3$
$\bar{r}_3$
$\bar{x}_3/\breve{\bar{x}}_3$
$\bar{x}_1/\breve{\bar{x}}_1$
$\bar{r}_1$
$\breve{\bar{x}}_1/\bar{x}_1$
$\bar{x}_2/\breve{\bar{x}}_2$
$\bar{r}_2$
$\breve{\bar{x}}_2/\bar{x}_2$

**Fig. 6.** Illustration of the solution's region (the volume surrounded by a dotted cube) for a 3-D search space, in which a random point is closer to the unknown solution than $x$ and $\breve{x}$.

### 4.2. The proof

Our proof will extend the above examples to show that in general, the probability of a paired-random guessing strategy being superior to the opposite strategy is constant and equal to 0.25. First, we determine the volume of an inner hypercube, representing the region where random guesses are closer to an optimal solution, on average. Then, we integrate the number of opposite guess pairs to arrive at the solution.

**Theorem 6.** *Let $H_o^N$ be an $N > 0$ dimensional hypercube with edge lengths $L_o > 0$. Moreover, let $H_i^N$ be an inscribed hypercube having a common center to $H_o^N$ and with edge lengths*

$$L_i(N) = \begin{cases} \dfrac{L_o}{4}, & \text{if } N = 1 \\ \dfrac{L_o\sqrt{2}}{4}, & \text{if } N \geq 2 \end{cases}. \tag{16}$$

*Then, the volume of $H_i^N$ is given by $1/2^{N+1}$. When considering opposites $(X, \breve{X})$, the probability $p_{\bar{R}}$ will then be constant and equal to 0.25.*

**Proof.** The outer hypercube $H_o^N$ will have volume $V_o(N) = L_o^N$ (product of all edge lengths). Similarly, the inner hypercube $H_i^N$ will have volume

$$V_i(N) = (L_i(N))^N = \left(\frac{L_o\sqrt{2}}{4}\right)^N = \frac{L_o^N}{2^{N+1}} \tag{17}$$

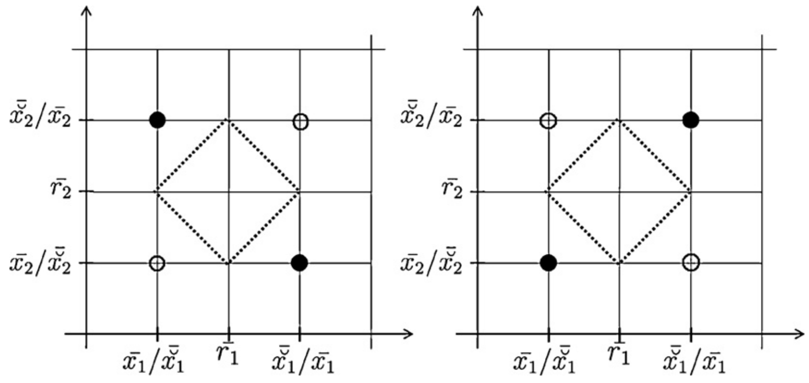

$\breve{\bar{x}}_2/\bar{x}_2$
$\bar{r}_2$
$\bar{x}_2/\breve{\bar{x}}_2$
$\bar{x}_1/\breve{\bar{x}}_1$
$\bar{r}_1$
$\breve{\bar{x}}_1/\bar{x}_1$
$\breve{\bar{x}}_2/\bar{x}_2$
$\bar{r}_2$
$\bar{x}_2/\breve{\bar{x}}_2$
$\bar{x}_1/\breve{\bar{x}}_1$
$\bar{r}_1$
$\breve{\bar{x}}_1/\bar{x}_1$

**Fig. 5.** Illustration of the two possible situations for a 2-D search space. The black points are selected points of each situation.

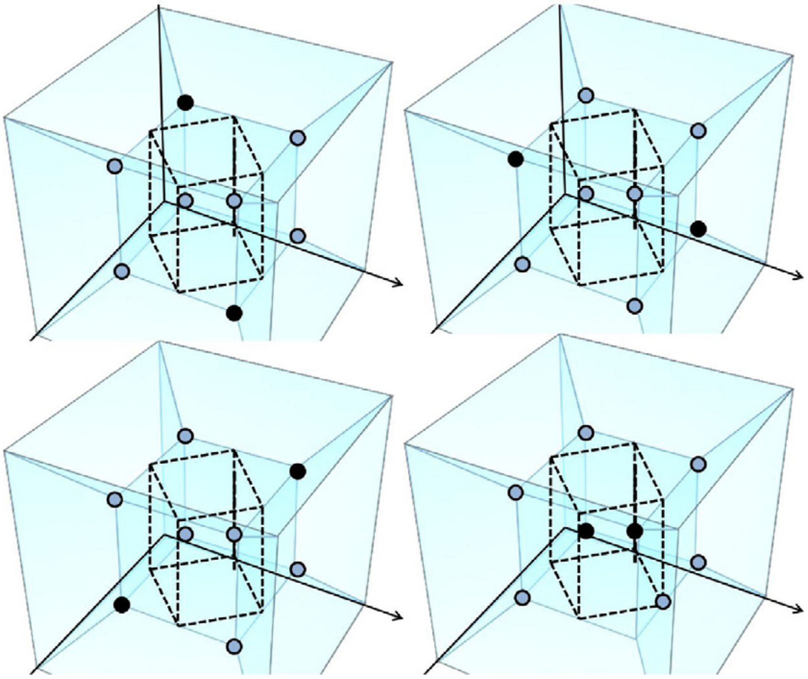


**Fig. 7.** Illustration of the four possible situations for 3-D search space. The black points are the selected points of each situation.

where the volume for the $N = 1$ case has been absorbed into $V_i(N)$. The ratio of inner to outer hypercube volume (after simplification) is

$$\frac{V_i}{V_o} = \frac{L_o^N}{2^{N+1}}\frac{1}{L_o^N} = \frac{1}{2^{N+1}}. \tag{18}$$

Every $N$ will have two opposite points, and thus one pair $(X, \breve{X})$. Therefore, the total number of opposite pairings at a given dimension $N$ will be $2^{N-1}$. Then, the probability the random guess is closer is calculated as

$$p_{\overline{R}} = \frac{2^{N-1}}{2^{N+1}} = \frac{1}{4}. \tag{19}$$

□

As a corollary, we now have in general that $p_R = 0.25$, $p_X = 0.375$ and $p_{\breve{X}} = 0.375$. This proves the opposite guessing strategy is more desirable than simply paired random guessing from the perspective of Euclidean distance for a finite-dimensional problem having solutions defined over a $N$-dimensional hypercube. Of course, the location of the optimal value in the search space will introduce differences from the theoretically expected values.

