Duality Search: A novel simple and efficient meta-heuristic algorithm

*Mohsen Shahrouzi[1)]

[1)] *Engineering Department, Kharazmi University, Tehran, Iran*
[1)] *shahruzi@khu.ac.ir*

ABSTRACT

In this article a novel meta-heuristic is proposed based on the concept of duality in nature. Many real world creatures apply this concept since they are found in couples: i.e. families of male and female with their children. Males are usually powerful in some features that females are weak in them and vice versa. Perfect performance of such a couple is dominated by cooperation of both these dual individuals. The present method applies this concept by subdividing the population of search agents into the primary individuals and their duals. Dual of an individual is determined by a fitness ranking procedure. A simple and efficient algorithm is thus proposed utilizing special movements of the artificial couples to search the design space. Several features of the proposed *Duality Search* including its memory-less structure, robustness and efficiency in different search spaces without intrinsic parameter tuning has made it very competitive to some other literature methods. A variety of problems including different test functions and engineering examples are treated to illustrate performance of the proposed method in global optimization.

Keywords: duality search, meta-heuristic, global optimization, diversity measure

1. INTRODUCTION

Many real-world applications require finding their best solution in the form of optimization problems. In this regard, zero-order optimization methods have received considerable attention as they do not require evaluating and approximating gradient of the objective function (Arora, 2004). Most of the meta-heuristic algorithms fall in this category as general-purpose optimization tools for various fields of engineering problems (Yang et al. 2012). They usually simulate some cultural, biological or physical phenomena in nature in order to achieve optimal solution; within a practical time (Kaveh, 2014; Yang, 2013).

The need to search for global optima in the non-convex, multi-modal and constrained design spaces has raised several attempts to develop new algorithms

---

[1)] Ph.D., Faculty of Engineering, Kharazmi University.

which can work in continuous and discrete problems with different complexities. Some of the most recent ones are PDS (Shahrouzi, 2011a), GBMO (Abdeshiri et al. 2013), RMO (Rahmani and Rubiyah, 2014), WWO (Zheng, 2015), DFO (Shahrouzi and Kaveh, 2015), ALO (Mirjalili, 2015), CSA (Akbarzadeh, 2016) and TWO (Kaveh and Zolghadr, 2016). Among them, population-based methods are generally more powerful than single-agent ones in global search.

Parameter tuning is a challenge in practical application of many meta-heuristic algorithms. Due to no-free-lunch theory (Wolpert and Macready, 2005), such parameters should be re-tuned for every other problem to truly achieve its best performance. The tuning process is usually performed via several trial runs with the charge of extra computational effort for implementation of an algorithm to efficiently seek global optimum and to avoid premature convergence. Therefore, fewer control parameters is more interesting from practical point of view. Some recent attempts have addressed this matter including CBO (Kaveh and Mahdavi, 2014, 2015), TWO (Kaveh and Zolghadr, 2016), CSA (Akbarzadeh, 2016), and TLBO (Rao et al. 2011, Rao and Patel, 2013). The most successful attempts have no intrinsic parameters rather than population size and number of iterations (Rao, 2016).

The present work introduces a new parameter-less meta-heuristic called *Duality Search,* DS. It applies a duality difference between capabilities of such males from females in the fitness scope. Consequently, the population of search agents is subdivided to primary individuals and their duals. The main algorithm steps rely on special dual walks toward better positions. The method has also stimulated a child individual that shares features of its parents in the corresponding family unit. A number of illustrative examples are treated to evaluate performance of the proposed method via comparison with some other well-known meta-heuristics.

2. DUALITY SEARCH CONCEPTS

Many creatures in the real-world are divided into male and female types. These two can be considered dual to each other based on their different capabilities in a specific feature. For example, male agents are usually more powerful in hard-working and competition with the others. In the other hand, females are more capable for child treatment than males. Survival of the species in the nature is preserved by cooperative action of males and females in couples rather than by single action of each. Every such a couple is called a family unit including one male and one female.

The idea is extended here to develop an efficient yet simple optimization method called *Duality Search*. In the DS terminology, artificial individuals in a population of search agents are subdivided into primary ones and their duals. Such a duality is defined using the fitness rank of agents so that the fittest member of the population is taken dual to the least fit one. Setting aside these two from the current group of individuals, another couple of duals is identified in the same way and the process is repeated until no further individual is remained for the last set. The population size is essentially even in this process so that finally it can be divided into 2 subpopulations; i.e. primary and dual.

Another concept is difference of walk types in the search space for an individual with its dual. In a single walk, either a primary individual or its dual (not both) can move toward a specific position; e.g. the fittest location that has already found by the algorithm. Besides, when an individual is fitter than its dual they can get farther from each other.

Note that the recent definition of duality uses fitness ranking and differs from definitions in the opposition-based algorithms which apply design variables' scope (Tizhoosh, 2005; Rahnamayan et al. 2008; Han and He, 2007). Another feature is property sharing between a primary individual and its dual by a third type of search agent called a *child*. Such a child provides extra direction of movement in addition to those revealed either by the primary or dual agents. This idea simulates crossover between chromosomes in the *Evolutionary Algorithms* (Back, 1996; He and Wang, 2007; Shahrouzi, 2011a), however, its combination with dual walks is a special feature of the proposed DS. In addition, the generated child is not added to the next population so that population size remains fixed during iterations of the search. The present method does not require any auxiliary memory of the previous individuals.

Since the current study aims to find the best objective function, the aforementioned walks are designated regarding the best position already found as a member of the current population. Such movements are similar to that in the *Differential Evolution,* (Price et al., 2005). As the present method uses fitness-based ranking in order to distinguish couple of dual agents it is expected that when a primary individual move from its position, its dual is simultaneously changed. Hence, search walks in any iteration are only performed for either primary or dual subpopulation rather than for the entire population. As can be realized further, this special feature has provided extra computational efficiency for DS.

3. A DUALITY SEARCH ALGORITHM

Regarding the aforementioned concepts, a DS algorithm is introduced for the fitness maximizing problem via the following steps:

3.1 Initiation

Set the number of total individuals as the prescribed $PopSize$. Randomly generate the entire population of individuals and evaluate their fitness.

