Sona Global Management Review
Vol 11, Issue 2, July - Decemeber 2017

MIMICKING LEADERSHIP IN ADAPTIVE SOCIAL EVOLUTION ALGORITHM FOR MULTI-OBJECTIVE OPTIMIZATION

NITISH GHUNE
Research Scholar, Indian Institute of Technology, Roorkee, India

SUBHASH C. SHARMA
Professor, Indian Institute of Technology, Roorkee, India

In the present study, the role of social leadership in learning organization is mimicked in developing an algorithm for the optimization of processes taking place in the industry. Here, the common behavior of the leaders (or 'heroes') present in the organization (or society) is considered and simulated for developing an evolutionary algorithm that can optimize multiple objectives simultaneously. The performance of this algorithm is then tested using test problems. Test problems play an important role in evaluating the performance of any multi-objective evolutionary algorithm (MOEA). Among a number of test problems available in the literature, WFG toolkit has its distinct place. In this paper, a two-objective optimization of nine WFG test-bed problems is carried out using a recently developed algorithm based on the social evolution, namely, Adaptive Social Evolution (ASE) algorithm. ASE takes its inspiration from the common social behavior of following the heroes present in the nearby society. The results obtained are promising for minimization of WFG problems. These are compared with the existing studies in the literature and found to be better in terms of convergence, operating structure and computational time.

Keywords: leadership, learning organization, adaptive social evolution, optimization, WFG test suite 4 ICFWO 2017

INTRODUCTION

Peter Senge (Senge, 1990) popularized the term "learning organization", which describes an organization with an ideal learning environment, perfectly in tune with its goals. In such organization employees persistently explore their capacity to create the results they actually desire, foster their patterns of thinking, and follow the phenomenon of continuous learning to constantly transform and achieve the desired set of objectives. Senge emphasized the role of the *leader* in the creation of such learning organization which encourages to a more interconnected way of thinking. Thus, such organization becomes more like a community for which employees (people) feel a commitment to and work harder for its success. A leader present in such an environment originates and develops the trend. He inspires the trust among the group members.

Such process in which one can benefit themselves as well as others present nearby for further improvement of individuals in the upcoming generations can be seen analogous in terms of the social environment, which is described in section 2.

It has been seen that decision-making in an organization involves more than one contradictory factors for fulfilling different objectives, which neither can be compared nor can be combined with each other. Until the 1970s, engineering processes mainly aspired to find the most efficient solution at least cost; and were considered as single criterion problems. Beyond 1990, the multi-criterion approaches led to a "paradigm shift" in decision-making as it became easy to solve the problems with multiple non-commensurate objectives. Such problems are termed as *multi-objective optimization* (MOO) problems. These require a search for the "*best*" configuration of a set of decision variables to attain desired goals or objectives.

MOO problem (MOOP) formulation processes became popular in the 90s with the advent of several meta-heuristic algorithms viz. genetic algorithm (GA) (Holland, 1975), simulated annealing (SA) (Kirkpatrick, Gelatt, & Vecchi, 1983), etc. These algorithms are primarily based on the themes from *nature*, *biological evolution*, a phenomenon based on *laws of physics* and *social behavior*. Among these, the *evolutionary algorithms* mimic the concept of natural biology and include the techniques such as GA which simulates Darwin's evolution principle in the context of genetics, differential evolution (DE) (Storn & Price, 1997) which simulates the perturbation of chromosomes with scaled differences, etc. Similarly the algorithms such as SA, gravitational search algorithm (GSA) (Rashedi, Nezamabadi-pour, & Saryazdi, 2009), etc. take the inspiration from the *physical laws* whereas, the *swarm-intelligence based algorithms* take inspiration from the species showing collective behavior, such as birds, ants, wolf, etc. These are described next.

The various prominent representatives of swarm-based algorithms are particle swarm optimization (PSO) (Kennedy & Eberhart, 1995) which simulates the *movement of particles* those numerically directed by best-known positions of their own and that of the entire population present in the search-space, ant colony optimization (ACO) (Dorigo, Maniezzo, & Colorni, 1996) which simulates the *cooperative food searching strategy* of ants, grey wolf optimizer (GWO) (Mirjalili, Mirjalili, & Lewis, 2014) which imitates the *leadership style and hunting behavior* of grey wolves present in nature, etc. All these algorithms have a common trait of mimicking the interesting features of different animals in the nature for their survival and food source. Further, the swarm-based algorithms have also tangled with the concepts from *social behavior and evolution*. Social spider optimization (SSO) algorithm (Cuevas, Cienfuegos, Zaldívar, & Pérez-cisneros, 2013) mimics the *co-operative hunting strategy* of spiders.