Fig. 8 presents the mean distance to an optimal solution in 1D. The black line represents the random strategy and the gray line corresponds to the opposition-based sampling approach, respectively. To generate this figure we fix the optimal solution location at $\{0.0, 0.01, 0.02, \ldots, 0.99, 1.00\}$. At each fixed location 10,000 pairs of samples are generated and the minimum distance to the optimum is recorded. The opposite strategy is clearly more desirable, unless the optimum is located near the center of the interval. A main source of variation from theoretical and experimental results will be related to the location of the optimal solution, as is further discussed below.

Results of previous mathematical proof [2], and the current proposed proof are summarized in Table 1. As seen, the difference between these results is less than 3%. The main reason for this difference is that the mentioned probabilities ($p_x$, $p_{\breve{x}}$) have their own optimum values (0.375) when the reference center points are located exactly at the center of intervals/regions, otherwise

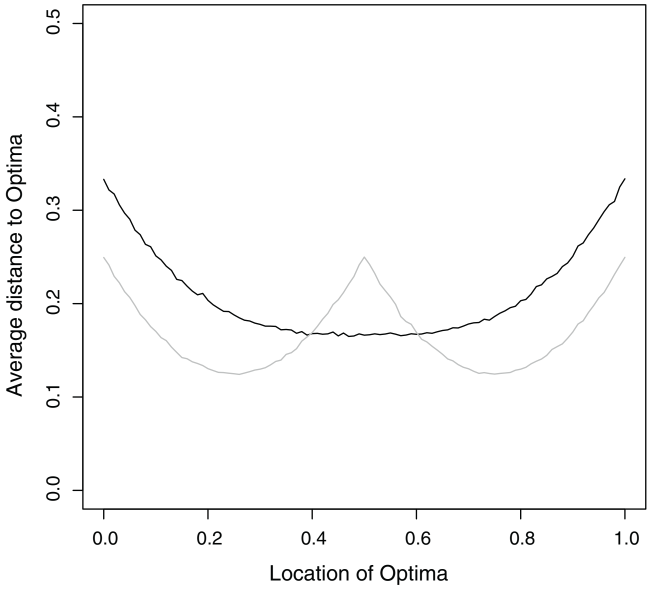


**Fig. 8.** The mean distance to optimal of the paired random guessing (black) and opposite strategy (gray). We fix the location of the optimal solution and generate 10,000 points and present the mean distance. This was repeated for optimal={0.0, 0.01, 0.02, . . ., 0.99, 1.00}.

**Table 1**
Comparison of two sets of results. As seen, the difference between the results of the proposed proof and previous results (mathematical proof presented in [2]) is less than 3%.

| | $p_x$ | $p_{\breve{x}}$ | $p_r$ |
|---|---|---|---|
| Previous mathematical proof [2] | 0.3613 | 0.3613 | 0.2773 |
| Proposed proof | 0.375 | 0.375 | 0.25 |

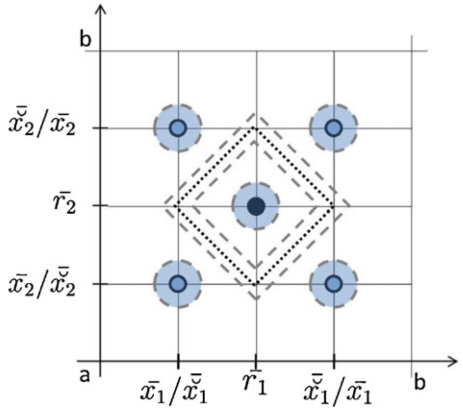

b
$\breve{\bar{x}}_2/\bar{x}_2$
$\bar{r}_2$
$\bar{x}_2/\breve{\bar{x}}_2$
a
$\bar{x}_1/\breve{\bar{x}}_1$
$\bar{r}_1$
$\breve{\bar{x}}_1/\bar{x}_1$
b

**Fig. 9.** Illustration of the reason for the observed tolerance (for 2-D). Dashed areas present the variation of the centers and the region because of the existence of non-zero variances.

any variation (standard deviation) affects their values (reduces from 0.375 to 0.3613), see Fig. 9 for the illustration. Dashed areas present the variation of the centers and the region because of the existence of non-zero variances for the average points (which is $\sigma^2 = (b-a)^2/12$ for uniform distribution in each dimension).

As seen simply in all cases, on average, the candidate solutions and their opposites are located at the center of subregions. Therefore, for each subregion, there is a representative which is in the best possible position. The best is understood in a sense that it has an overall minimum distance from all points in that subregion. The search space can then divided into subregions (two in 1-D, four in 2-D, eight in 3-D, and finally $2^N$ in N dimensional spaces) and their centers are conquered by candidates and opposite candidates. This is why an opposite point has a higher chance to be closer to solution than a pure random point. This behavior, to some extent, is similar to that of stratified random sampling methods, e.g., Latin Hypercube Sampling, which often yields better efficiency than a pure random sampling. The mentioned proof offers us the intuitive explanation of the philosophy behind the opposition.

## 5. Computational analysis

We now perform various computational experiments and analyses to confirm the above results, as well as to investigate various properties of the proposed system. First, we provide evidence supporting the proof in Section 4.2. Then we examine the best/worst and mean case scenarios from a distance-based perspective. Our last experiment considers purely random versus paired opposite guesses in the context of an ensemble of solutions.

### 5.1. *Confirmation of the main proof*

First we confirm the theoretically derived probabilities discussed in the main theorem. Evidence is provided in Fig. 10, where we see the constant probabilities of $p_R \approx 0.25$ and $p_X = p_{\breve{X}} \approx 0.375$. The values were generated by selecting an optimal value $s \in [0, 1]^N$ and recording the frequencies associated with the three possible outcomes of guessing strategy. To gain an accurate estimate 2000 pairs of uniform and opposite guesses are generated at each iteration for a single $s$. Due to the noise associated with placement of $s$, we average these results over 100 uniform values. This process is repeated for $N = 1, \ldots, 25$. Aside from having approximate values as those theoretically derived, the behavior is constant which was also a requirement of the proposed theorem.