3.2 Selection of subpopulation

Sort the population in ascending order of fitness and identify the fittest individual X^{Gbest} as the global best solution. Then determine the index of dual for each individual using the following relation.

$$DualInd_i = PopSize + 1 - i, \quad (1)$$

Divide the entire population into primary and dual subpopulations. Choose either primary or dual subpopulation to modify positions of their individuals.

3.3 Inner Loop

Repeat the following steps for every i^{th} individual X_i and its dual X_i^D, in the selected subpopulation.

3.3.1 Duality-based walk

Choose either X_i or its dual X_i^D and initiate the search direction to move the individual:

$S_1 = X^{Gbest} - Y_i$, (2)

Introduce additional moving direction by:

$S_2 = X_i - X_i^D$, (3)

3.3.2 Walk toward the Child

For each couple produce their child X_i^{Ch}. It can be generated by a uniform crossover over the parents that are dual of each other. Identify an extra direction vector as:

$S_3 = X^{Child} - Y_i$, (4)

3.3.3 Selection and replacement

Generate the candidate solution and evaluate its fitness.

$X_i^{Cand} = X_i + V_i$, (5)

V_i is the velocity vector constructed by vector-sum of each walk direction after its production by a random generator. Replace the individual X_i with X_i^{Cand} if such a candidate is fitter than the current individual.

3.4 Termination or repeat the outer loop

Go back to step 2 and repeat until termination criterion is satisfied. Termination may occur when sum of velocity norms in the current active subpopulation tends to zero; that means no further movement of individuals will occur from their last positions. However, an alternate termination criterion is more common: to repeat until the iteration number reaches a prescribed value $Iter_{\max}$.

3.5 Results announcement

At this final step, announce the updated X^{Gbest} as the optimum. Fig.1 reveals pseudo-code of the proposed DS. Since X^{Gbest} is updated as a member of every population no extra memory is needed to save it.

4. ILLUSTRATIVE EXAMPLES

Some examples of unimodal and multimodal test functions (Yang, 2010) are selected for illustrative purposes. The results are treated by DS and also compared with two other meta-heuristics; *Particle Swarm Optimization,* PSO (Kennedy and Eberhart, 2001) and *Teaching-Learning-based-Optimization,* TLBO. PSO is widely used in many engineering applications and has a very simple algorithmic structure. As the second method, TLBO is selected due to its parameter-less structure (Rao, 2014). For the sake of true comparison in each case, the initial population and the number of iterations is taken similar. All the algorithms are run with the same $PopSize$ of 10 which is a relatively small number to test their capability in searching with such a few agents. However, the number of objective function evaluations is taken the base of comparison in the fitness view.

Other control parameters of PSO are taken $C_{Inertial}=1, C_{Cognitive}=C_{Social}=2$; implemented by the following relation:

$$V_i^{new}=C_{Inertial}V_i+rand\times C_{Cognitive}(X^{Pbest}-X_i)+rand\times C_{Social}(X^{Gbest}-X_i), \quad (6)$$

Applying termination criterion of DS as $\sum_i \|V\|_i>\varepsilon$ the resulted number of iterations for each example are derived and used to run PSO and TLBO, as well.

The proposed TLBO is implemented with an elitist strategy which saves the global best individual during the search by an auxiliary memory. As TLBO has no intrinsic parameters; it is run just with the same initial population and number of iterations as DS.

Regarding diversity variation as the search progresses, trace of $\sum_i \|V\|_i$ is drawn to study behavior of the treated algorithms. In addition, total CPU time on the same platform is compared between the treated methods to evaluate their computational efficiency.

Test function minimization is transformed here to a fitness maximizing problem:

$$\begin{aligned}&Maximize\ Fitness(X)=-f(X),\\&X=<x_1,...,x_n>,\ x^{LB}\le x_j\le x^{UB}\end{aligned} \quad (7)$$

Prior to fitness evaluation, each variable of the design vector is forced to fall between x^{LB}, x^{UB} as its lower and upper bounds, respectively.

$$x_i=\min(\max(x^{LB},x_i),x^{UB}), \quad (8)$$

The number of design variables n is taken 2 in the following examples for illustrative purposes.

- ***Initiation***
  - *Generate a population P of $PopSize$ individuals*
  - *Evaluate fitness*

- ***Outer Loop:*** *Repeat until termination criterion is satisfied*
  - *Extract either P1(Primary) and P2 (Dual) out of P*
  - ***Inner Loop:*** *for each X_i in the activated subpopulation*
    - *Combine parents: $X_i^{Ch} = X_i \otimes X_i^D$*
    - *Choose Y_i out of parents*
    - $X_i^{Cand.} = X_i + V_i$
    - *Evaluate fitness of $X_i^{Cand.}$*
    - *Replace X_i with fitter $X_i^{Cand.}$*
  - *Update X^{Gbest}*
- ***Announce*** *the optimum ; X^{Gbest}*

Fig.1 Pseudo-code of the proposed DS algorithm

4.1 Test problem 1: sphere function

Minimizing this benchmark is selected as an example of a convex unimodal problem. It is defined by the following relation:

$$f(X) = \sum_{j=1}^{n} x_j^{\,2}, \quad (9)$$

Whereas feasible hyper-cube is limited to $x^{LB} = -5.12,\ x^{UB} = 5.12$. It has no local optima but one global minimum obviously located at the origin: $X^* = 0$ resulting in $f^* = 0$.

Performance of DS is compared with PSO in Fig.3. According to the Pseudo-code of Fig.1, the number of fitness evaluation calls in every iteration of PSO is twice that of DS. Therefore, true comparison of convergence should be performed via function calls.

As depicted in Fig.3a, it can be realized that DS convergence is competitive to PSO using the same initial population and consequent starting elite fitness. Applying $\varepsilon = 0.1$, the proposed DS could achieve $f(X^*) = 0.000002849$ within only 56 iterations. It is while PSO obtained $f(X^*) = 0.005237058$. The reason is declared when comparing diversity trace of these methods. Fig.3b shows that despite PSO, DS has successfully reduced $\sum_i \|V_i\|$ to refine its search toward optimum in this convex example.