In this, the spiders locate the positions of prey by analyzing the vibrations on the web. Social learning particle swarm optimization (SL-PSO) (Cheng & Jin, 2015) incorporates *social learning phenomenon* into PSO, where every particle learns from any other better particle(s) present in the swarm. Social learning optimization (SLO) (Liu, Chu, Song, Xue, & Lu, 2016) mimics the *evolution process of human intelligence*.

Several algorithms and their variants were extensively applied to real-life industrial problems and were reviewed by many researchers (Babu & Munawar, 2007; G.P. Rangaiah, 2009; Valadi & Siarry, 2014) in their study. Despite their phenomenal success, there still exist few challenges like slower and premature convergence, parameter tuning, complex algorithmic structure, which need to be addressed.

Objective Behind the Study

From the above literature review, it is clear that several search strategies, social learning and feature adaptation strategies are reported in the literature to improve the performance of algorithms for optimization. However, these improved variants were often associated with increased number of operations and parameters (features) requiring a cumbersome parameter tuning before application. To overcome these shortcomings, there was a need of developing such an algorithm which is *simple in structure*, *prevents premature convergence* while retaining the *faster speed of convergence*. Moreover, the need of such algorithm becomes more obvious particularly for solving industrial MOOPs which often involve complicated problem equations requiring inordinately large computational time for each function calculation. Relatively less amount of attention is given on this in the literature. Hence, in order to cater these requirements and to address aforesaid shortcomings of existing algorithms, a novel real-coded hybrid algorithm is developed in the present work, namely Adaptive Social Evolution (ASE) which is based on "*social evolution by following heroes*" and utilizes the background knowledge of existing algorithms like GA and PSO.

In this chapter, the motivation used in developing the ASE algorithm is described in section 2, developed algorithm is described in section 3, results and discussion by comparing the obtained results with that of real-coded version of non-dominated sorting PSO (NSPSO) (Sedighizadeh, Faramarzi, Mahmoodi, & Sarvi, 2014) for 9 benchmark optimization problem from two objective WFG test suite (Huband, Hingston, Barone, & While, 2006) is described in section 4 followed by summary in section 5. ASE has been previously applied to problems from DTLZ test suite (Ghune, Trivedi, & Ramteke, 2014) and also to industrial processes (Ramteke, Trivedi, & Ghune, 2013).

MOTIVATION FROM SOCIAL EVOLUTION

The phenomenon of social evolution has some unique features as compared to the natural biological evolution process, e.g. society indulges in transformation willingly and purposefully depending upon the circumstances and can transform completely even more than once by inheriting the acquired characteristics (Grinin, Korotayev, & Markov, 2011). These properties make social evolution proceed faster than biological evolution. The individuals are the basic unit of a society; they adapt and improve through mutual interactions. Some of these individuals (within the society) have such a quality that enables them to influence a larger section of individuals in the society. Such individuals are termed as 'heroes' or leaders. Many individuals improve themselves by inheriting the characteristics from these heroes. Also, multiple heroes can be present in a society at the same time. So, there is a substantial chance of heroes being improved through the interactions with the other heroes present in the society.

This behavioral pattern resembles the progress of the population-based optimization algorithms in which the major changes occur in initial populations whereas the populations in later generations become more and more stable, finally achieving the convergence for optimality. This analogy suggests that the concept of social evolution can be utilized to improve the convergence speed of an algorithm at least in the initial generations and has a direct relevance in solving the complex industrial problems. This is accomplished in ASE using the operators, namely, following heroes (FH) and personalized deviation (PD).

Let us consider an example of a great Indian leader turned politician, M. K. Gandhi whose ideology of truth, equality, and non-violence inspired the common men present in the society at that time. The characteristics of this hero further influenced other great leaders in the context of African and American society; namely, Nelson Mandela and Martin Luther King Jr. respectively. These leaders were the source of inspiration for the common men in the society and inherited the characteristics from M. K. Gandhi. This is a continuous process where the leaders keep on shaping the future generations. A similar analogy is taken into consideration for the development of FH operator, where multiple heroes are identified which inspire each other as well as the rest of the individuals. Thus, the overall population improves over the generations and the individuals with better characteristics become heroes in the future generations.