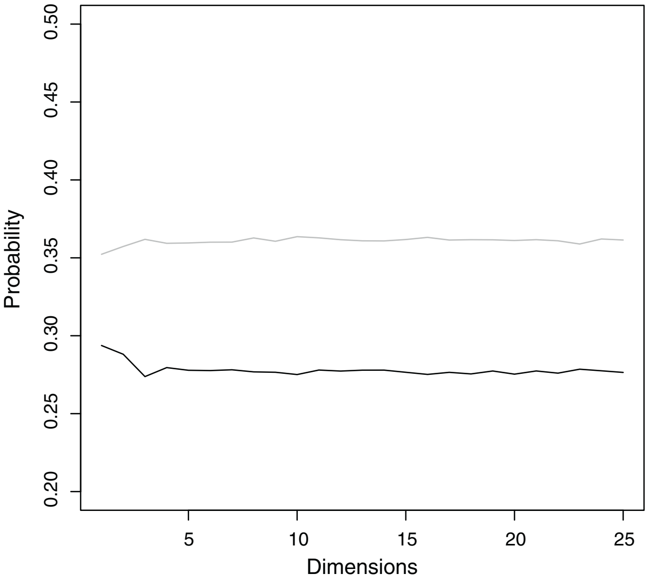


**Fig. 10.** Probability of opposite (gray) or random (black) guess-based solution being closer to the optimal $s \in [0, 1]^N$. The number of dimensions $N = 1, \ldots, 25$ is varied and the probabilities are estimated from 100 placements of $s$, each utilizing 2000 sample pairs. The opposite strategy yields a more desirable result. Moreover, the values are constant, as predicted.

### 5.2. *Distance to optima*

Fig. 8 highlights the mean distance to the optima for the 1D case. When the optimal solution is near 0.25 and 0.75 the opposite-based samples (OBS) are expected to show better performance compared to when the location is near 0.5, when the random-based samples (RBS) actually achieves a lower expected distance to the optimal solution (DTO). In Fig. 11, we further examine these three situations.

In total, 10,000 paired samples are generated for each case, and the resulting minimum distance to optima is recorded. The first row corresponds to the optimal being located at 0.25, then 0.5 and 0.75 for the second and third rows, respectively. The first column displays a frequency plot of DTO for the paired-random strategy, and the second column shows the corresponding figure for opposition-based sampling. Finally, the cumulative distribution function for the two methods is given in the third column. The dotted line represents the opposition-based method and the solid line for the random-based approach, respectively.

As can be expected, the range of possible DTO values is greater for RBS than for OBS. Moreover, the distribution of OBS is essentially uniform over the smaller range whereas the RBS do not. The associated cumulative distribution plots capture this behavior. OBS show a linear increase in probability whereas the RBS exhibit a nonlinear increase. These distributions highlight the observed experimental results in the best and worst case situations for OBS.

#### 5.2.1. *Increasing ensemble size*

Many search heuristics utilize populations, or ensembles. In this experiment, we focus on the performance of OBS and RBS as the number of sample pairs increases. Intuitively, greedy selection of only the best solution in an ensemble should rapidly yield similar results over both strategies. That is,

$$\lim_{n\to\infty} \min(X, \breve{X}) = \min(X, Y) \tag{20}$$

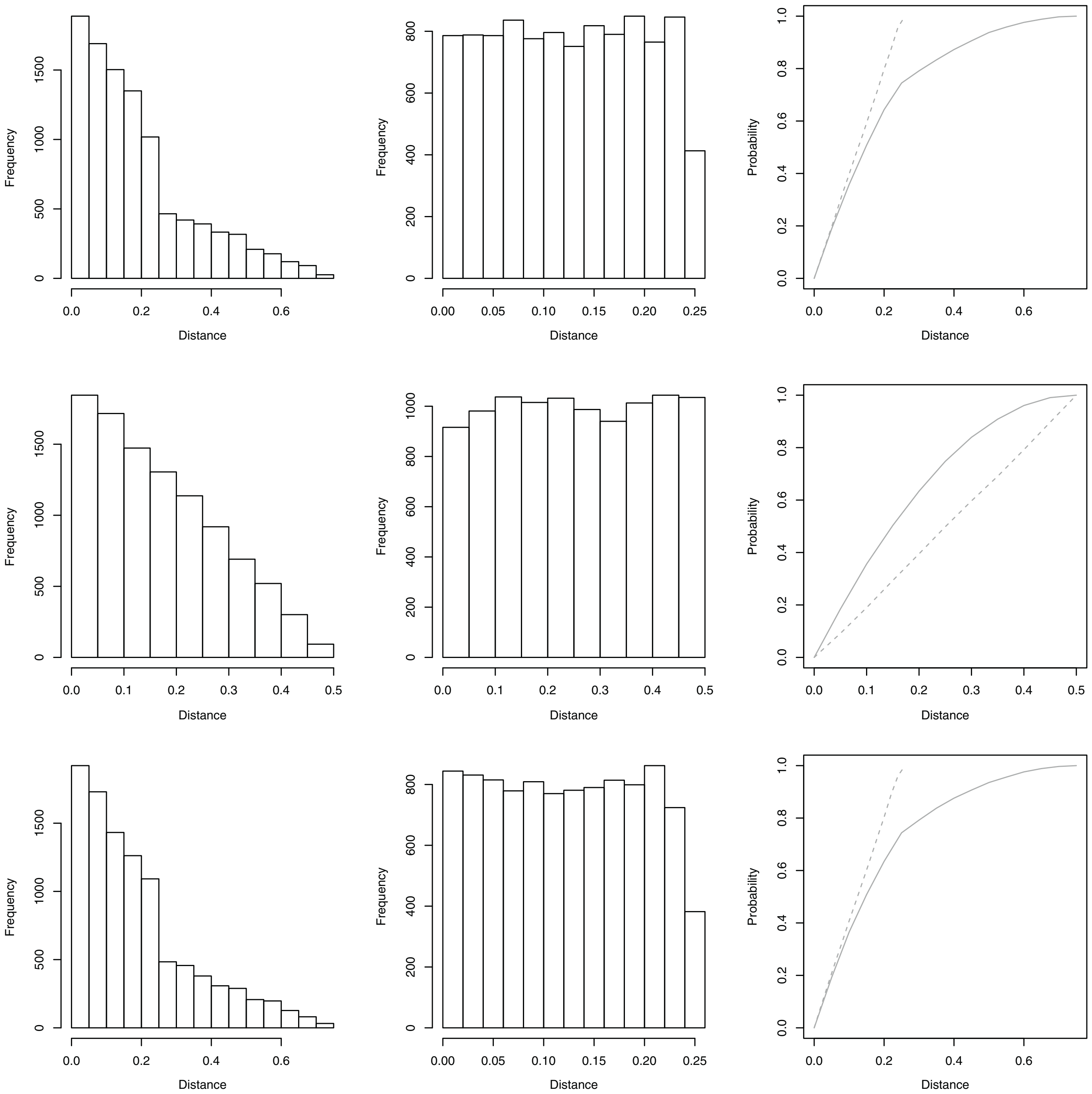


**Fig. 11.** Frequency plots for RBS (first column) and OBS (second column) are given for optimal solution location at 0.25 (first row), 0.5 (second row) and 0.75 (third row). The third column presents the cumulative distribution of these results. The results are obtained over 10,000 samples per optimal solution location.