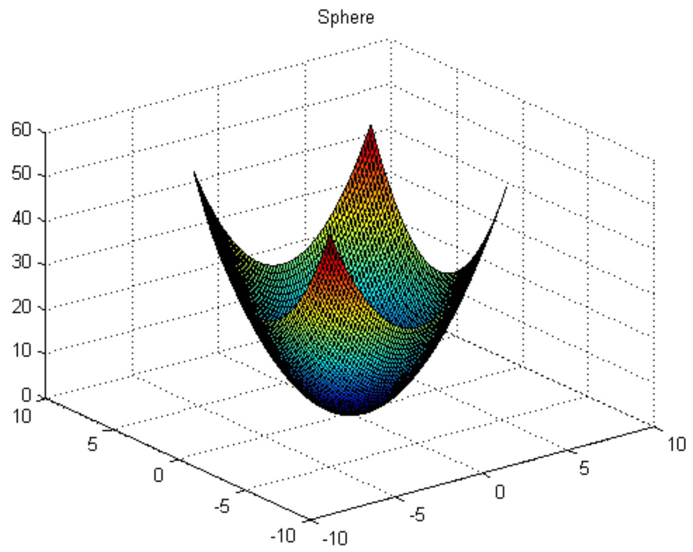
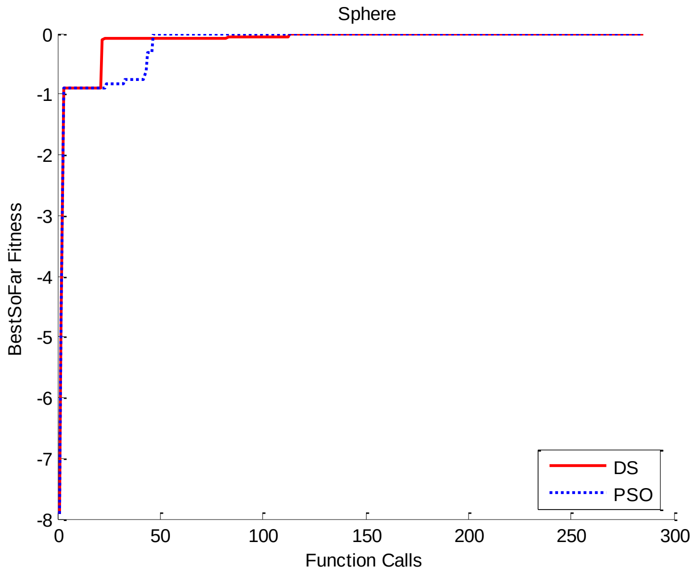


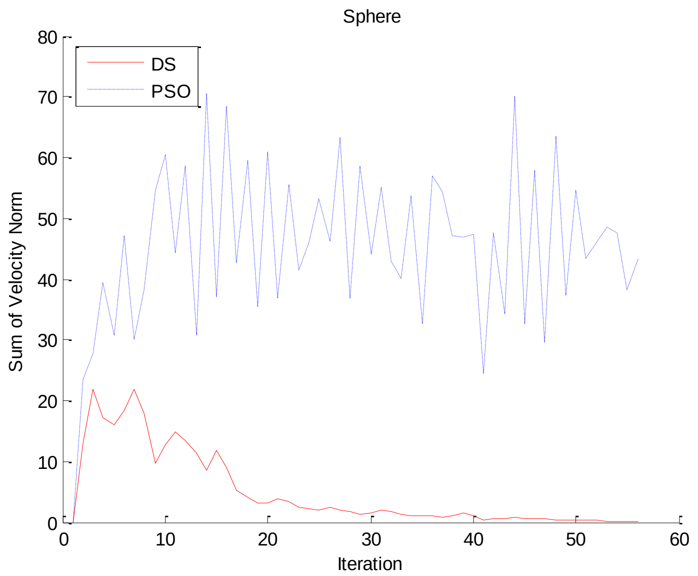
Fig.2 Sphere test function

According to Fig.4, TLBO has more rapid iteration-wise decrease in the diversity than DS; however, the proposed method has overridden its convergence curve in early function calls. In another word, DS has shown proper balance between intensification and diversification in this example. Note that TLBO applies $2 \times PopSize \times \#iterations$ fitness evaluations while this number is $PopSize \times \#iterations$ for PSO and $\frac{1}{2} \times PopSize \times \#iterations$ for DS. Therefore DS is expected to be the most efficient among these methods.

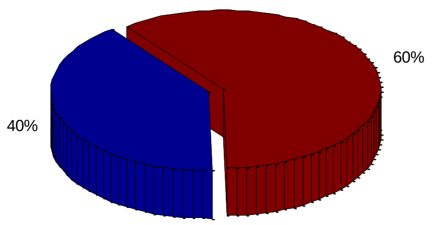
The matter is confirmed regarding elapsed time of the algorithms in every comparative run for a couple of methods. According to Fig3.c, the percentage of computational time in a single run is distributed as 40% for DS and 60% for PSO. This ratio is more sever in Fig.4c as 23% for DS vs. 77% for TLBO. While the number of function calls in the same iteration for DS is 25% of TLBO and 50% of PSO, the recent results of Fig.3c and Fig.4c, give extra information about time complexity of the algorithms.



(a)

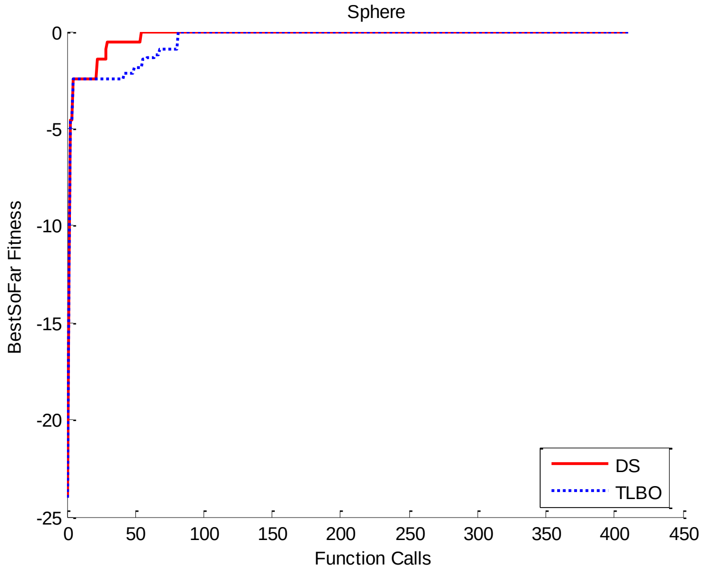


(b)