As per the general observation, it has been found that in addition to the social behavior of being inspired by others or heroes, there also prevails a self-induced change in one's personality which may lead to the transformation of that individual. Considering the same illustration to show such transformation, where M. K. Gandhi who used to be a well-dressed young lawyer, later became an austerely dressed freedom fighter. Martin Luther King Jr. who was awarded a doctorate in theology at a young age later became the leader of Civil Rights Movement for combating racial inequality through non-violence in America. Nelson Mandela started his childhood by living a tribal life, grew up by facing racism and later became the President of South Africa. Such facets are mimicked in PD operator in which the selected characteristics of the individuals are perturbed randomly.

Though the impact of heroes in the society is explored by several researchers in detail (Bligh & Robinson, 2010; Kurtz, 2010) in the context of sociology, the present study incorporates the simplest interpretation of the aforementioned concept. The implementation of this theme to constitute an optimization algorithm is described next.

ASE ALGORITHM

ASE algorithm proceeds in the following order of steps: *initialization*, *fitness* calculation, *ranking* (Deb, 2001), *following heroes* (FH), *personalized deviation* (PD) and *elitism* (Deb, 2001). The flowchart of ASE is shown in Fig. 1. The algorithm initiates with the random generation of a population consisting N_p solutions (individuals). Each individual is an array of decision variables in which a variable resembles a specific characteristic of that individual. These individuals are generated by randomly generating the characteristics (variables) within the predefined lower-upper bounds as follows:

$$X_{i,j} = X_{i,j}^{Low} + RN \times (X_{i,j}^{High} - X_{i,j}^{Low}) \quad \ldots (1)$$

Here, $X_{i,j}$ is j^{th} characteristic (variables) of the i^{th} individual (solution) and RN is the random number $\in [0,1]$. This process is repeated for all N_p individuals.

Each individual is then ranked according to its goodness. The goodness is represented through the fitness function (ff_i) and is calculated based on characteristics of an individual. Next, the individuals with rank =1 are designated as *heroes* (see Box H in Fig. 1). All individuals in the population are sequentially inspired from randomly selected heroes in the FH operator with a probability P_{FH} to improve their characteristics (see Box P''). This operator perturbs the original value of a given characteristic with that of a randomly selected *hero*. In this way, all the characteristics of an individual are perturbed to the inspired characteristics using that of different heroes. All N_p members of the population undergo this inspiration process one after another. In this, j^{th} characteristic of an i^{th} individual is inspired as follows:

$$X_{i,j}^{New} = X_{i,j}^{Old} + RN \times (X_{h,j}^{Hero} - X_{i,j}^{Old}) \quad \ldots (2)$$

Here, $X_{h,j}^{Hero}$ is j^{th} characteristic of randomly selected h^{th} hero.

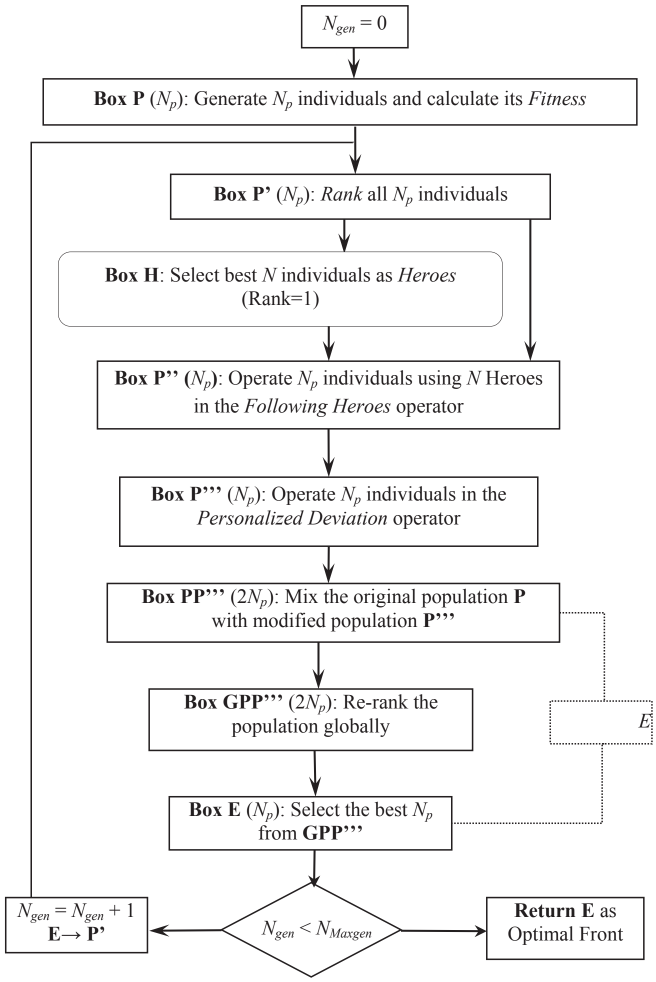

Fig. 1: Flowchart of ASE by Following Heroes.