for $n > 0$ sample pairs, where it is implied that $X, \breve{X}, Y$ are sets of $n$ samples over some $N$-dimensional problem. In practice, the number of samples required to achieve nearly identical minimum DTO is very small (as seen below in Fig. 12). However, in most ensemble-based approaches the minimum itself is not necessarily the best estimate of ensemble quality (i.e., the ensemble mean distance to optima may be of more importance).

Fig. 12 presents results that examine properties of the ensemble. In all cases sample sizes of 1, ..., 50 are considered and values are estimated over 5000 trials where the optima is randomly reassigned at each trial. The lower left plot displays the average distance[6] of samples in the ensemble to the optimal location. These values are approximately constant and equal to about 0.20 for RBS and 0.16 for OBS. The standard deviation about the mean for each strategy is shown in the upper right plot of this figure. A sharp increase for small sample sizes, however the value stabilizes for both strategies. The lower standard deviation for OBS indicates a higher degree of reliability in the outcomes.

[6] It is not the average minimum distance to the optimal.

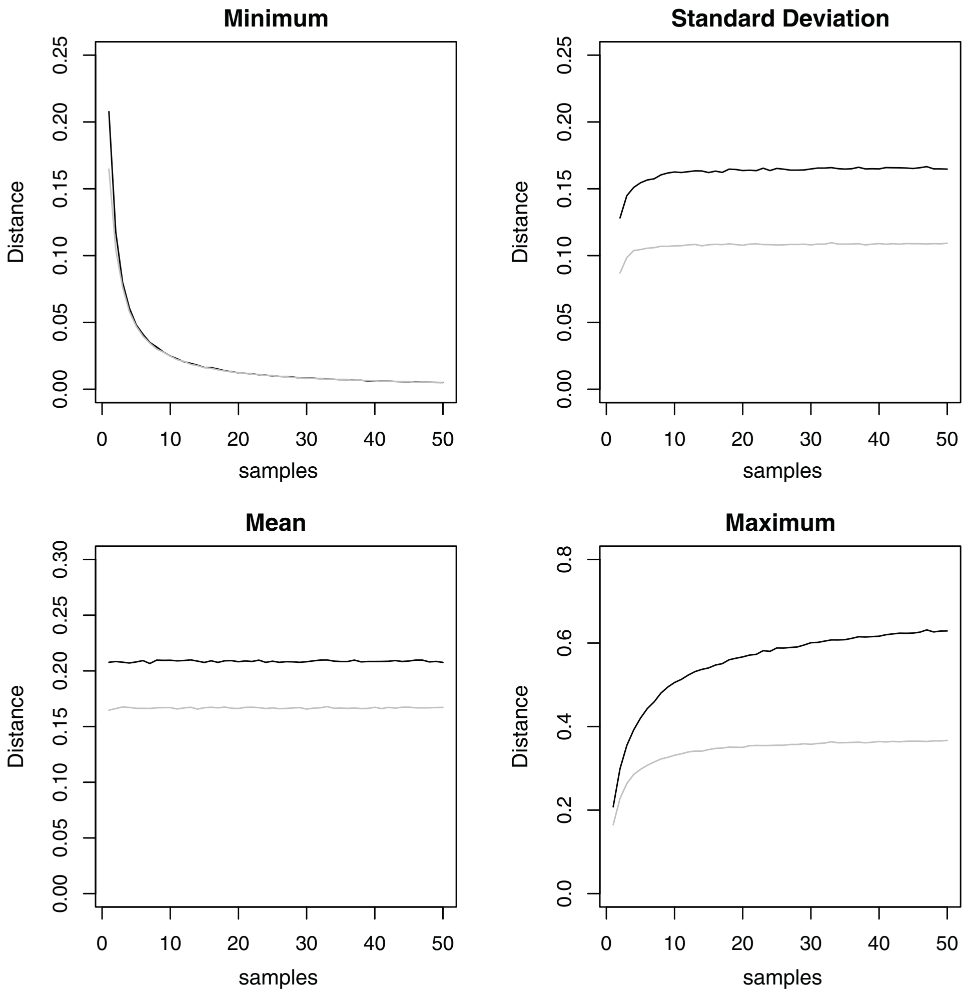


**Fig. 12.** Comparison of minimum (upper left), mean (lower left) and maximum (lower right) DTO for ensemble size of 1, . . ., 50 solution pairs. The standard deviation of an ensemble DTO is also given (upper right). In all cases the RBS is represented by the black line and OBS with the gray line.

The maximum DTO is also recorded and shown in the lower right plot of Fig. 12. The mean maximum distance for OBS is relatively constant and equal to about 0.3 after ensemble size 10. However, when utilizing RBS the mean maximum DTO does not show a limiting behavior over this sample size and is nearly twice as large as when using OBS. When using an ensemble it is important to efficiently search in the local area[7] of the optimal solution and therefore these larger distances when using RBS are not desirable.

Fig. 13 compares the probability of the ensemble mean to be closer to the optimal solution for 1D problems. The ensemble size is varied between 1, . . ., 50 where the experimental design is as utilized above. The solid gray line corresponds to the situation where OBS < RBS, the black line represents OBS > RBS and the dashed gray line is used to show OBS = RBS, respectively. As would be expected, the probability OBS = RBS quickly reduces to 0 as the ensemble size increases. This reduction initially causes a small increase in probability for RBS. When the probability RBS = OBS declines to 0, the probability Pr(OBS > RBS) also continues to reduce. The Pr(OBS < RBS) consistently increases as the ensemble/sample size increases. These probabilities seem to converge at approximately Pr(OBS < RBS) = 0.8 and Pr(OBS > RBS) = 0.2.