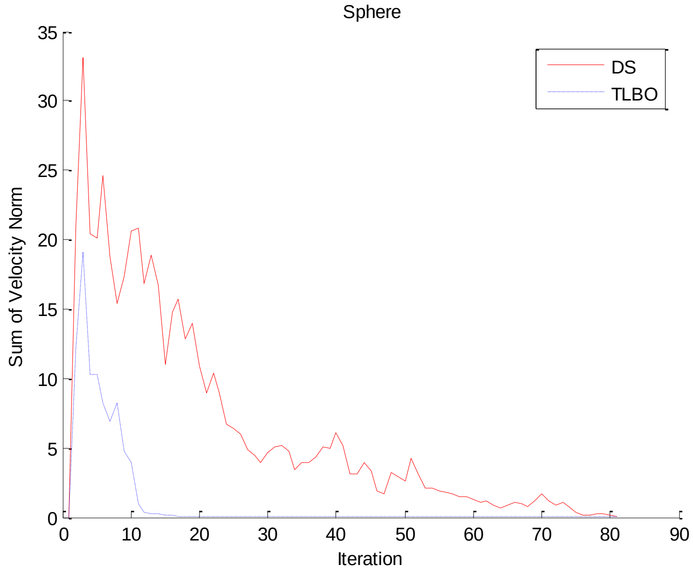


(c)

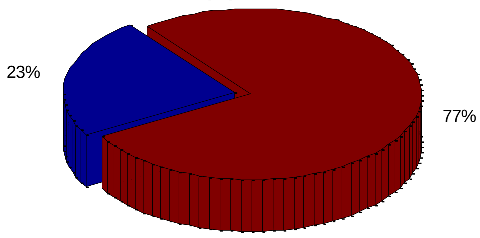
Fig.3 Comparison of DS with PSO: (a) fitness (b) diversity and (c) computational time percentage for the sphere function



(a)



(b)



(c)

Fig.4 Comparison of DS with TLBO: (a) fitness (b) diversity and (c) computational time percentage for the sphere function

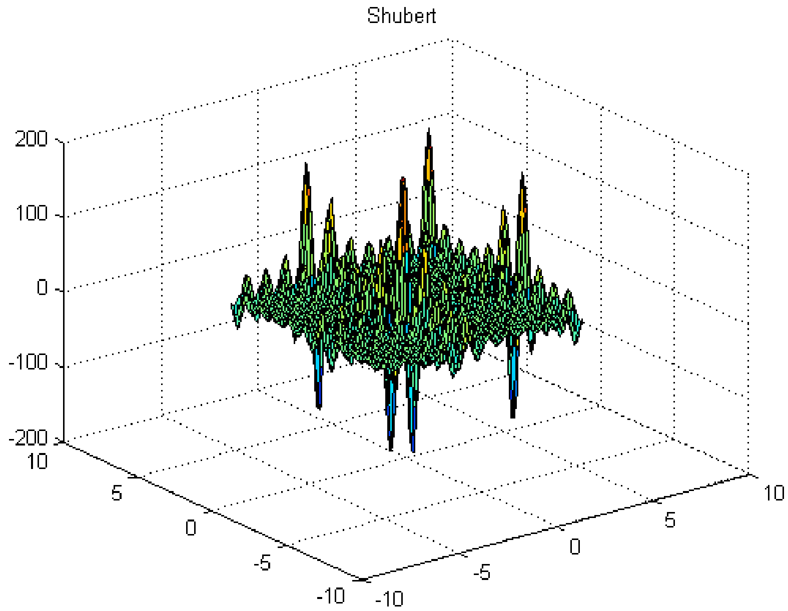


Fig.5 Shubert's test function

4.2 Test problem 2: Shubert's function

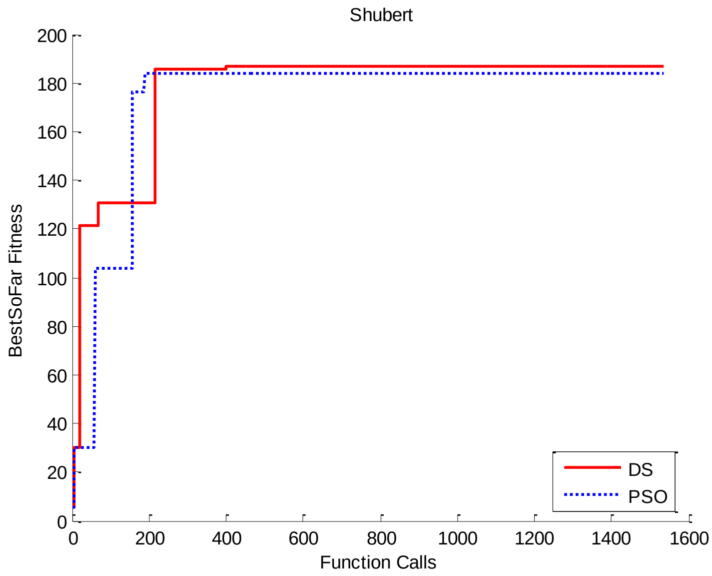
The previous tests are repeated here treating an example of a complex multimodal benchmark with several local optima and 18 global optima as depicted in Fig.5. It is defined by the following relation:

$$f(X) = \sum_{j=1}^{n} \sum_{k=1}^{5} k(\cos[(k+1)x_j \ + k)], \quad (10)$$

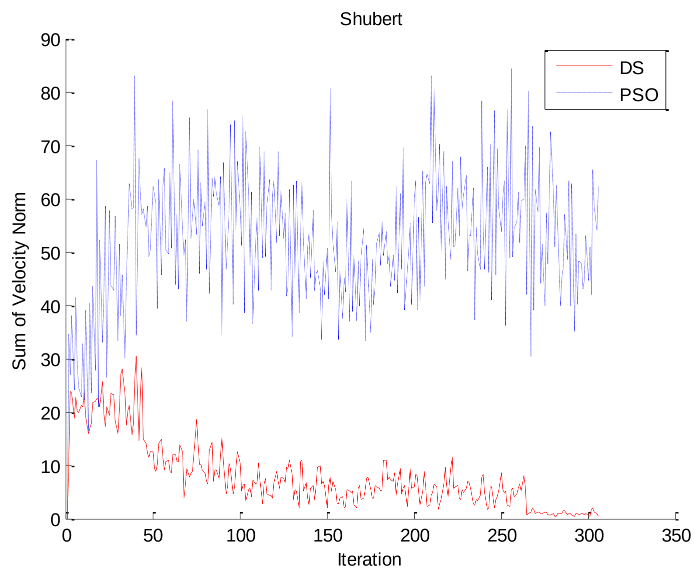
The variable limits are taken $x^{LB} = -5.12, \ x^{UB} = 5.12$. Value of the Shubert's function at its global optima is the same as $f^* = -186.7309088$ (Karimi and Siarry, 2012). Applying the same threshold of $\varepsilon = 0.1$ as the termination criterion, number of iterations is raised by DS to more than 3 times than the previous example.