Despite perturbation occurring in FH operation, the simulated social structure needs to be perturbed more rigorously for the further consistent emergence of better heroes having higher fitness in each generation and inhibiting the trapping in the local optima. This is enhanced by incorporating the concept of random perturbation from artificial intelligence. This is achieved in *personalized deviation* (PD) operation in which the characteristics (variables) of the inspired individuals from FH operation are perturbed randomly with a probability P_{PD} (see Box P'''). This is carried out on a selected individual by replacing its characteristic (variable) values by freshly generated random values between their numerical bounds as follows:

$$X_{i,j}^{New} = X_{i,j}^{Low} + RN \times (X_{i,j}^{High} - X_{i,j}^{Low}) \quad \text{...(3)}$$

This modified (improved) population after PD operation may also lead to some bad solutions, which are 'screened' out in *Elitism* operation; originally developed for GA. In this step, the modified population (N_p) is mixed with the original initial population (N_p) (see Box PP'''). These $2N_p$ individuals are re-ranked (see Box GPP''') and the best N_p individuals from this mixed population are selected as *elites* (see Box E). These N_p best solutions then form the starting population for the following generation and the process is iterated till the algorithm converges to the optimal solutions, or till the completion of a maximum (specified) number of generations. The developed algorithm has only three parameters i.e. population size (N_p), probabilities of FH (P_{FH}) and PD (P_{PD}) operations.

RESULTS AND DISCUSSION

The efficacy of ASE algorithm is analyzed quantitatively by solving nine test problems from WFG test suite (WFG1-WFG9), each scaled for 2 objectives (2D) (Huband, Hingston, Barone, & While, 2006). The test or benchmark problems act as a test bed for evaluating the performance of any optimization algorithm by analyzing the ability in handling the constraints and convergence capability to achieve the near optimal solutions. They also represent the complexities associated with various real-world based problems.

The performance of ASE is then evaluated by comparing the results with NSPSO on the basis of two well-known metrics i.e. generational distance (GD) (Van Veldhuizen & Lamont, 1998) and spacing (S) (Schott, 1995). The lower value of GD and S approaching ≈ 0 indicates the better convergence and evenly spaced distribution respectively. The parameters used in the compared algorithm: NSPSO and ASE are given in Table 1 and are selected as best values reported in the literature (Ghune et al., 2014; Sedighizadeh et al., 2014).

ASE is first applied to 9 test problems from WFG test suites. Formulation and characteristics of WFG problems are given in Annexure Table A.1 and A.2. Each of these problems is executed for 50 different random numbers and the average values (AV) of GD and S with their standard deviations (SD) are reported. These results are then compared to that obtained using NSPSO as given in Tables 2-4 in the format of AV±SD.

Table 2 shows the comparison of GD metric values of ASE and NSPSO for 50, 100, 200, and 500 generations. In these results, ASE is found to be having better values for 7, 6, 6 and 5 problems for 50, 100, 200 and 500 generations, respectively in comparison with NSPSO (marked 'bold'). The results in the smaller number of generations show better convergence using ASE than using NSPSO and thus establish its usefulness for the real-life industrial problems where the numbers of generations are severely restricted (Kasat & Gupta, 2003; Masuduzzaman & Rangaiah, 2009). The results also indicate that the number of problems where ASE is superior to NSPSO decreases with the increase in a number of generations as both algorithm approaches to global optimality.

Further, in Table 3 the comparison of S metric values of ASE and NSPSO is reported for 500 generations as the results are only compared after the maximum convergence for most of the problems. The result clearly shows that ASE has a better spacing for all 9 problems compared to NSPSO and confirms its superiority.

Also, the performance of ASE algorithm is further analyzed in terms of CPU time in Table 4. Here, the CPU time (secs) required by NSPSO and ASE to execute 50 different random runs of all nine problems for 500 generations on a workstation (Intel Xeon (R) E5 – 2640 @ 2.50 GHz processor, 16 GB RAM, and Windows 10 OS) is compared. The comparison shows that the CPU time of ASE is lower for all problems. This achieves the objective of the *faster speed of convergence* along with the *simpler structure* of the developed algorithm.

