

Center-Based Sampling for Population-Based Algorithms

Shahryar Rahnamayan, *Member, IEEE*, G. Gary Wang

Abstract—Population-based algorithms, such as Differential Evolution (DE), Particle Swarm Optimization (PSO), Genetic Algorithms (GAs), and Evolutionary Strategies (ES), are commonly used approaches to solve complex problems from science and engineering. They work with a population of candidate solutions. In this paper, a novel center-based sampling is proposed for these algorithms. Reducing the number of function evaluations to tackle with high-dimensional problems is a worthwhile attempt; the center-based sampling can open a new research area in this direction. Our simulation results confirm that this sampling, which can be utilized during population initialization and/or generating successive generations, could be valuable in solving large-scale problems efficiently. Quasi-Oppositional Differential Evolution is briefly discussed as an evidence to support the proposed sampling theory. Furthermore, opposition-based sampling and center-based sampling are compared in this paper. Black-box optimization is considered in this paper and all details about the conducted simulations are provided.

I. INTRODUCTION

Population-based algorithms are utilized to solve real-world complex problems. These algorithms start with a randomly generated candidate solutions when there is no a priori knowledge about the location of the global optima. We call this process population initialization.

There are various sampling methods (such as Normal, Halton, Sobol, and Faure). Applying these methods to initialize the population can affect the best found objective function value. Effects of population initialization are noticeable when we solve real-life problems (expensive optimizations) and when the algorithm has been stopped prematurely because of a long computation time [1]. It means the best found objective function value is different just in early generations. Generally, the effects of population initialization diminish when the dimensionality of the search space increases and the population becomes highly sparse [1]. In the current paper, to address this shortcoming, a new sampling approach is proposed as a strategy to combat the curse of dimensionality.

S. Rahnamayan is with the Faculty of Engineering and Applied Science, University of Ontario Institute of Technology (UOIT), 2000 Simcoe Street North, Oshawa, Ontario, Canada L1H 7K4 (phone: +1-905-721-8668 Ext. 3843; email: shahryar.rahnamayan@uoit.ca). **G.G. Wang** is with the School of Engineering Science, Simon Fraser University, 250-13450 102 Avenue Surrey, BC, Canada V3T 0A3 (phone: +1-778-782-8495; email: gary_wang@sfu.ca).

Sometimes the sampling methods are used not only in initialization stage but also during the search, learning, and optimization processes. To mention some examples, Random Search (RS) and Mode-Pursuing Sampling (MPS) methods [2], [3] use sampling during the optimization process.

Opposition-based learning (OBL) was introduced by Tizhoosh in 2005 [4]. 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, OBL was applied to introduce opposition-based differential evolution (ODE) [5], [6], opposition-based reinforcement learning (ORL) [7], and opposition-based neural networks [8]. All of these algorithms have tried to enhance searching or learning in different fields of soft computing by utilizing opposition-based sampling. This paper also partially compares the proposed center-based sampling with the opposition-based sampling.

The authors try to investigate the possibility of introducing better point(s) than the opposite point to enhance all aforementioned opposition-based algorithms. Simulation results in this paper are promising and confirm the mentioned possibility. This paper can be considered as a first step towards a novel sampling method which is specialized for high-dimensional problems.

The paper is organized as follows: The proposed sampling theory with all corresponding simulation results are presented in Section II. The center and opposite points are compared over different search space dimensions in Section III. Quasi-Oppositional Differential Evolution (QODE), a promising evidence for the proposed sampling method, is described briefly in Section IV. The paper is concluded in Section V.

II. PROPOSED SAMPLING THEORY

Before explaining the proposed sampling theory, we need to conduct some simulations to answer following questions:

- 1) For a black-box problem (no a priori knowledge about the location of the solution), do all points in the search interval have the same chance to be closer to an unknown solution compared to a randomly generated point?

2) If the answer for the first question is no, what is the pattern for this closeness probability? And whether does this pattern remain the same for all search space dimensions? Following conducted simulations will answer properly to all these questions.

A. Closeness to Solution: Simulation-Based Investigation

Let us start with the probability definitions which have been calculated in our simulations.