Besides, according to Fig.6 and Fig.7 it is realized that trend of diversity decrease with iteration has got slower. Note that this example has several local and global optima and is more complex than the previous convex function. It is evident from Fig.6 that despite DS, the employed PSO has not been successful in diversity balance as the search progresses; because it has led to premature convergence. PSO has achieved $f = -183.9098$ which is quite different from global optimum captured by DS as $f^* = -186.73091$.

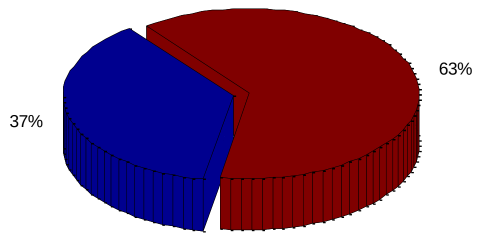
In another word, DS has better capability of escaping from local optima in such a complex design space with respect to PSO. According to Fig6.c and Fig3.c, the time complexity ratio is slightly altered with respect to previous example from 40% to 37% comparing DS over PSO time.



(a)

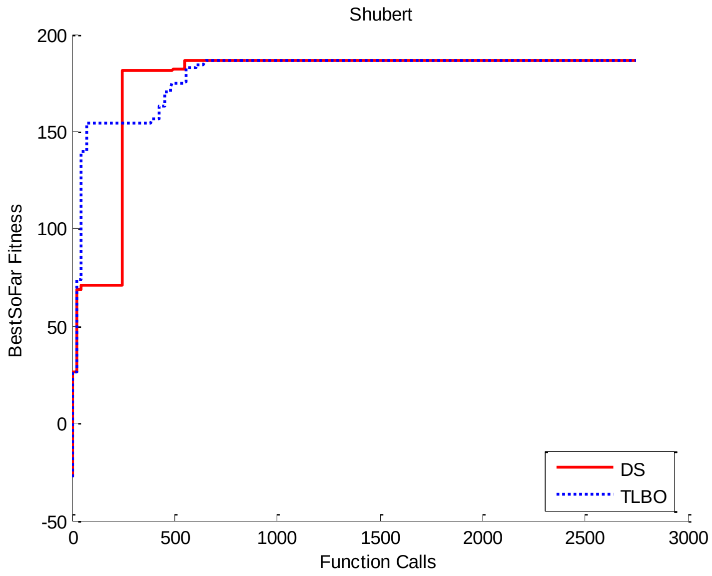


(b)

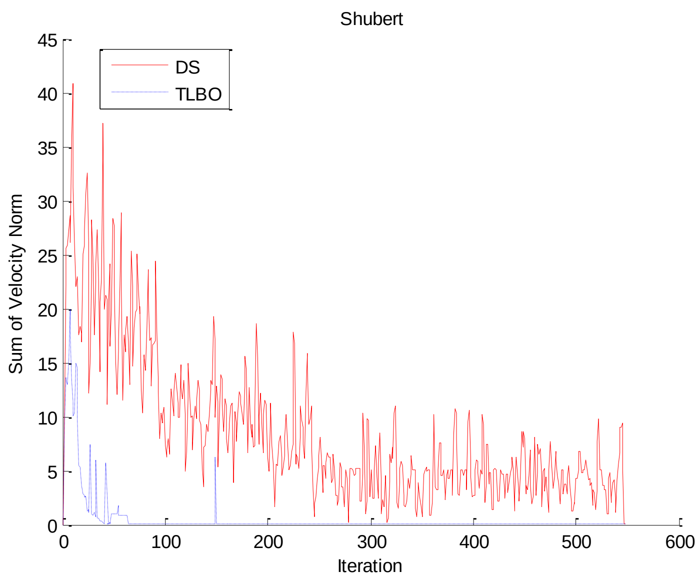


(c)

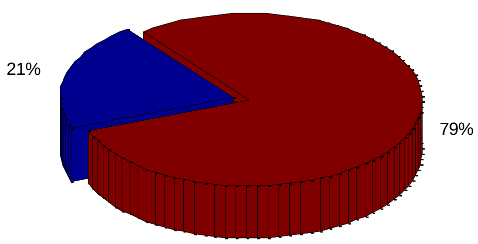
Fig.6 Comparison of DS with PSO: (a) fitness (b) diversity and (c) computational time percentage for the Shubert's function



(a)



(b)



(c)

Fig.7 Comparison of DS with TLBO: (a) fitness (b) diversity and (c) computational time percentage for the Shubert's function

Regarding Fig.7a, DS has again shown comparable convergence with TLBO in obtaining final value of $f^* = -186.73091$, in optimizing this multimodal function. DS has captured such a global optimum by 2015 function calls via 402 iterations. TLBO has achieved it within 183 iterations but by 3650 fitness evaluations. Comparison of the total elapsed time up to total 548 iterations, declares time complexity of DS is 21% of TLBO. Hence, it is concluded that DS is not only robust but also efficient in capturing global optima in such a complex multimodal search space.