Table 1: Parameters used for NSPSO and ASE.

NSPSO		ASE	
N_P	100	N_P	100
w	0.4 - 0.9	P_{FH}	0.9
c_1	2.0	P_{PD}	0.1
c_2	2.2		

Table 2: Comparison of GD metric for NSPSO and ASE for varying number of generations.

WFG-2D	50 Generations		100 Generations		200 Generations		500 Generations	
	NSPSO	ASE	NSPSO	ASE	NSPSO	ASE	NSPSO	ASE
WFG1	0.1242 ± 0.0005	0.1220 ± 0.0008	0.1225 ± 0.0004	0.1185 ± 0.0009	0.1201 ± 0.0006	0.1147 ± 0.0013	0.1150 ± 0.0007	0.1091 ± 0.0026
WFG2	0.0116 ± 0.0017	0.0943 ± 0.0104	0.0064 ± 0.0013	0.0982 ± 0.0095	0.0029 ± 0.0005	0.0984 ± 0.0082	0.0017 ± 0.0002	0.0917 ± 0.0102
WFG3	0.0102 ± 0.0013	0.0056 ± 0.0010	0.0052 ± 0.0010	0.0044 ± 0.0008	0.0039 ± 0.0007	0.0025 ± 0.0004	0.0036 ± 0.0004	0.0013 ± 0.0001
WFG4	0.0119 ± 0.0005	0.0114 ± 0.0004	0.0100 ± 0.0004	0.0096 ± 0.0004	0.0083 ± 0.0006	0.0080 ± 0.0005	0.0040 ± 0.0002	0.0042 ± 0.0004
WFG5	0.0269 ± 0.0077	0.0507 ± 0.0026	0.0083 ± 0.0014	0.0420 ± 0.0030	0.0066 ± 0.0002	0.0313 ± 0.0025	0.0062 ± 0.0000	0.0206 ± 0.0013
WFG6	0.0205 ± 0.0014	0.0144 ± 0.0025	0.0152 ± 0.0013	0.0080 ± 0.0014	0.0095 ± 0.0016	0.0055 ± 0.0008	0.0045 ± 0.0009	0.0044 ± 0.0005
WFG7	0.0232 ± 0.0009	0.0227 ± 0.0013	0.0183 ± 0.0007	0.0215 ± 0.0018	0.0045 ± 0.0005	0.0179 ± 0.0015	0.0017 ± 0.0002	0.0130 ± 0.0015

WFG8	0.0304 ± 0.0016	0.0283 ± 0.0022	0.0257 ± 0.0010	0.0253 ± 0.0017	0.0224 ± 0.0008	0.0222 ± 0.0003	0.0186 ± 0.0005	0.0185 ± 0.0004
WFG9	0.0111 ± 0.0016	0.0098 ± 0.0023	0.0084 ± 0.0016	0.0078 ± 0.0020	0.0069 ± 0.0023	0.0062 ± 0.0018	0.0049 ± 0.0012	0.0042 ± 0.0016

Table 3: Comparison of S metric for NSPSO and ASE over 500 generations.

WFG-2D	NSPSO	ASE
WFG1	0.0239 ± 0.0025	0.0058 ± 0.0021
WFG2	0.0183 ± 0.0031	0.0155 ± 0.0042
WFG3	0.0280 ± 0.0024	0.0200 ± 0.0029
WFG4	0.0339 ± 0.0042	0.0222 ± 0.0054
WFG5	0.0337 ± 0.0027	0.0221 ± 0.0061
WFG6	0.0358 ± 0.0042	0.0230 ± 0.0046
WFG7	0.0377 ± 0.0035	0.0225 ± 0.0034
WFG8	0.0341 ± 0.0034	0.0305 ± 0.0109
WFG9	0.0353 ± 0.0038	0.0292 ± 0.0108

Table 4: Comparison of CPU time for NSPSO and ASE over 500 generations.