## 6. Discussion

To this point evidence supporting the utility of OBS has been provided using a purely distance-based argument. The main proof demonstrates that under a uniformly distributed solution assumption, OBS is more likely to yield a guess near the optimal solution than a uniformly chosen paired sample. However, in practise a measure of distance to the optimal in representation space may not be possible to compute due to lack of information about the location of the optimal solution. The choice of algorithm $\mathcal{A}$ and parameterization will thus affect the usefulness of OBS.

The other main factors influencing the usefulness of an OBS-based approach are the structure of the search-evaluation space $\mathcal{L}$ as well as the definition of opposite employed $\mathcal{O}$. Indeed it may be possible that OBS could yield poorer performance than alternative sampling strategies (even uniform-based sampling). Without loss of generality the following will assume a maximization problem. It is reasonable to believe that one could be confident of an improvement using OBS for $f : \mathbb{R}^n \mapsto \mathbb{R}$ if,

$$\begin{aligned}&\Pr(\max(f(x_1), \ldots, f(x_{n/2}), f(\breve{x}_1), \ldots, f(\breve{x}_{n/2}))\\&\quad > \max(f(x_1), \ldots, f(x_n))) > 0.5. \qquad (21)\end{aligned}$$

[7] Recall, we are not considering the evaluation of a solution.

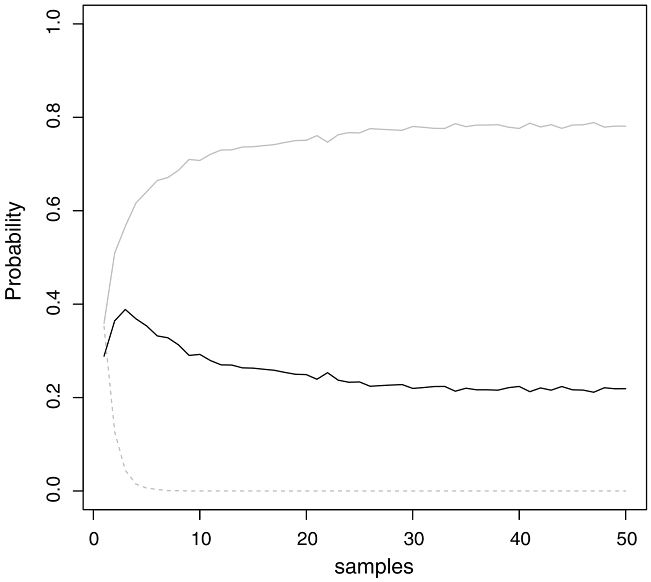


**Fig. 13.** The probability of $p_R$, $p_X$, $p_{\breve{X}}$ having closest mean distance to a random optima given an ensemble of certain sample size. The dashed gray line represents the situation where OB = RBS, the black line corresponds to RBS being closer than OBS and the solid gray depicts the case where OBS is closer than RBS. OBS shows an increase in probability up to ensemble size of 50 samples whereas RBS slightly increases for small sample sizes, then decreases to approximately 0.20. This evidence strongly supports the use of OBS.

Assuming (21) is satisfied, this implies that

$$\mathbb{E}\left[\underbrace{\max(f(x_1),\ldots,f(x_{n/2}),f(\breve{x}_1),\ldots,f(\breve{x}_{n/2}))}_{\text{OBS}}\right] > \mathbb{E}\left[\underbrace{\max(f(x_1),\ldots,f(x_n))}_{\text{ALT}}\right], \qquad (22)$$

where OBS corresponds to the opposition-based sampling method and ALT represents any alternative approach, respectively. We can explicitly state the dependence on solution landscape ($\mathcal{L}$), definition of opposite ($\mathcal{O}$) and algorithm ($\mathcal{A}$),

$$\mathbb{E}[\text{OBS}|\mathcal{L},\mathcal{O},\mathcal{A}] > \mathbb{E}[\text{ALT}|\mathcal{L},\mathcal{O},\mathcal{A}]. \qquad (23)$$

Further analysis of $\mathcal{L}$, $\mathcal{O}$, $\mathcal{A}$ and their impact on solution quality and convergence rates are underway. However, we provide the following examples to highlight when and under what circumstances OBS may yield benefits and when it may be detrimental, with respect to $\mathcal{L}$.

It is important to realize using OBS implies a transformation of the evaluation function w.r.t. the definition of opposite. This is because OBS is a (static) dependent sampling approach whereby one returns the most desirable evaluation between a pair of guesses that have a constant functional relationship. That is, given a guess we will always compute the same opposite for it and the most desirable evaluation between the two will be retained, i.e., we should always have $f(x) = f(\breve{x}) \equiv \max(f(x), f(\breve{x}))$.

### 6.1. Example 1: symmetric evaluation function

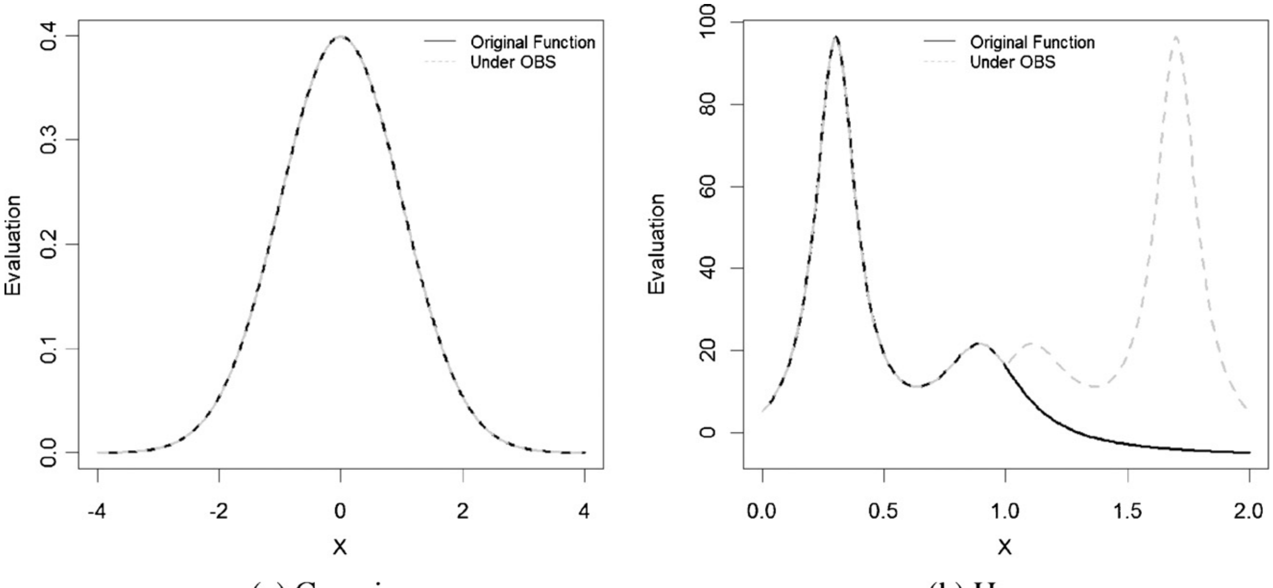
Assume the evaluation function is symmetric about zero. For instance, consider the 1-dimensional Gaussian function (Fig. 14(a))