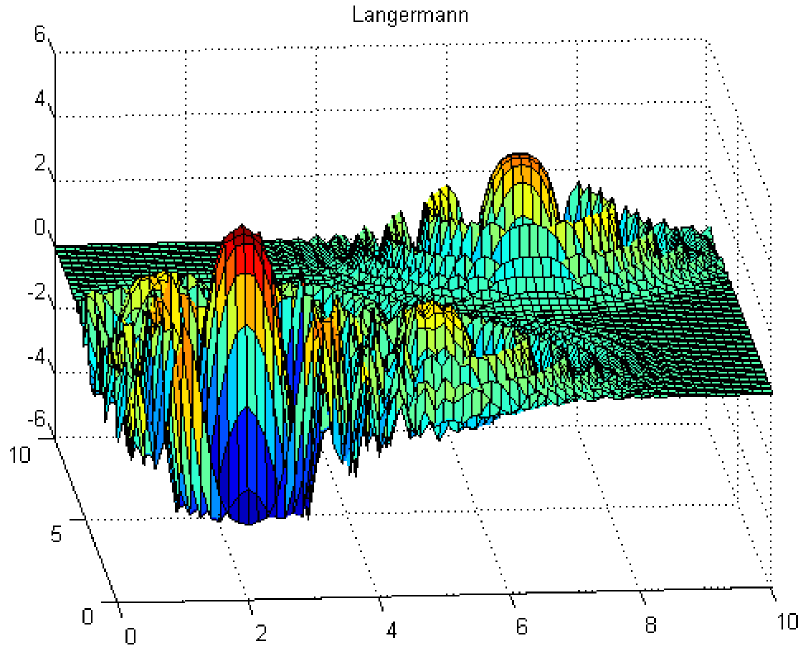


Fig.8 Langermann's test function

4.3 Test problem 3: Langermann's function

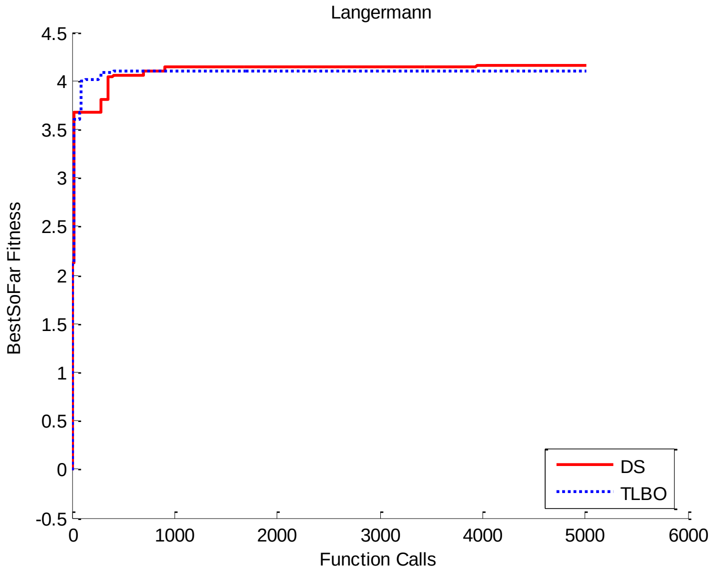
This example is a sample of a non-convex function with an asymmetric multimodal design space, defined by the following Langermann's function (Fig.8):

$$f(X) = -\sum_{j=1}^{5} c_j \sum_{i=1}^{n} \exp\left(-\frac{(x_i - a_{ij})^2}{\pi}\right) \times \cos\left(\pi (x_i - a_{ij})^2\right), \quad (11)$$

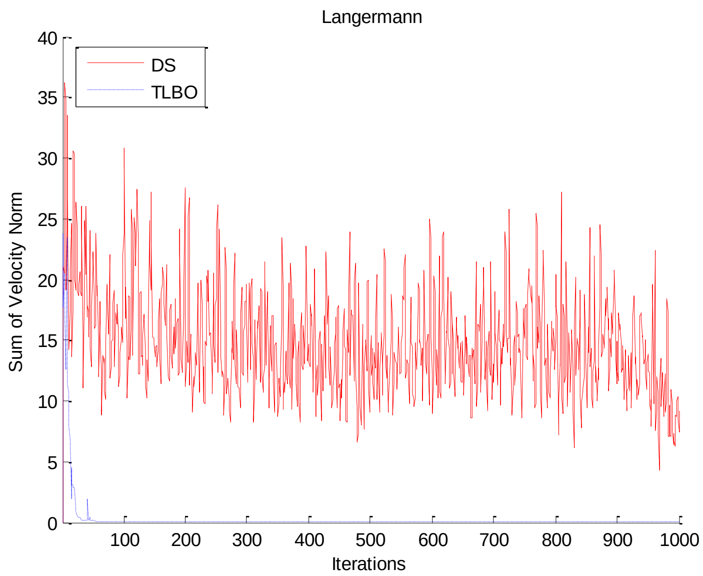
The fixed parameters are given by:

$$A = \begin{bmatrix} 3 & 5 & 2 & 1 & 7 \\ 5 & 2 & 1 & 4 & 9 \end{bmatrix}^T, \ \mathrm{C} = <1 \quad 2 \quad 5 \quad 2 \quad 3>^T, \quad (12)$$

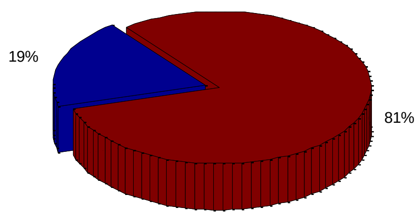
Limits on the variables are set to $x^{LB} = 0,\ x^{UB} = 10$. According to Fig.9a, DS can exhibit stable convergence toward optimum without premature convergence. The reason is behind proper robustness of DS by keeping more diversity than TLBO during early iterations of the search in such a complex problem (See Fig.9b).



(a)



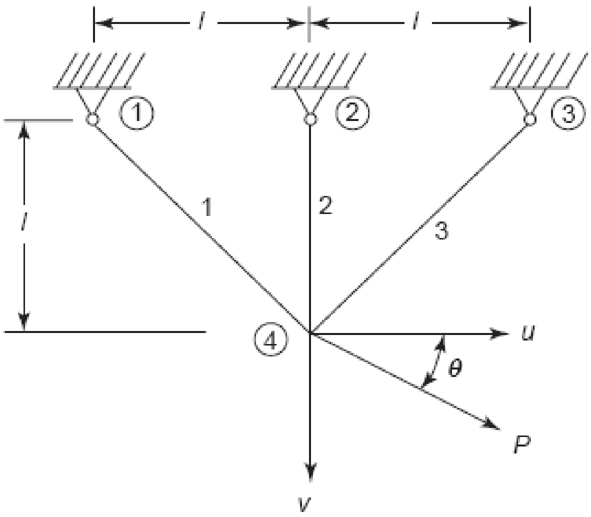
(b)