Definition 1 The probability of closeness to an unknown solution (s) for the candidate solution (x) and a random point (r) are defined as follows:

$$p_x = p[d(x, s) < d(r, s)], \quad (1)$$

$$p_r = p[d(r, s) < d(x, s)], \quad (2)$$

where d is Euclidean distance and p stands for probability function.

Algorithm 1 implements our simulation (Monte Carlo method) to calculate p_x for D-dimensional search space (where x is a D-dimensional vector with the same value for all elements). Figure 1 and Figure 2 depict the results for some sample dimensions (1D, 2D, 3D, 5D, ..., 1000D) graphically.

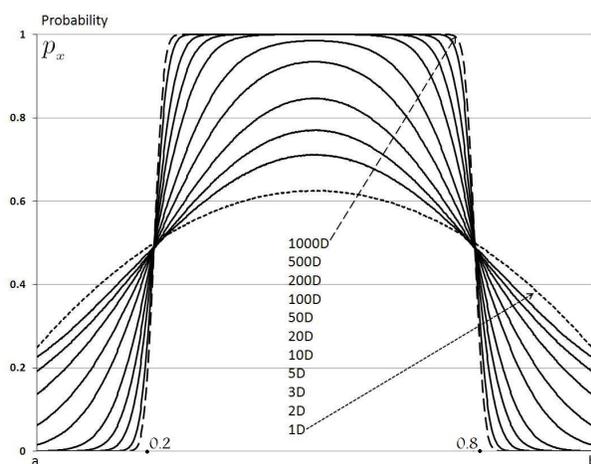


Fig. 1. Probability of the closeness of $\vec{x} = [x, x, \dots, x]$, $x \in [a, b] = [0, 1]$ (p_x , where x is a D-dimensional vector with the same value for all elements) to a uniformly generated random solution compared to the closeness probability of a uniformly generated second random point to that solution. Utilizing a vector with the same value for all elements helps us to show 2D map for higher dimensions. By this way, the points on the diameter are investigated.

As seen in Figure 1, the points which are closer to the center of the search space have a higher chance

Algorithm 1 Calculating p_x (probability of closeness of x to a random solution) and \bar{d}_x (average distance of x from a random solution) by the simulation.

```

1:  $x_i \in [a_i, b_i] = [0, 1]$  where  $i = 1, 2, 3, \dots, D$ 
2:  $\bar{d}_x = 0$ ,  $\bar{d}_r = 0$ 
3:  $c_x = 0$ ,  $c_r = 0$ 
4: TRIALS  $\leftarrow 10^6$ 
5: for  $\vec{x}$  = a to b (stepsize:  $10^{-3}$ ,  $\vec{x}$  is a vector with
   the same value for all elements) do
6:   for  $R = 1$  to TRIALS do
7:     Generate two random points  $\vec{s}$  and  $\vec{r}$  in the D-
       dimensional space (use interval  $[0, 1]$  for each
       dimension)
8:     Calculate the Euclidean distance of  $\vec{x}$  and  $\vec{r}$ 
       from solution  $\vec{s}$  ( $d_x$  and  $d_r$ )
9:      $\bar{d}_x \leftarrow \bar{d}_x + d_x$ 
10:     $\bar{d}_r \leftarrow \bar{d}_r + d_r$ 
11:    if ( $d_x < d_r$ ) then
12:       $c_x \leftarrow c_x + 1$ 
13:    else if ( $d_r < d_x$ ) then
14:       $c_r \leftarrow c_r + 1$ 
15:    end if
16:  end for
17:   $\bar{d}_x \leftarrow \bar{d}_x / \text{TRIALS}$ 
18:   $\bar{d}_r \leftarrow \bar{d}_r / \text{TRIALS}$ 
19:   $p_x \leftarrow c_x / \text{TRIALS}$ 
20:   $p_r \leftarrow c_r / \text{TRIALS}$ 
21:  Save  $\bar{d}_x$  and  $\bar{d}_r$  for  $\vec{x}$ 
22:  Save  $p_x$  and  $p_r$  for  $\vec{x}$ 
23: end for

```

to be closer to unknown solution as well. This chance increases directly with the dimensionality of the search space. Accordingly, the average distance of the points which are closer to the center from the unknown solution is lower (Figure 2) and similarly decreases sharply for the higher dimensions. Obviously, the center point has the maximum chance to be closer to the solution and at the same time the minimum average distance from the solution. That is a clear evidence which shows that why a center point is a valuable point.

