

Bespoke Shuffled Frog Leaping Algorithm and its Engineering Applications

Anurag Tripathi

Maharishi Ved Vyas Engineering College Jagadhari, Haryana, India
E-mail: anuragtripathi1305@gmail.com

Tarun K. Sharma, Vipul Singh

Amity University Rajasthan, Jaipur, India and JUET, Guna, MP, India
E-mail: {taruniitr1, om.vipul15}@gmail.com

Abstract— Shuffled Frog Leap Algorithm (SFLA), a metaheuristic algorithms inspired by PSO and DE has proved its efficacy in solving discrete optimization problems. In this paper we have modified SFLA to solve constrained engineering design problems. The proposed modification integrates a simple mechanism to update the position of frog in its memplex in order to accelerate the basic SFLA algorithm. The proposal is validated on four engineering design problems and the statistical results are compared with the state-of-art algorithms. The simulated statistical results indicate that our proposal is a promising alternative to solve these types of optimization problems in terms of convergence speed.

Index Terms— Shuffled Leap Frog Algorithm, SFLA, Engineering Design Problems, Optimization, Constrained Handling.

I. INTRODUCTION

Optimization is a process of choosing the best or alternative solution that also satisfies all the constraints, from the given set of solutions. Optimization plays a vital role in our real life too. Now a day's most of the researchers are working and applying various optimization techniques in optimizing real world engineering design problems, management problems, and agriculture problems and almost in every sphere. Stochastic techniques are gaining much attention in Optimization as they don't need any auxiliary knowledge about problem domain as well as their computational cost and times is also comparatively less than traditional techniques. Stochastic techniques are either inspired by some evolution, natural or biological phenomenon. Genetic Algorithm (GA) [1], Differential Evolution (DE) [2], Particle Swarm Optimization (PSO) [3], Artificial Bee Colony (ABC) [4], Shuffled Frog Leaping Algorithm (SFLA) [5] and a like falls into the category of stochastic algorithms. Their applications into various domains have already proved their efficacy to solve almost every type of optimization problems [6] – [12].

In this paper we are focusing on recently introduced Shuffled Frog Leaping Algorithm by Eusuff and Lansey, 2003 [5]. SFLA, mimics the social and natural behavior of species. SFLA is formulated on the concept of evolution of memplexes in Frogs. SFLA combines the advantages of local search process of PSO and

information exchanging of the shuffled complex evolution. Like other stochastic algorithms SFLA also bears some convergence limitations. In this study we tried to improve the convergence speed of SFLA by incorporating a new search mechanism to update the position of frogs. The proposal is named as Bespoke – SFLA.

Rest of the study is structured as follows: a brief introduction of SFLA is given in Section 2. Section 3 details the proposed Bespoke – SFLA with constrained handling procedure. Engineering design problems with mathematical models are briefed in Section 4. Parameter tuning is discussed in Section 5. Results are analyzed in Section 6. Finally, Conclusions drawn from the study are given in Section 7.

II. INTRODUCTION TO SFLA

SFLA is a population based metaheuristic for solving discrete optimization problems. SFLA also, like others, is inspired by natural memetics. In SFLA, during the local exploitation process, the frogs use to share and inform the ideas to one another. The local searching process in each memplexes is similar to that of PSO local search process. After exploitation process in each memplexes the information is shuffled among different memplexes, and thus local search process moves towards global search process (exploration). Thus, the benefits of GA as well as PSO are combined in SFLA.

SFLA consists of a population of frogs partitioned into different Memplexes. In each memplex a local independent search is performed by the set of frogs. This local searching process is influenced by PSO. Later on exploration or diversification is performed by exchanging information among each memplexes. This phase is inspired by evolution process of GA. SFLA has fewer numbers of parameters to be set and is as:

A. No. of frogs called population size (P) or initial solutions.,

B. number of Memplexes (m), and

C. number of frogs in each memplex (n).

Fig. 1 depicts the searching process of SFLA in a search space.