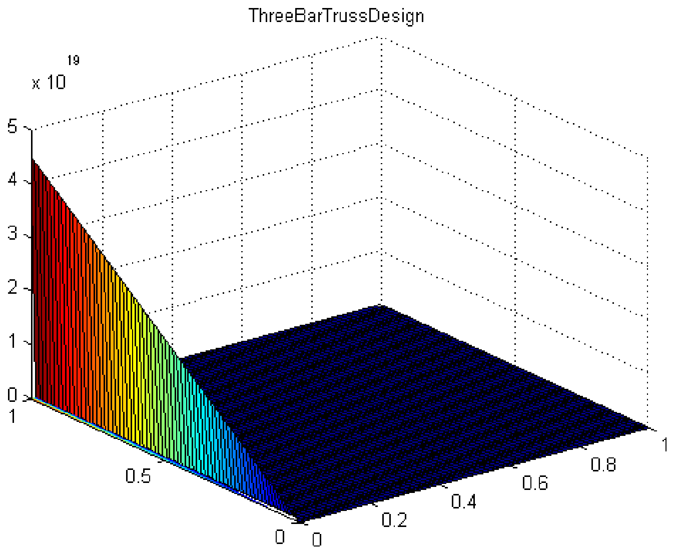
(c)

Fig.9 Comparison of DS with TLBO: (a) fitness (b) diversity and (c) computational time percentage for the Langermann's function

In this example DS could capture the optimum of $f(2.7934,1.5972) = -4.1558$ by 19% total time of TLBO which resulted in $f(1.4165, 1.7991) = -4.1009$ after 1000 iterations.



(a)



(b)

Fig.10 (a) The structural model and (b) design space of the 3-bar truss problem

4.4 Test problem 4: 3-bar truss design

This is an example of a constrained engineering problem that is chosen because its design space is illustratable (Shahrouzi and Kaveh, 2015). It has only 2 design variables $\underline{X} = (x_1, x_2)$ as section areas of the truss members, shown in Fig.10a. The objective is to minimize material consumption subjected to structural behaviour constraints where design variables are limited with in [0,1] domain. The problem is formulated as follows with fixed parameters of $l = 100cm$, $\sigma = 2kN/cm^2$ and $P = 2kN$ as the vertical load:

$$Minimize \quad f(X) = \rho l \times (2\sqrt{2x_1} + x_2), \quad (13)$$

$$Subject\ to$$

$$g_1 = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0 \quad (14)$$

$$g_2 = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0 \quad (15)$$

$$g_3 = \frac{1}{x_1 + \sqrt{2}x_2} P - \sigma \leq 0 \quad (16)$$

It is treated in the following unconstrained form. The penalty coefficient of $k_p = 50$ is employed in this study.

$$Maximize\ Fitness(X) = -f(X) \times (1 + k_p \sum_{i=1}^{3} \max(0, g_i(X))), \quad (17)$$

According to Fig.11, it is realized that TLBO have the most rapid diversity decrease. In contrary, PSO diversity fluctuates about a nearly constant higher value. DS trend is between PSO and TLBO so that it has a high diversity at early stages of optimization which gradually decreases as progressing to the end. Numerical results of DS, PSO and TLBO are compared in Table 1 for this constrained structural optimization problem. It is observed that the minimal weight result of DS is between those of PSO and TLBO with a difference of less than 1%. However, DS has obtained such a result within nearly one forth of TLBO time and half of PSO time. Average measure of velocity norm for DS in these sample runs, has been less than 20% of PSO and 1900% of TLBO among the entire iterations.

Table.1 Comparative results of PSO, TLBO and DS in 3-bar truss design

Method:	DS	PSO	TLBO
Cost function	263.91	263.92	263.90
Infeasibility	0	0	0
Elapsed time Ratio	26%	46%	100%
Mean Sum of Velocity Norm	0.5836	4.4765	0.0338

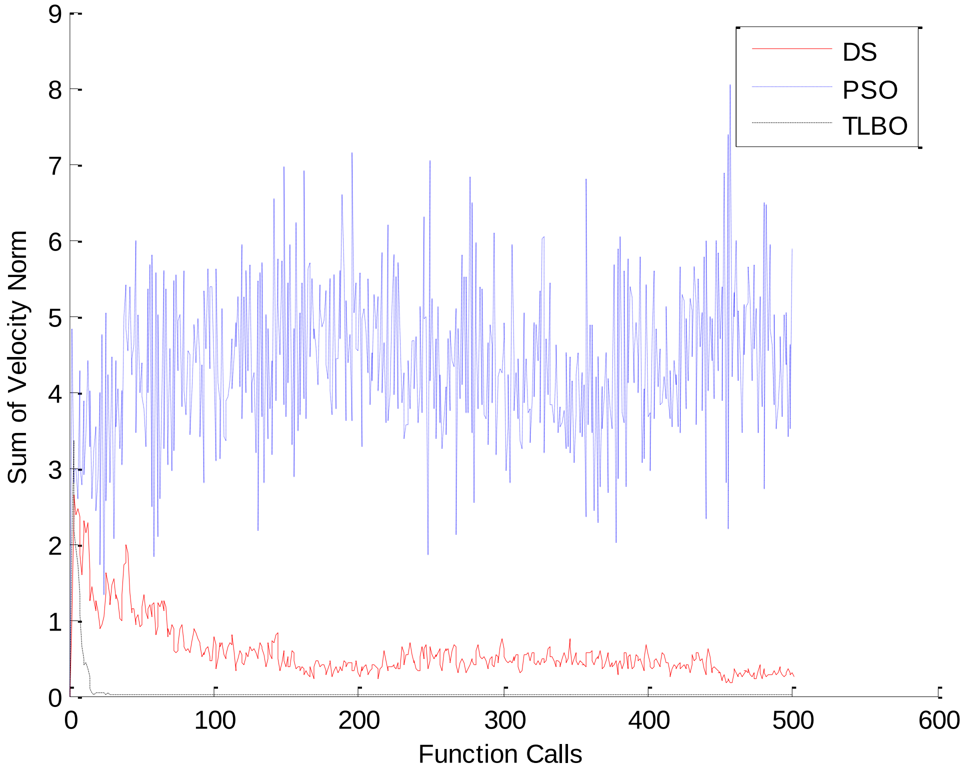


Fig.11 Diversity comparison between PSO, TLBO and DS in the 3-bar truss design

5. CONCLUSIONS

A new meta-heuristic algorithm was introduced regarding duality of search agents in population based stochastic optimization. DS applies special duality measure after dynamic sorting of individuals based on their fitness rank at every iteration. The algorithm takes benefit of both the primary individual and its dual as a couple of agents via its special walks. Since an either primary or dual part of the population is activated, DS is run with low function calls for the same number of iterations. Additional walk of the dual parents toward their child has provided an enhanced capability for the proposed algorithm.