Now, we want to investigate the probability of the closeness of the center point (p_c) to the solution, compared to a second random point. The simulation results are presented in Figure 3. As shown, p_c increases sharply with the dimension and interestingly for the higher dimensions ($D > 30$) it is very close (converges) to one.

The center point is a unique point and can not be utilized for sampling in the population-based algorithms because of diversity problem. Let us look at Figure 1

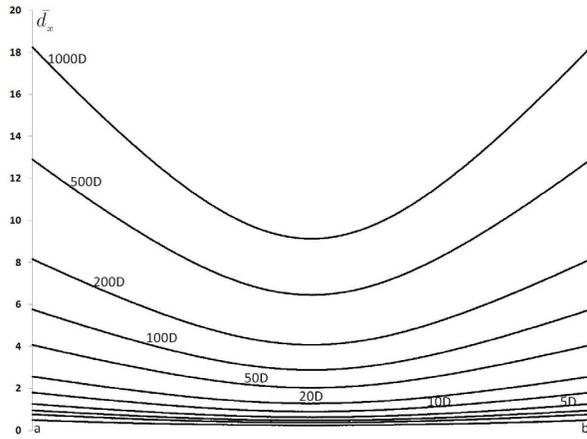


Fig. 2. Average distance from random solution (\bar{d}_x , $\vec{x} = [x, x, \dots, x]$, $x \in [a, b]$) for different search space dimensions.

again, the middle part of the graph is flat when the dimensionality of the search space increases toward a very big number (e.g., 500D or 1000D). It happens in interval $[0.2, 0.8]$ which means 60% of the interval's middle part. Now, this time we generate a uniform random number in this interval $([0.2, 0.8], p_c)$ and compare its closeness to solution with a second uniform random number's closeness generated over the whole interval $([0, 1], p_r)$. The result is given in Figure 4. By comparing Figures 3 and 4, we notice that for the first one p_c increases faster than the second one, although, both of them converge to one for higher dimensions ($D > 30$ and $D > 100$ for the first and second graphs, respectively). It was predictable because relaxation of the center point over a sub-interval can reduce the closeness probability value.

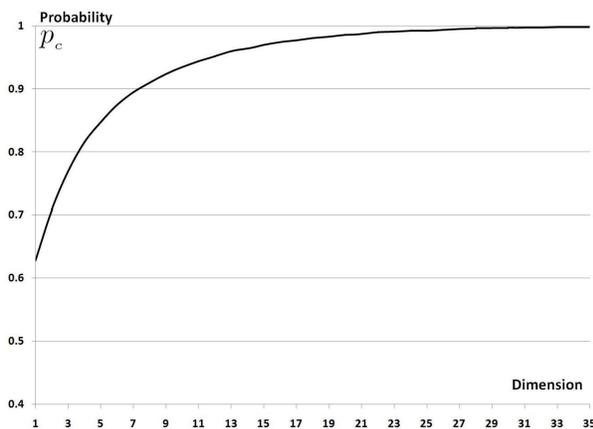


Fig. 3. Probability of center-point closeness to solution (compared to a uniformly generated random point, $p_c + p_r = 1$) versus dimension of search space.

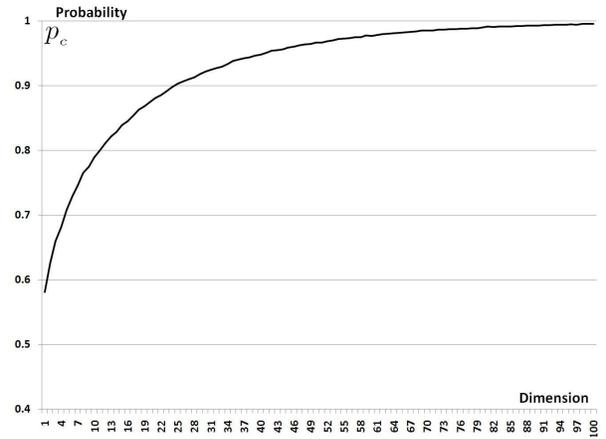


Fig. 4. Probability of closeness to solution (a uniform random point generated in $[0.2, 0.8]$ and the second one is generated in $[0, 1]$, $p_c + p_r = 1$) versus dimension of search space.