WFG-2D	NSPSO	ASE
WFG1	161.64	104.50
WFG2	153.17	103.64
WFG3	207.82	105.78
WFG4	208.23	105.52
WFG5	210.39	106.16
WFG6	199.37	96.30
WFG7	183.04	95.50
WFG8	180.25	102.34
WFG9	156.25	118.89

ANNEXURE

Table A.1: General formulations and details of WFG problems (Huband et al., 2006)

Given,

$$Z=\{z_1,\ldots,z_k,z_{k+1},\ldots,z_n\}$$

$$Z_{[0,1]}=\{z_{1,[0,1]},\ldots,z_{n,[0,1]}\}=\{z_1/z_{1,\max},\ldots,z_n/z_{n,\max}\}$$

$Z_{[0,1]}\Rightarrow t^1\Rightarrow\ldots\Rightarrow t^{p-1}\Rightarrow t^p=\{t_1^p,\ldots,t_M^p\}$; where t^i is the i^{th} transformation

$$X=\{x_1,\ldots,x_M\}=\left\{\max\left(t_M^p,A_1\right)\left(t_1^p-0.5\right)+0.5,\ldots,\max\left(t_M^p,A_{M-1}\right)\left(t_{M-1}^p-0.5\right)+0.5,t_M^p\right\}$$

Minimize

$$OF_{m=1:M}\left(X\right)=Dx_M+S_m h_m\left(x_1,\ldots,x_{M-1}\right)$$

Where *Z, t, A, X, D, S and h* are Working Parameter, Transition Parameter, Degeneracy Constant, Position & Distance Vector, Distance Scaling Constant, Scaling Constant and Shape Function respectively. Here, M = 2 for *two objective* formulation and n = 24.

$$S_{m=1:M}=2m;D=1;A_1=1,A_{2:M-1}=\begin{cases}1, & otherwise\\ 0, & for\ WFG3\end{cases}$$

$$z_{i=1:n,\max}=2i$$

SHAPE FUNCTIONS

Linear, Convex, Concave, Mixed (convex/concave), Disconnected

TRANSFORMATION FUNCTIONS

Bias: Polynomial, Flat Region, Parameter Dependent

Shift: Linear, Deceptive, Multi-modal

Reduction: Weighted Sum, Non-separable

Note: Refer (Huband et al., 2006) for detailed shape and transformation functions

Table A.2: Mathematical formulations of WFG problems and their characteristics (Huband et al., 2006).

Name of Problem	Formulation	Parameter Domain	Characteristics				
			Geometry	Separability	Modality	Bias	Many-to-one Mapping
WFG1	$h_{m=1:M-1}=convex_m$ $h_M=mixed_M\ (with\ \alpha=1\ and\ A=5)$ $t_{i=1:k}^1=y_i$ $t_{i=k+1:n}^1=s\_linear(y_i,0.35)$ $t_{i=1:k}^2=y_i$ $t_{i=k+1:n}^2=b\_flat(y_i,0.8,0.75,0.85)$ $t_{i=1:n}^3=b\_poly(y_i,0.02)$ $t_{i=1:M-1}^4=r\_sum\left(\begin{matrix}\{y_{(i-1)k/(M-1)+1},\ldots,y_{ik/(M-1)}\},\\ \{2(i-1)k/(M-1)+1,\ldots,2ik/(M-1)\}\end{matrix}\right)$ $t_M^4=r\_sum\left(\{y_{k+1},\ldots,y_n\},\{2(k+1),\ldots,2n\}\right)$ *Note:* Refer Table A.1 for specific information about general formulation, shape, and transformation functions for all WFG problems.	[0,2*i*]	Convex, Mixed	Separable	Unimodal	Polynomial, Flat	-