Although that the number of fitness evaluations is reduced compared to many other algorithms, DS showed competitive performance in seeking global optima not only in simple unimodal test functions but also in complex non-convex symmetric or asymmetric design spaces.

The proposed method was successfully tested in unconstrained and constrained problems tracing its diversity in comparison with PSO as a vector-sum optimization and TLBO as a parameter-less algorithm. It was concluded that DS can robustly alter its trend of diversity variation without requiring rigorous parameter tuning as the complexity of the design space changes.

Applying a relatively small population size in addition to activating half of it at any iteration; DS is proven to be very efficient and competitive search algorithm in the treated problems. It requires considerably less computational effort and function calls to achieve similar results to other treated algorithms.

In the light of the current study, the proposed duality search can be recommended as a simple method with outstanding efficiency and robust diversity control in optimization. The parameter-less and memory-less architecture of the proposed DS has made it interesting for practical implementation in engineering problems.

REFERENCES

Abdechiri, M., Meybodi, M.R., Bahrami, H. (2013), "Gases brownian motion optimization: An algorithm for optimization (GBMO)". *Appl. Soft Comput.*, **13**(5),.2932–2946.

Akbarzadeh, A. (2016), "A novel metaheuristic method for solving constrained engineering optimization problems: Crow Search Algorithm", *Comput. Struct.,* **162,** 1-12.

Arora J.S. (2004), *Introduction to optimum design, Elsevier*, US.

Back, T. (1996), *Evolutionary Algorithms in Theory and Practice, Oxford University Press,* US.

Karimi, A., Siarry, P. (2012), "Global simplex optimization: a simple and efficient meta-heuristic for continuous optimization", *Eng. Appl. Artif. Intell.,* **25**, 48–55.

Kennedy J. and Eberhart R. (2001), *Swarm Intelligence*, *Academic Press*, UK.

Kaveh A. (2014)*, Advances in Metaheuristic Algorithms for Optimal Design of Structures,* Springer International Publishing, Switzerland.

Kaveh, A., Mahdavi, M.R.(2014), "Colliding Bodies Optimization method for optimum discrete design of truss structures", *Comput. Struct.,* **139**, 43-53.

Kaveh, A., Mahdavi, M.R.(2015), *Colliding Bodies Optimization: extension and applications,* Springer International Publishing, Switzerland.

Kaveh A, Zolghadr, A. (2016), "A novel meta-heuristic algorithm: Tug of War Optimization", *Int. J. Optim. Civil Eng.,***6(**4),469-492.

Han, L., He, X.S. (2007), "A novel opposition-based particle swarm optimization for noisy problems", *Third Int. Conf. Natural Comput*., Haikou, China, 624-629.

He, Q., Wang, L. (2007), "An effective co-evolutionary particle swarm optimization for constrained engineering design problems", *Eng. Appl. Artif. Intell*, **20**, 89–99

Mirjalili, S. (2015), "The Ant Lion Optimizer", *Adv. Eng soft*, **83**, 80-98.

Price, K., Storn, R.M., Lampinen, J.A. (2005), *Differential Evolution: A Practical Approach to Global Optimization* , *Natural Computing Series*, *Springer*; 1st Ed.

Rahmani, R., Rubiyah, Y. (2014), "A new simple, fast and efficient algorithm for global optimization over continuous search-space problems: Radial Movement Optimization", *Appl. Math. Comput.*, **248**, 287-300.

Rahnamayan, S., Tizhoosh, H.R., Salama, M.M.A. (2008), "Opposition versus Randomness in Soft Computing Techniques", *Appl. Soft Comput.*, **8**, 906-918.

Rao, RV., Savsani, VJ., Vakharia, DP. (2011), "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems", *Computer-Aided Des.***, 43**(3), 303-315.

Rao, R.V., Patel, V. (2013), "An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems", *Scientia Iranica*, **20**(3), 710-720.
Rao R.V. (2016), *Teaching Learning Based Optimization Algorithm And Its Engineering Applications*, Springer International Publishing, Switzerland.
Shahrouzi, M. (2011a), "A new hybrid genetic and swarm optimization for earthquake accelerogram scaling", *Int. J. Opt. Civil Eng.,* **1**(1), 127-140.
Shahrouzi, M. (2011b), "Pseudo-random directional search: a new heuristic for optimization", *Int. J. Opt. Civil Eng.,* **1**(2), 341-355.
Shahrouzi, M., Kaveh, A. (2015), "Dynamic Fuzzy-membership Optimization: An Enhanced Meta-heuristic Search", *Asian J. Civil Eng.,* **4**(3), 381-397.
Tizhoosh, H.R. (2005), "Opposition-Based Learning: A New Scheme for Machine Intelligence", *Int. Conf. Comput. Intell. Modelling Control Automation* (CIMCA'2005), Vienna, Austria, **I**, 695-701.
Wolpert, D.H., Macready, W.G. (2005), "Co-evolutionary free lunches", *IEEE Trans. Evolut. Comput.*, **9**(6), 721-735.
Yang X.-S. (2010), "Test problems in optimization", in: *Engineering Optimization: An Introduction with Metaheuristic Applications* (Eds Xin-She Yang), John Wiley & Sons.
Yang, X.-S. (2013), "Bat algorithm: literature review and applications", Int. J. Bio-Inspired Computation, **5** (3), 141–149.
Yang, X-S., Gandomi, A.H., Talatahari, S., Alavi, S. (2012), *Metaheuristics in water, geotechnical and transport engineering*, *Elsevier,* US.
Zheng, Y.J. (2015), "Water wave optimization: A new nature-inspired metaheuristic", *Comput. Oper. Res.,* **55**, 1-11.