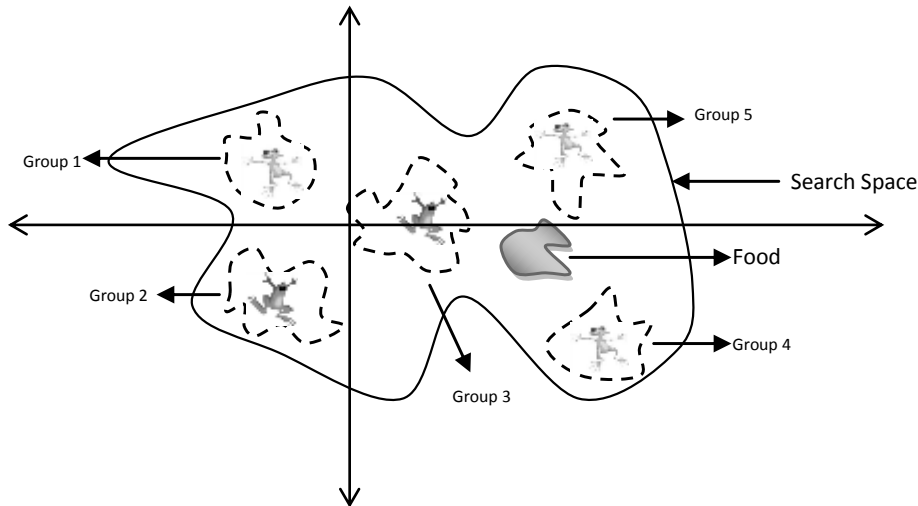


Fig. 1. Searching process in SFLA (initially locally then performs information exchange)

The steps of SFL algorithm are as follows:

1. The population of frogs (P solutions) for D – dimension is generated randomly, a frog (solution) “ i ” is represented as $X_i = (x_{i1}, x_{i2}, \dots, x_{iS})$.
2. Fitness function value is calculated for each solution (frog).
3. Arrange the solutions (frogs based on their of their fitness values) in a descending order
4. Then population of solutions (frogs) is divided into “ m ” number of memeplexes, each containing “ n ” frogs. Thus, $P = m * n$.
5. Assign the frogs to the Memeplexes in such a way that first goes to the first memeplex, the second frog goes to the second memeplex, ..., frog “ m ” goes to the m^{th} memeplex, and frog “ $m+1$ ” goes back to the first memeplex, etc.
6. Then in each memeplex, the frogs (solutions) with worst and best positions are identified.
7. In each memeplex the local search is performed using the equation (1):

$$X_{i+1} = X_i + r \times (X_b - X_w) \quad (1)$$

r is a randomly generated number between 0 and 1. If fitness value of “ X_{i+1} ” is superior to “ X_i ”, reinstate current frog position “ X_i ” with new position “ X_{i+1} ”. Else proceed to Step 8.

8. The frog with the global best fitness (X_g) is identified and global search is done using equation (2):

$$X_{i+1} = X_i + r \times (X_g - X_w) \quad (2)$$

If “ X_{i+1} ” fitness value is better than the fitness value “ X_i ”, replace current frog position “ X_i ” with new position “ X_{i+1} ”. Else go to Step 9.

9. Generate new frog position “ X_i ” randomly to replace the worst frog.

$$X_{i+1} = X_{\min} + r \times (X_{\max} - X_{\min}) \quad (3)$$

$$X_{\min} = (x_{\min 1}, x_{\min 2}, x_{\min 3}, \dots, x_{\min S})$$

$$\text{and } X_{\max} = (x_{\max 1}, x_{\max 2}, x_{\max 3}, \dots, x_{\max S})$$

10. Does the termination criteria met? If yes then stop. Else go to Step 2.

III. BESPOKE SFLA AND COSNTRIAINT HANDLING

In this study we tried to enhance the local search mechanism and convergence rate of basic SFLA by incorporating a scheme that is based on the best individual vector and Scaling factor at a particular generation. The updating scheme is as follows:

$$X_{i+1} = X_b + F \times (X_b - X_w) \quad (4)$$

where F is scaling factor lies between 0 and 1 and rest of the notations are same as discussed in equation (1).

The modification is intended to keep the weighted difference vector between the best and the worst as it is in equation (1) and the remaining base vector X_i and randomly selected vector are replaced with the best vector and a new parameter called scaling factor.

The proposed scheme favors the exploitation since all the vectors are biased by the same direction as of best vector. As a result, the new position updating scheme has better local search ability and faster convergence rate. However, proper tuning of F is required.

Here authors would like to mention that the proposed scheme is also inspired by the behavior of social animals called human beings. Everyone in his life style would like to follow the successful peoples or their paths in order to avoid the failures in achieving their goals.