B. Result Analysis

Our simulation results confirm that when the sampling points are closer to the center of the search space they have a higher chance to be closer to an unknown solution. Also in average, their distance from solution is lower as well. Furthermore, for higher dimensions the mentioned advantageous increases sharply and for very high dimensions (e.g., $D > 1000$) a specific sub-interval (i.e., $[0.2, 0.8]$) presents a flat area for the mentioned probability value ($p \simeq 1$). Also, for these search spaces the population is highly sparse and individuals have a pattern free distribution.

III. CENTER VS. OPPOSITE

Before comparison of the center point and opposite point, we have to review opposite point's definition in one and also higher dimensions.

Definition 2 Let x be a real number in an interval $[a, b]$ ($x \in [a, b]$); the opposite of x , denoted by \check{x} , is defined by

$$\check{x} = a + b - x. \quad (3)$$

Figure 5 (the top figure) illustrates x and its opposite \check{x} in interval $[a, b]$. As seen, x and \check{x} are located in equal distance from the interval's center ($|(a+b)/2 - x| = |\check{x} - (a+b)/2|$) and the interval's boundaries ($|x - a| = |b - \check{x}|$) as well.

This definition can be extended to higher dimensions by applying the same formula to each dimension.

Definition 3 Let $P(x_1, x_2, \dots, x_D)$ be a point in D-dimensional space, where x_1, x_2, \dots, x_D are real numbers

and $x_i \in [a_i, b_i]$, $i = 1, 2, \dots, D$. The opposite point of P is denoted by $\check{P}(\check{x}_1, \check{x}_2, \dots, \check{x}_D)$ where

$$\check{x}_i = a_i + b_i - x_i. \quad (4)$$

Figure 5 illustrates a sample point and its corresponding opposite point in one, two, and three dimensional spaces.

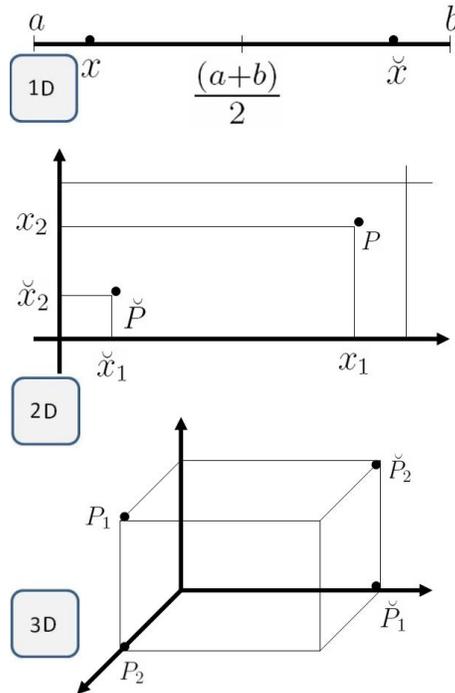


Fig. 5. Illustration of a point and its corresponding opposite for 1D, 2D, and 3D search spaces.

In [10], it was shown mathematically why the opposite point has more chance to be closer to an unknown solution than a randomly generated point. Now, we are ready to compare the opposite and center points in term of closeness to an unknown solution. Algorithm 2 implements the simulation in order to compare these two points. The results are presented in Figure 6. Opposite point performs better just for 1D space, for the rest of the dimensions center point outperforms that. For higher dimensions, the closeness probability of the center point and the opposite point converge to one and zero, respectively.

In order to confirm the simulation results for one-dimensional search space (which opposite point is better than the center point), following mathematical proof is proposed:

Proof In Figure 7 if $x \in [a, c]$ then $\check{x} \in [c, b]$ and vice versa (c stands for the center of $[a, b]$). But we know

that the candidate solution and so its opposite (x and \check{x}) can appear anywhere in the interval $[a, b]$. It means that we can suppose a uniform distribution behavior for their appearance. By this way, their average values (\bar{x} and $\bar{\check{x}}$) would be located in the middle of the half intervals $[a, c]$ and $[c, b]$. Now, just when the solution is in the marked interval ($[k_1, k_2]$), the solution would be closer to c than \bar{x} or $\bar{\check{x}}$, otherwise, it would be closer to \bar{x} or $\bar{\check{x}}$.