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)/\sigma)^2}. \qquad (24)$$

Using Definition 1, $f(x) = \max(f(x), f(\breve{x}))$, for $x, \breve{x} \in \mathbb{R}$. In this particular scenario, $f(x) = f(\breve{x})$. Consider a uniform-sampling alternative with paired samples $(x, y)$, for $x \neq y$ and $f(x) \neq f(y)$. Then, it follows that the expected value of the joint distribution of possible evaluations under each sampling scheme will be

$$\mathbb{E}_{ALT}[f(\mathbf{x})] > \mathbb{E}_{OBS}[f(\mathbf{x})] \qquad (25)$$

for $\mathbf{x}$ the vector of samples. Thus, we would not expect using OBS will yield any benefits over uniform sampling in this situation since both samples $(x, \breve{x})$ are identical. For any alternative methods, two random samples $(x, y)$ will not likely be equal and the maximum of these values is taken as the sample.



(a) Gaussian

(b) Humps

**Fig. 14.** Example evaluation functions when considering independent uniform sampling versus OBS dependent sampling. (a) shows a symmetric Gaussian function and (b) depicts the non-symmetric Matlab humps function.

### 6.2. Example 2: non-symmetric evaluation function

Let us now consider the non-symmetric "humps" function

$$f(x) = \frac{1}{(x-0.3)^2+0.01} + \frac{1}{(x-0.9)^2+0.04} - 6. \tag{26}$$

This function is plotted in Fig. 14(b) and shown as the solid black line. This graph represents the evaluations being sampled from under the alternative sampling scheme. We will use *ALT* as uniform sampling, as above. The gray dashed line in the same plot gives the result[8] of considering OBS and under the assumption of a maximization problem.[9] Clearly, the expected value of the joint distribution over function evaluations is higher for the OBS approach, i.e.,

$$\mathbb{E}_{ALT}[f(\mathbf{x})] < \mathbb{E}_{OBS}[f(\mathbf{x})] \tag{27}$$

Additionally, in this case under the OBS scheme we have a lower variance than using uniform sampling and thus yielding a higher solution reliability. Therefore although OBS is a generally applicable method, it is useful only when the solution landscape well corresponds to the definition of opposition and the optimization algorithm is suitable designed.

## 7. Conclusion and future work

This paper reviews the application and existence of the opposition concept in the real-world, and also approaches that have aimed to employ these principles in soft computing methodologies. We mathematically and experimentally examine the intuition behind opposition-based sampling from the viewpoint of distance-to-optimal solution in a given search space. Our main theorem calculates the probability of opposite versus random paired sampling and shows that the distance to an unknown solution is lower under the OBS approach. Experiments confirm the main theorem statement by varying problem dimensionality, solution location and ensemble size of the sample. We also discuss the practicality of this result and provide two examples which highlight when OBS may be useful or detrimental (vs. uniform sampling).

In this paper, we assume that the algorithm being employed is able to utilize the distance-to-optimal solution information. However, in practise information may not be accessible. We provide examples of symmetric and non-symmetric evaluation functions to give an idea of expected behavior without regard to a specific optimization approach. These results indicate the practitioner should use prior knowledge or intuition about the evaluation function when considering OBS. Further analysis into this aspect of the problem are underway as are developing adaptive methods based on approximated shape of the search-evaluation space $\mathcal{L}$.

To date, our understanding of OBS is very limited. Future endeavors will concentrate on fully exploring the influencing factors of evaluation-search space, algorithm and definition of opposite being employed. A comprehensive and mathematical understanding of under what circumstances OBS is beneficial (and/or detrimental) is required. One could additionally formulate the decision of best opposite function as the one that maximizes the difference between the left and right hand sides of Eq. (23). Other approaches include self-adaptive algorithms that incorporate acquired knowledge of the search-evaluation space. Indeed, there are numerous interesting venues for future work.

[8] Recall, $f(x) = \max(f(x), f(\breve{x}))$.

[9] For minimization, we simply allow $f(x) = \min(f(x), f(\breve{x}))$.

## References

[1] S. Rahnamayan, Opposition-based differential evolution, Ph.D. Thesis, Department of Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada, May 2007.

[2] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition versus randomness in soft computing techniques, Journal on Applied Soft Computing 8 (March) (2008) 906–918.

[3] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition-based differential evolution algorithms, in: IEEE World Congress on Computational Intelligence, Vancouver, Canada, 2006, pp. 7363–7370.

[4] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition-based differential evolution for optimization of noisy problems, in: IEEE World Congress on Computational Intelligence, Vancouver, Canada, 2006, pp. 6756–6763.

[5] S. Das, N. Suganthan, Differential evolution: a survey of the state-of-the-art, IEEE Transactions on Evolutionary Computation 11 (February (1)) (2011) 4–31.

[6] F. Neri, V. Tirronen, Recent advances in differential evolution: a review and experimental analysis, Artificial Intelligence Review, Springer 33 (February (1)) (2010) 61–106.

[7] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition-based differential evolution, in: Advances in Differential Evolution, Series: Studies in Computational Intelligence, Springer-Verlag, 2008, ISBN: 978-3-540-68827-3.

[8] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Differential evolution via exploiting opposite populations, in: Oppositional Concepts in Computational Intelligence, Series: Studies in Computational Intelligence, Springer-Verlag, 2008, ISBN: 978-3-540-70826-1.