The new position update mechanism or scheme is embedded into basic SFLA as follows:

$$X_{i+1} = \begin{cases} X_i + r \times (X_g - X_w) & \text{if } \alpha \leq 0.5 \\ X_b + F \times (X_b - X_w) & \text{else} \end{cases} \quad (5)$$

where α is a random real number lies between 0 and 1. If the value of α is below 0.5 then the algorithm will

behave as basic SFLA otherwise bespoke – SFLA will be invoked.

Constraint Handling

In our proposal we have adopted simple constraint-handling methods. Frog positions are compared by pairs:

- if the position of two frogs in a memplex are feasible, we choose the one with a better fitness function value;
- if the position of two frogs in a memplex are infeasible, we choose the frog position with the lower infeasibility degree;
- if position of one frog is feasible and the other is infeasible, we choose the feasible one.

IV. ENGINEERING DESIGN PROBLEMS

To evaluate the performance of the proposal a test bed of four engineering design optimization problems [12] are considered and solved.

A. Welded Beam Design (WBD)

Minimizes the cost of the beam subject to constraints on shear stress, τ , bending stress in the beam, σ , buckling load on the bar, P_c , end deflection of the beam, δ , and side constraints. This problem consists of a nonlinear objective function, five nonlinear and two linear inequality constraints. The solution is located on the boundaries of the feasible region. The ratio of feasible region to entire search space is quite small for welded beam problem. More details can be found in [13].

There are four design parameters x_1, x_2, x_3 and x_4 correspond to h, l, t and b variables, respectively, shown in Fig.2.

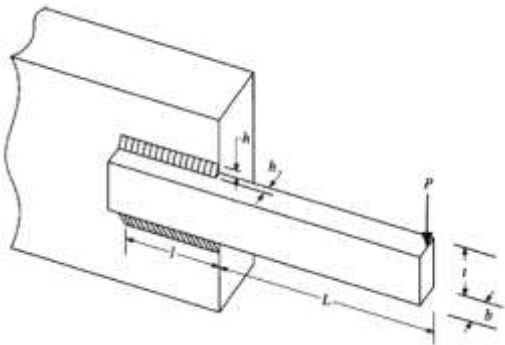


Fig. 2. The welded beam problem

$$\min_x f(X) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2)$$

- Subject to
- $g_1(X): \tau(x) - \tau_{\max} \leq 0$
 - $g_2(X): \sigma(x) - \sigma_{\max} \leq 0$
 - $g_3(X): x_1 - x_4 \leq 0$
 - $g_4(X): 1.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \leq 0$
 - $g_5(X): 0.125 - x_1 \leq 0$
 - $g_6(X): \delta(x) - \delta_{\max} \leq 0$
 - $g_7(X): P - P_c(x) \leq 0$

where $\tau(X) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2}$,

$$\tau' = \frac{P}{\sqrt{2}x_1x_2}$$

$$\tau'' = \frac{MR}{J}$$

$$M = P(L + \frac{x_2}{2})$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}$$

$$J = 2\left\{\frac{x_1x_2}{\sqrt{2}}\left[\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\}$$

$$\sigma(X) = \frac{6PL}{x_4x_3^2}$$

$$\delta(X) = \frac{4PL^3}{Ex_3^3x_4}$$

$$P_c = \frac{4.013E\sqrt{x_3^2x_4^6}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$$

and $P = 6000\text{lb.}$, $L = 14\text{ in.}$, $\delta_{\max} = 0.25\text{ in.}$, $E = 30 \times 10^6\text{ psi}$, $G = 12 \times 10^6\text{ psi}$, $\tau_{\max} = 1,3600\text{ psi}$, $\sigma_{\max} = 3,0000\text{ psi}$, $X = (x_1, x_2, x_3, x_4)^T$, $0.1 \leq x_1, x_4 \leq 2.0$, $0.1 \leq x_2, x_3 \leq 10$.

B. Pressure Vessel Problem (PVP)

The objective of the PVP is to minimize the total cost of material, forming and welding of a cylindrical vessel shown in Fig.3. The nature of the problem is nonlinear objective function with a nonlinear and three linear inequality constraints. The details of the problem can be found in [13]. The mathematical model of PVP is given as:

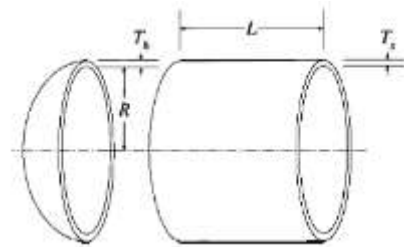


Fig. 3. The pressure vessel problem

$$\min_x f(X) = 0.6224x_1x_2x_3 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

- Subject to
- $g_1(X): -x_1 + 0.0193x_3 \leq 0$
 - $g_2(X): -x_2 + 0.00954 \leq 0$
 - $g_3(X): -\pi x_3^2x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0$
 - $g_4(X): x_4 - 240 \leq 0$

where $X = (x_1, x_2, x_3, x_4)^T$. The ranges of the design parameters are $0 \leq x_1, x_2 \leq 99$, $10 \leq x_3, x_4 \leq 200$.