So,

$$p_c = \frac{|k_2 - k_1|}{|b - a|}, p_{\bar{x}} = \frac{1}{4} = 0.25, \quad (5)$$

We know that

$$p_x + p_{\bar{x}} + p_c = 1, \quad (6)$$

and

$$p_x = p_{\bar{x}}, \quad (7)$$

And therefore,

$$p_x = 0.375, p_{\bar{x}} = 0.375. \quad (8)$$

These values (p_x , p_c , and $p_{\bar{x}}$) match exactly with our simulation's numerical results for one-dimensional space. For higher dimensions, center point presents higher probability value and converges to one for these dimensions, see Figure 6.

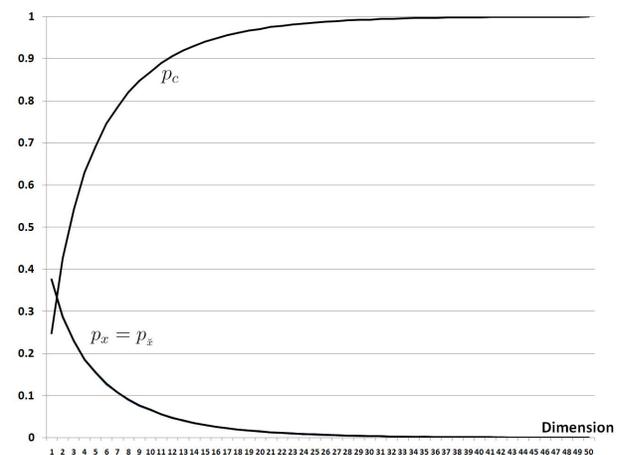


Fig. 6. Simulation results for p_c , p_x , and $p_{\bar{x}}$ on different search space dimensions. As expected, $p_x = p_{\bar{x}}$.

The results of this section confirm that the center point shows better properties (closeness to solution and distance from solution) than the opposite point.

Algorithm 2 Calculating p_x , $p_{\tilde{x}}$, and p_c by the simulation.

```

1:  $c_x = 0$ ,  $c_r = 0$ ,  $c_{\tilde{x}} = 0$ 
2: TRIALS  $\leftarrow 10^6$ 
3:  $x_i \in [a_i, b_i] = [0, 1]$  where  $i = 1, 2, 3, \dots, D$ 
4: for  $R = 1$  to TRIALS do
5:   Generate two random points  $x$  and  $s$  in the  $D$ -
     dimensional search space {use interval  $[0, 1]$  for
     each dimension}
6:   Calculate the opposite point of  $x$  ( $\tilde{x}$ )
7:   Calculate the Euclidean distance of  $x$ ,  $\tilde{x}$ , and  $c$ 
     from  $s$  ( $d_x$ ,  $d_{\tilde{x}}$ ,  $d_c$ )
8:   if  $(d_x < d_{\tilde{x}}) \wedge (d_x < d_c)$  then
9:      $c_x \leftarrow c_x + 1$ 
     { $x$  is the closest to the solution}
10:  else if  $(d_{\tilde{x}} < d_x) \wedge (d_{\tilde{x}} < d_c)$  then
11:     $c_{\tilde{x}} \leftarrow c_{\tilde{x}} + 1$  { $\tilde{x}$  is the closest to the solution}
12:  else if  $(d_c < d_x) \wedge (d_c < d_{\tilde{x}})$  then
13:     $c_c \leftarrow c_c + 1$  { $c$  is the closest to the solution}
14:  end if
15: end for
16:  $p_x \leftarrow c_x/\text{TRIALS}$ 
17:  $p_{\tilde{x}} \leftarrow c_{\tilde{x}}/\text{TRIALS}$ 
18:  $p_c \leftarrow c_c/\text{TRIALS}$ 

```

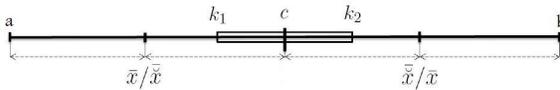


Fig. 7. Calculating p_c for 1D search space. k_1 and k_2 are the centers of intervals $[\tilde{x}/\tilde{x}, c]$ and $[c, \tilde{x}/\tilde{x}]$, respectively.