WFG2	$h_{m=1:M-1} = convex_m$ $h_M = disc_M$ $(with\ \alpha=\beta=1 and\ A=5)$ $As\ t^1\ from WFG1(linear\ shift)$ $t^2_{i=1:k} = y_i$ $t^2_{i=k+1:k+l/2} = r\_nonsep\left(\left\{y_{k+2(i-k)-1}, y_{k+2(i-k)}\right\}, 2\right)$ $t^3_{i=1:M-1} = r\_sum\left(\left\{y_{(i-1)k/(M-1)+1}, \ldots, y_{ik/(M-1)}\right\}, \{1,\ldots,1\}\right)$ $t^3_M = r\_sum\left(\left\{y_{k+1}, \ldots, y_{k+l/2}\right\}, \{1,\ldots,1\}\right)$	[0,2i]	Convex, Disconnected	Non-Separable	Unimodal except OF_m	No	-
WFG3	$h_{m=1:M} = linear_m\,(degenerate)$ $As\ t^{1:3}\ from WFG2(linear\ shift, non\text{-}separable,$ $weighted\ sum\ reduction)$	[0,2i]	Linear, Degenerate	Non-Separable	Unimodal	No	-
WFG4	$h_{m=1:M} = concave_m$ $t^1_{i=1:n} = s\_multi(y_i, 30, 10, 0.35)$ $t^2_{i=1:M-1} = r\_sum\left(\left\{y_{(i-1)k/(M-1)+1}, \ldots, y_{ik/(M-1)}\right\}, \{1,\ldots,1\}\right)$ $t^2_M = r\_sum\left(\left\{y_{k+1}, \ldots, y_{k+l/2}\right\}, \{1,\ldots,1\}\right)$	[0,2i]	Concave	Separable	Multimodal	No	-
WFG5	$h_{m=1:M} = concave_m$ $t^1_{i=1:n} = s\_decept(y_i, 0.35, 0.001, 0.05)$ $As\ t^2\ from WFG4(weighted\ sum\ reduction)$	[0,2i]	Concave	Separable	Deceptive	No	-
WFG6	$h_{m=1:M} = concave_m$ $As\ t^1\ from WFG1(linear\ shift)$ $t^2_{i=1:M-1} = r\_nonsep\left(\left\{y_{(i-1)k/(M-1)+1}, \ldots, y_{ik/(M-1)}\right\}, k/(M-1)\right)$ $t^2_M = r\_nonsep\left(\left\{y_{k+1}, \ldots, y_n\right\}, l\right)$	[0,2i]	Concave	Non-Separable	Unimodal	No	-
WFG7	$h_{m=1:M} = concave_m$ $t^1_{i=1:k} = b\_param\left(y_i, r\_sum\left(\left\{y_{i+1}, \ldots, y_n\right\}, \{1,\ldots,1\}\right), \frac{0.98}{49.98}, 0.02, 50\right)$ $t^1_{i=k+1:n} = y_i$ $As\ t^1\ from WFG1(linear\ shift)$ $As\ t^2\ from WFG4(weighted\ sum\ reduction)$	[0,2i]	Concave	Separable	Unimodal	Parameter Dependent	-
WFG8	$h_{m=1:M} = concave_m$ $t^1_{i=1:k} = y_i$ $t^1_{i=k+1:n} = b\_param\left(y_i, r\_sum\left(\left\{y_1, \ldots, y_{i-1}\right\}, \{1,\ldots,1\}\right), \frac{0.98}{49.98}, 0.02, 50\right)$ $As\ t^1\ from WFG1(linear\ shift)$ $As\ t^2\ from WFG4(weighted\ sum\ reduction)$	[0,2i]	Concave	Non-Separable	Unimodal	Parameter Dependent	-
WFG9	$h_{m=1:M} = concave_m$ $t^1_{i=1:n-1} = b\_param\left(y_i, r\_sum\left(\left\{y_{i+1}, \ldots, y_n\right\}, \{1,\ldots,1\}\right), \frac{0.98}{49.98}, 0.02, 50\right)$ $t^1_n = y_n$ $t^2_{i=1:k} = s\_decept(y_i, 0.35, 0.001, 0.05)$ $t^2_{k+1:n} = s\_multi(y_i, 30, 95, 0.35)$ $As\ t^2\ from WFG6(non-separable\ \ reduction)$	[0,2i]	Concave	Non-Separable	Multimodal, Deceptive	Parameter Dependent	-

REFERENCES

Babu, B. V, & Munawar, S. A. (2007). Differential evolution strategies for optimal design of shell-and-tube heat exchangers. ***Chemical Engineering Science***, 62(14), 3720–3739.

Bligh, M. C., & Robinson, J. L. (2010). Was Gandhi "charismatic"? Exploring the rhetorical leadership of Mahatma Gandhi. ***Leadership Quarterly***, 21(5), 844–855. https://doi.org/10.1016/j.leaqua.2010.07.011

Cheng, R., & Jin, Y. (2015). A social learning particle swarm optimization algorithm for scalable optimization. ***Information Sciences***, 291, 43–60.

Cuevas, E., Cienfuegos, M., Zaldívar, D., & Pérez-cisneros, M. (2013). Expert Systems with Applications A swarm optimization algorithm inspired in the behavior of the social-spider. ***Expert Systems with Applications***, 40(16), 6374–6384. https://doi.org/10.1016/j.eswa.2013.05.041

Deb, K. (2001). ***Multi-Objective Optimization Using Evolutionary Algorithms***. John Wiley & sons, LTD. Chichester, UK: Wiley. https://doi.org/10.1109/TEVC.2002.804322

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: Optimization by a colony of cooperating agents. ***IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics***, 26(1), 29–41. https://doi.org/10.1109/3477.484436

Ghune, N., Trivedi, V., & Ramteke, M. (2014). Comparison of Adaptive Social Evolution and Genetic Algorithm for Multi-objective DTLZ Toolkit. ***In Advances in Intelligent Systems and Computing*** (Vol. 259, pp. 565–576). https://doi.org/10.1007/978-81-322-1768-8_50

Grinin, L. E., Korotayev, A. V, & Markov, A. V. (2011). Biological and Social Phases of Big History : ***Similarities and Differences of Evolutionary Principles and Mechanisms***, 158–199.