[9] H.R. Tizhoosh, M. Ventresca, S. Rahnamayan, Opposition-based computing, in: Oppositional Concepts in Computational Intelligence, Series: Studies in Computational Intelligence, Springer-Verlag, 2008, ISBN: 978-3-540-70826-1.

[10] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Quasi-oppositional differential evolution, in: IEEE Congress on Evolutionary Computation (CEC-2007), Singapore, September 2007, pp. 2229–2236.

[11] R. Balamurugan, S. Subramanian, Emission-constrained dynamic economic dispatch using opposition-based self-adaptive differential evolution algorithm, International Energy Journal 10 (December (4)) (2009).

[12] L. Peng, Y. Wang, Differential evolution using uniform-quasi-opposition for initializing the population, Information Technology Journal 9 (8) (2010) 1629–1634.

[13] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition-based differential evolution (ODE) with variable jumping rate, in: Proc. of IEEE Symposium on Foundations of Computational Intelligence, Honolulu, Hawaii, USA, April 2007, pp. 81–88.

[14] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition-based differential evolution, IEEE Transactions on Evolutionary Computation 12 (February (1)) (2008) 64–79.

[16] S. Rahnamayan, H.R. Tizhoosh, Image thresholding using micro opposition-based differential evolution (micro-ODE), in: IEEE World Congress on Computational Intelligence (WCCI-2008), Hong Kong, June 2008, pp. 1409–1416.

[17] S. Rahnamayan, G. Gary Wang, Investigating in scalability of opposition-based differential evolution, in: 8th WSEAS International Conference on Simulation, Modeling and Optimization (SMO'08), Santander, Cantabria, Spain, September 23–25, 2008, pp. 105–111.

[18] H.R. Tizhoosh, Opposition-based learning: a new scheme for machine intelligence, in: International Conf. on Computational Intelligence for Modelling Control and Automation (CIMCA'2005), vol. I, Vienna, Austria, 2005, pp. 695–701.

[19] H.R. Tizhoosh, Reinforcement learning based on actions and opposite actions, in: ICGST International Conference on Artificial Intelligence and Machine Learning (AIML-05), Cairo, Egypt, 2005.

[20] H.R. Tizhoosh, Opposition-based reinforcement learning, Journal of Advanced Computational Intelligence and Intelligent Informatics 10 (4) (2006) 578–585.

[21] H.R. Tizhoosh, M. Ventresca (Eds.), Oppositional Concepts in Computational Intelligence, Physica-Verlag, Springer, 2008.

[22] A. Bhattacharya, P.K. Chattopadhyay, Solution of economic power dispatch problems using oppositional biogeography-based optimization, Electric Power Components and Systems 38 (July (10)) (2010) 1139–1160.

[23] M. Ergezer, D. Simon, D. Du, Oppositional biogeography-based optimization, in: IEEE Conference on Systems, Man, and Cybernetics, 2009, pp. 1035–1040.

[24] M. Shokri, H.R. Tizhoosh, M.S. Kamel, Opposition-based $Q(\lambda)$ algorithm, in: 2006 IEEE World Congress on Computational Intelligence (IJCNN-2006), Vancouver, Canada, 2006, pp. 646–653.

[25] M. Shokri, H.R. Tizhoosh, M.S. Kamel, Opposition-based $Q(\lambda)$ with non-Markovian update, in: Proceedings of IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning (ADPRL-2007), Hawaii, 2007, pp. 288–295.

[26] M. Shokri, H.R. Tizhoosh, M.S. Kamel, Tradeoff between exploration and exploitation of $OQ(\lambda)$ with non-Markovian update in dynamic environments, in: IEEE World Congress on Computational Intelligence (WCCI-2008), Hong Kong, June 2008, pp. 2916–2922.

[27] M. Ventresca, H.R. Tizhoosh, Improving the convergence of backpropagation by opposite transfer functions, in: 2006 IEEE World Congress on Computational Intelligence (IJCNN-2006), Vancouver, Canada, 2006, pp. 9527–9534.

[28] M. Ventresca, H.R. Tizhoosh, Opposite transfer functions and backpropagation through time, in: Proceedings of IEEE Symposium on Foundations of Computational Intelligence (FOCI-2007), Hawaii, April 1–5, 2007, pp. 570–577.

[29] M. Ventresca, H.R. Tizhoosh, Simulated annealing with opposite neighbors, in: Proceedings of IEEE Symposium on Foundations of Computational Intelligence (FOCI-2007), Hawaii, April 1–5, 2007, pp. 186–192.

[30] M. Ventresca, H.R. Tizhoosh, Numerical condition of feedforward networks with opposite transfer functions, in: IEEE World Congress on Computational Intelligence (WCCI-2008), Hong Kong, June 2008, pp. 3232–3239.

[31] M. Ventresca, H.R. Tizhoosh, A diversity maintaining population-based incremental learning algorithm, Information Sciences 178 (21) (2008).

[32] G.H. Mahamed, Z. Omran, A. Woo Geem, Salman, Improving the performance of harmony search using opposition-based learning and quadratic interpolation, International Journal of Mathematical Modelling and Numerical Optimisation 2 (1) (2011) 28–50.

[33] F. Sahba, H.R. Tizhoosh, M.M.A. Salama, Application of opposition-based reinforcement learning in image segmentation, in: Proceedings of IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning (ADPRL-2007), Hawaii, April 1–5, 2007, pp. 246–251.

[34] M. Mahootchi, H.R. Tizhoosh, K. Ponnambalam, Opposition-based reinforcement learning in the management of water resources, in: Proceedings of IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning (ADPRL-2007), Hawaii, April 1–5, 2007, pp. 217–224.

[35] A.R. Malisia, H.R. Tizhoosh, Applying opposition-based ideas to the ant colony system, in: Proceedings of IEEE Swarm Intelligence Symposium (SIS-2007), Hawaii, April 1-5, 2007, pp. 182–189.

[36] A.R. Malisia, Investigating the Application of Opposition-Based Ideas to Ant Algorithms, M.Sc. Thesis, Department of Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada, Sep. 2007.

[37] J. Bresta, A. Zamuda, B. Bošković, M.S. Maučec, V. Žumer, High-dimensional real-parameter optimization using self-adaptive differential evolution algorithm with population size reduction, in: IEEE World Congress on Computational Intelligence (WCCI-2008), Hong Kong, June 2008, pp. 2032–2039.