IV. AN EVIDENCE TO SUPPORT PROPOSED SAMPLING THEORY: QUASI-OPPOSITIONAL DIFFERENTIAL EVOLUTION (QODE)

Opposition-based differential evolution (ODE) was proposed in 2006 [5]. ODE employs opposite points to accelerate differential evolution (DE). Later, opposite points were replaced by quasi-opposite points to introduce quasi-oppositional differential evolution (QODE) [9]. Both of algorithms (ODE and QODE) utilize the same schemes for population initialization and generation jumping. But, QODE uses quasi-opposite points instead of opposite points. A quasi-opposite point is a uniformly generated random point between the middle point (M) and the opposite point. Figure 8 shows the quasi-opposite region/volume for 1D, 2D, and 3D. In these regions/volumes a uniform random point is generated and used as a quasi-opposite point. Experimental results in [9] (conducted on 30 optimization problems) show that QODE outperforms ODE and DE (the parent

algorithm). The number of function calls, success rate, and success performance are three benchmark metrics which have been employed to compare DE, ODE, and QODE in that experimental study.

In average the quasi-opposite points are closer to the center-point compared to the opposite-points. By this way, QODE can be a promising evidence to support proposed center-based sampling theory in this paper.

Quasi-opposition based sampling can reduce diversity of the population faster. This phenomenon can cause a premature convergence. To tackle this problem, applying a diversity maintaining approach to increase diversity of the population can be useful [11]. Further investigation is required.

V. CONCLUSIONS

During solving large-scale problems by population-based algorithms, we are faced with highly sparse populations, very low coverage by candidate solutions. So, smart use of the individuals is highly important. Our simulation results in this study showed that the center point has the highest probability to be closer to an unknown solution than other points in that search space. For higher dimensions this probability value grows sharply. Also, the average distance of the center point from the solution is the lowest. These properties make the center point an special point in the search space. Furthermore, accordingly, the points which are closer to the center have more chance to be closer to the solution as well. For very high dimensions ($D > 1000$), the probability graph presents a flat area in the middle part of the interval (almost on 60% of that). So, uniformly generated samples in this area have the highest probability to be closer to an unknown solution. This observation would be correct even for multi-solution (more than one optima) problems if the solutions are independent from each other.

Comparison of the opposite point and the center point confirms that, except for 1D, center point supports a higher probability value, specially for higher dimensions. QODE can be assumed a promising case study in this direction. But still many unanswered questions remain which will be addressed in our future works. Some of these questions are: which sampling pattern is better to utilize center point's positive properties? Should that be formulated based on dimensionality of the search space? How can we use this pattern to accelerate population-based algorithms? How does the proposed pattern affect the diversity of the population?

The main difference between the opposition-based sampling and center-based sampling is that, first one generate the opposite points just based on the current candidate solutions but the second one can generate new