Holland, J. H. (1975). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. U Michigan Press.

Huband, S., Hingston, P., Barone, L., & While, L. (2006). A review of multiobjective test problems and a scalable test problem toolkit. ***IEEE Transactions on Evolutionary Computation***, 10(5), 477–506. https://doi.org/10.1109/TEVC.2005.861417

Kasat, R. B., & Gupta, S. K. (2003). Multi-objective optimization of an industrial fluidized-bed catalytic cracking unit (FCCU) using genetic algorithm (GA) with the jumping genes operator. ***Computers and Chemical Engineering***, 27(12), 1785–1800. https://doi.org/10.1016/S0098-1354(03)00153-4

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. Neural Networks, 1995. Proceedings., IEEE International Conference on, 4, 1942–1948 vol.4. https://doi.org/10.1109/ICNN.1995.488968

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). ***Optimization by Simulated Annealing. Science,*** 220(4598), 671–680. https://doi.org/10.1126/science.220.4598.671

Kurtz, L. (2010). Gandhi And His Legacies. ***In Encyclopedia of Violence, Peace, and Conflict*** (pp. 837–851). https://doi.org/10.1016/B978-012373985-8.00073-8

Liu, Z.-Z., Chu, D.-H., Song, C., Xue, X., & Lu, B.-Y. (2016). Social learning optimization (SLO) algorithm paradigm and its application in QoS-aware cloud service composition. ***Information Sciences***, 326, 315–333. https://doi.org/10.1016/j.ins.2015.08.004

Masuduzzaman, & Rangaiah, G. P. (2009). ***Multi-Objective Optimization Applications in Chemical Engineering***. (pp. 27–52). Singapore: World Scientific.

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey Wolf Optimizer. ***Advances in Engineering Software***, 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007

Ramteke, M., Trivedi, V., & Ghune, N. (2013). ***Multi-objective Optimization of an Industrial FCC Unit using Adaptive Social Evolution (ASE) Algorithm***. In Energy System Modeling and Optimization Conference. NIT Durgapur.

Rangaiah, G. P. (2009). Multi-objective Optimization: Techniques and Applications in Chemical Engineering. (1st ed.). Singapore: World Scientific.

Rashedi, E., Nezamabadi-pour, H., & Saryazdi, S. (2009). GSA: ***A Gravitational Search Algorithm. Information Sciences***, 179(13), 2232–2248.

Schott, J. R. (1995). ***Fault tolerant design using single and multi-criteria genetic algorithms***. Massachusetts Institute of Technology, Boston, MA.

Sedighizadeh, M., Faramarzi, H., Mahmoodi, M. M., & Sarvi, M. (2014). Hybrid approach to FACTS devices allocation using multi-objective function with NSPSO and NSGA-II algorithms in Fuzzy framework. ***International Journal of Electrical Power & Energy Systems***, 62, 586–598. https://doi.org/10.1016/j.ijepes.2014.04.058

Senge, P. (1990). ***The fifth discipline: The art and science of the learning organization***. New York: Currency Doubleday.

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. ***Journal of Global Optimization***, 341–359. https://doi.org/10.1023/A:1008202821328

Valadi, J., & Siarry, P. (Eds.). (2014). Applications of Metaheuristics in Process Engineering (1st ed.). Cham: ***Springer International Publishing***. https://doi.org/10.1007/978-3-319-06508-3

Van Veldhuizen, D. A., & Lamont, G. B. (1998). Multiobjective Evolutionary Algorithm Research: A History and Analysis. ***Multiobjective Evolutionary Algorithms and Applications***, 88. techreport.

NITISH GHUNE is a Research Scholar in the discipline of Industrial Engineering & Management from IIT-Roorkee.

E-mail : nitishghune@gmail.com

SUBHASH C. SHARMA works as a professor in Electronics & Computer Engg, IIT-Roorkee, Saharanpur Campus. *E-mail: scs60fpt@iitr.ernet.in*

Article received on 06.10.17 | Article Accepted on 26.11.17