[38] L. Han, X. He, A novel opposition-based particle swarm optimization for noisy problems, in: Third International Conference on Natural Computation (ICNC-2007), 2007, pp. 624–629.

[39] B. Bošković, S. Greiner, J. Brest, A. Zamuda, V. Žumer, an adaptive differential evolution algorithm with opposition-based mechanisms, applied to the tuning of a Chess program, in: U.K. Chakraborty (Ed.), Advances in Differential Evolution, Studies in Computational Intelligence, vol. 143, Springer, June 2008.

[40] H. Wang, Y. Liu, S. Zeng, H. Li, C. Li, Opposition-based particle swarm algorithm with Cauchy mutation, in: IEEE Congress on Evolutionary Computation (CEC-2007), Singapore, September 2007, pp. 4750–4756.

[41] B. Sharif, G.G. Wang, T. El-Mekkawy, Mode pursing sampling method for discrete variable optimization on expensive black-box functions, ASME Transactions, Journal of Mechanical Design 130 (2008), pp. 021402-1-11.

[42] L. Wang, S. Shan, G.G. Wang, Mode-pursuing sampling method for global optimization on expensive black-box functions, Journal of Engineering Optimization 36 (August (4)) (2004) 419–438.

[45] M.G. Omran, Using opposition-based learning with particle swarm optimization and barebones differential evolution, in: Particle Swarm Optimization, InTech Education and Publishing, 2009, pp. 343–384 (Chapter 23).

[46] H. Lin, H. Xingshi, A novel opposition-based particle swarm optimization for noisy problems, in: Third International Conference on Natural Computation, ICNC, vol. 3, 2007, pp. 624–629.

[47] H. Jabeen, Z. Jalil, A.R. Baig, Opposition based initialization in particle swarm optimization (O-PSO), in: The 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference, 2009, pp. 2047–2052.

[48] F. Shahzad, A.R. Baig, S. Masood, M. Kamran, N. Naveed, Opposition-based particle swarm optimization with velocity clamping (OVCPSO), in: Advances in Computational Intelligence, vol. 116, Springer Berlin, 2009, pp. 339–348.

[49] M. Weber, V. Tirronen, F. Neri, Scale factor inheritance mechanism in distributed differential evolution, Soft Computing—A Fusion of Foundations, Methodologies and Applications, Springer 14 (September (11)) (2010) 1187–1207.

[50] J. Lampinen, I. Zelinka, On stagnation of the differential evolution algorithm, in: The 6th International Mendel Conference on Soft Computing, 2000, pp. 76–83.

[51] H-Y. Fan, J. Lampinen, A trigonometric mutation operation to differential evolution, Journal of Global Optimization 7 (1) (2003) 105–129.

[52] N. Noman, H. Iba, Accelerating differential evolution using an adaptive local search, IEEE Transaction in Evolutionary Computation 2 (1) (2008) 107–125.

[53] V. Tirronen, F. Neri, T. Kärkkä inen, K. Majava, T. Rossi, An enhanced memetic differential evolution in filter design for defect detection in paper production, Evolutionary Computation 16 (2008) 529–555.

[54] A. Caponio, F. Neri, V. Tirronen, Super-fit control adaptation in memetic differential evolution frameworks, Soft Computing—A Fusion of Foundations, Methodologies and Applications 13 (8) (2000) 811–831.

[55] F. Neri, V. Tirronen, Scale factor local search in differential evolution, Memetic Computing Journal 1 (2) (2009) 153–171.

[56] J. Brest, V. Žumer, M. Maucec, Self-adaptive differential evolution algorithm in constrained real-parameter optimization, in: Proceedings of the IEEE Congress on Evolutionary Computation, 2006, pp. 215–222.

[57] J. Brest, S. Greiner, B. Boškovíc, M. Mernik, V. Žumer, Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems, IEEE Transaction in Evolutionary Computation 10 (6) (2006) 646–657.

[58] J. Brest, A. Zamuda, B. Boškovíc, M. Maucec, V. Žumer, High-dimensional real-parameter optimization using self-adaptive differential evolution algorithm with population size reduction, in: Proceedings of the IEEE World Congress on Computational Intelligence, 2008, pp. 2032–2039.

[59] A. Zamuda, J. Brest, B. Boškovíc, V. Žumer, Differential evolution for multiobjective optimization with self adaptation, in: Proceedings of the IEEE World Congress on Computational Intelligence, 2007, pp. 3617–3624.

[60] U.K. Chakraborty, S. Das, A. Konar, Differential evolution with local neighborhood, in: Proceedings of the IEEE Congress on Evolutionary Computation, 2006, pp. 2042–2049.

[61] S. Das, A. Abraham, U.K. Chakraborty, Differential evolution with a neighborhood-based mutation operator, in: Proceedings of the IEEE Congress on Evolutionary Computation, 2009, pp. 526–553.

[62] A.K. Qin, P.N. Suganthan, Self-adaptive differential evolution algorithm for numerical optimization, in: Proceedings of the IEEE Congress on Evolutionary Computation, vol. 2, 2005, pp. 1785–1791.

[63] A.K. Qin, V.L. Huang, P.N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, IEEE Transaction in Evolutionary Computation 13 (2) (2009) 398–417.

[64] Z. Yang, K. Tang, X. Yao, Self-adaptive differential evolution with neighborhood search, in: Proceedings of the World Congress on Computational Intelligence, 2008, pp. 1110–1116.

[65] A. Zamuda, J. Brest, B. Boškovíc, V. Žumer, Large scale global optimization using differential evolution with self-adaptation and cooperative co-evolution, in: Proceedings of the IEEE World Congress on Computational Intelligence, 2008, pp. 3719–3726.

[66] O. Olorunda, A. Engelbrecht, Differential evolution in high-dimensional search spaces, in: Proceedings of the IEEE Congress on Evolutionary Computation, 2007, pp. 1934–1941.

[67] Z. Yang, K. Tang, X. Yao, Differential evolution for high-dimensional function optimization, in: Proceedings of the IEEE Congress on Evolutionary Computation, 2007, pp. 3523–3530.

[68] N. Noman, H. Iba, Enhancing differential evolution performance with local search for high dimensional function optimization, in: Conference on Genetic and Evolutionary Computation, ACM, New York, 2005, pp. 967–974.

[69] Y. Gao, Y.-J. Wang, A memetic differential evolutionary algorithm for high dimensional functions' optimization, in: Proceedings of the 3rd International Conference on Natural Computation, 2007, pp. 188–192.