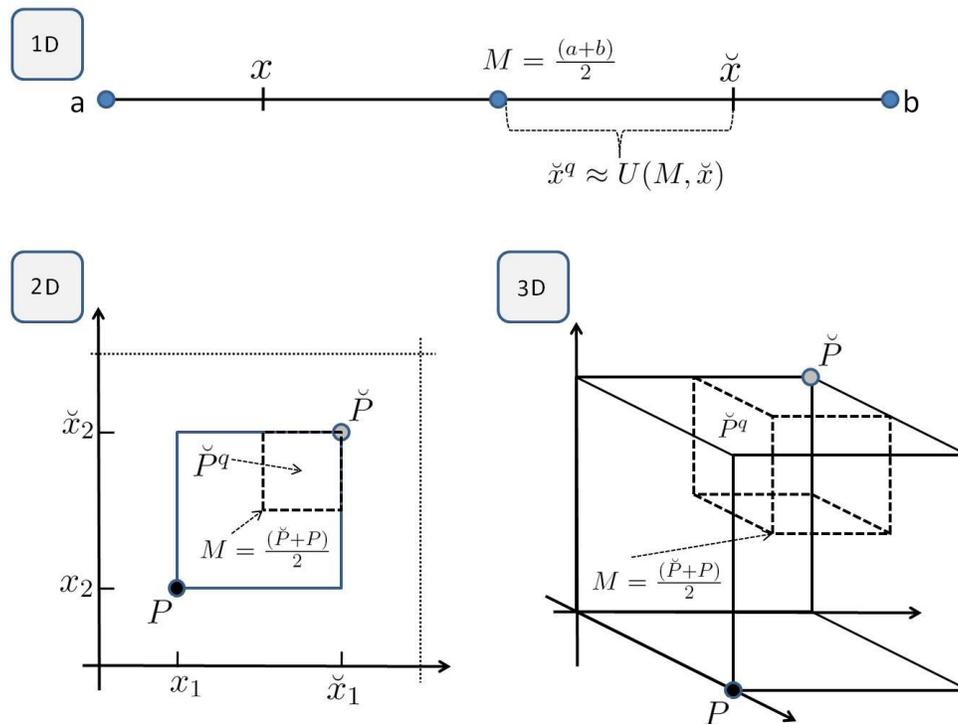


Fig. 8. Illustration of opposite-point and quasi-opposite region for 1D, 2D, and 3D spaces. For 2D and 3D spaces quasi-opposite region is marked by dashed lines.

samples in two ways 1) completely independent from the current solutions or 2) with a degree of dependency to the current solutions (or their opposites, like QODE). Pursuing one of these approaches or a combination of them can result different acceleration schemes for population-based algorithms which builds our current research direction.

The proposed sampling theory not only can be investigated to accelerate population-based algorithms but also, like opposition-based sampling, can be employed to enhance other search, learn, and optimization techniques in soft computing.

References:

- [1] H. Maaranen, K. Miettinen, A. Penttinen, *On initial populations of a genetic algorithm for continuous optimization problems*, Journal of Global Optimization, Vol. 37, No. 3, 2007, pp. 405-436.
- [2] B. Sharif, G.G. Wang, and T. El-Mekkawy, *Mode Pursuing Sampling Method for Discrete Variable Optimization on Expensive Black-box Functions*, ASME Transactions, Journal of Mechanical Design, Vol. 130, 2008, pp.021402-1-11.
- [3] L. Wang, S. Shan, and G.G. Wang, *Mode-Pursuing Sampling Method for Global Optimization on Expensive Black-box Functions*, Journal of Engineering Optimization, Vol. 36, No. 4, August 2004, pp. 419-438.
- [4] H.R. Tizhoosh, *Opposition-Based Learning: A New Scheme for Machine Intelligence*, International Conf. on Computational Intelligence for Modelling Control and Automation (CIMCA'2005), Vienna, Austria, Vol. I, 2005, pp. 695-701.
- [5] S. Rahnamayan, H.R. Tizhoosh, M.M.A Salama, *Opposition-Based Differential Evolution*, IEEE Transactions on Evolutionary Computation, Volume 12, Issue 1, Feb. 2008, pp. 64-79.
- [6] S. Rahnamayan, G. Gary Wang, *Investigating in Scalability of Opposition-Based Differential Evolution*, 8th WSEAS International Conference on Simulation, Modeling and Optimization (SMO'08), Santander, Cantabria, Spain, September 23-25, 2008, pp. 105-111.
- [7] H.R. Tizhoosh, *Opposition-Based Reinforcement Learning*, Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol. 10, No. 4, 2006, pp. 578-585.
- [8] M. Ventresca and H.R. Tizhoosh, *Improving the Convergence of Backpropagation by Opposite Transfer Functions*, 2006 IEEE World Congress on Computational Intelligence (IJCNN-2006), Vancouver, Canada, 2006, pp. 9527-9534.
- [9] S. Rahnamayan, H.R. Tizhoosh, M.M.A Salama, *Quasi-Oppositional Differential Evolution*, IEEE Congress on Evolutionary Computation (CEC-2007), Singapore, Sep. 2007, pp. 2229-2236.
- [10] S. Rahnamayan, H.R. Tizhoosh, M.M.A Salama, *Opposition versus Randomness in Soft Computing Techniques*, Elsevier Journal on Applied Soft Computing, Volume 8, March 2008, pp. 906-918.
- [11] M. Ventresca and H. R. Tizhoosh, *A Diversity Maintaining Population-Based Incremental Learning Algorithm*, Information Sciences, 178(21):4038-4056, 2008.