C. Speed Reducer Design (SRD)

The objective of SRD is to minimize the weights of the speed reducer. Fig. 4 presents the outline graph of the SRD. This problem consists of 11 constraints out of which 4 are linear and 7 are nonlinear. The mathematical model is given below and the details of the SRD problem can be consulted in [13].

$$\min_x f(X) = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0394) - 1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3)$$

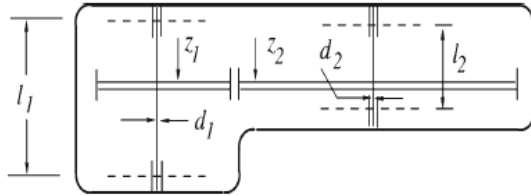


Fig. 4. The speed reducer problem

$$\begin{aligned} \text{Subject to } g_1(X) &: \frac{27}{x_1x_2^2x_3} - 1 \leq 0 \\ g_2(X) &: \frac{397.5}{x_1x_2^2x_3} - 1 \leq 0 \\ g_3(X) &: \frac{1.93x_4^3}{x_2x_3x_6^4} - 1 \leq 0 \\ g_4(X) &: \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \leq 0 \\ g_5(X) &: \frac{\left(\left(\frac{745x_4}{x_2x_3}\right)^2 + 157.5 \times 10^6\right)^{1/2}}{110.0x_6^3} - 1 \leq 0 \\ g_6(X) &: \frac{\left(\left(\frac{745x_4}{x_2x_3}\right)^2 + 157.5 \times 10^6\right)^{1/2}}{85.0x_7^3} - 1 \leq 0 \\ g_7(X) &: \frac{x_2x_3}{40} - 1 \leq 0 \\ g_8(X) &: \frac{5x_2}{x_1} - 1 \leq 0 \\ g_9(X) &: \frac{x_1}{12x_2} - 1 \leq 0 \\ g_{10}(X) &: \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0 \\ g_{11}(X) &: \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0 \end{aligned}$$

where $2.6 \leq x_1 \leq 3.6$, $0.7 \leq x_2 \leq 0.8$, $17 \leq x_3 \leq 28$, $7.3 \leq x_4 \leq 8.3$, $7.8 \leq x_5 \leq 8.3$, $2.9 \leq x_6 \leq 3.9$, $5.0 \leq x_7 \leq 5.5$.

D. Tension/Compression Problem (TP)

The objective of TP is to minimize of the weight of the tension/compression spring revealed in Fig. 5. This problem has one linear and 3 non-linear inequality constraints with non linear objective function. Details about TP can be referred in [13].

Mathematically the minimization TP problem is formulated as:

$$\min_x f(X) = (N+2)Dd^2$$

$$\begin{aligned} \text{Subject to } g_1(X) &: 1 - \frac{D^3N}{71785d^4} \leq 0 \\ g_2(X) &: \frac{4D^2 - dD}{12566(Dd^3 - d^4)} + \frac{1}{5108d^2} - 1 \leq 0 \\ g_3(X) &: 1 - \frac{140.45d}{D^2N} \leq 0 \\ g_4(X) &: \frac{D+d}{1.5} - 1 \leq 0 \end{aligned}$$

$$X = (d, D, N)^T, 0.05 \leq d \leq 2.0, 0.25 \leq D \leq 1.3, 2.0 \leq N \leq 15.0$$

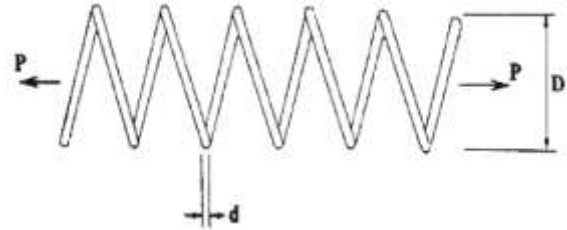


Fig. 5. The tension/compression spring problem

V. PARAMETER SETTINGS

The parameter settings of SFLA and Bespoke-SFLA, for the fair comparison are stated in Table 1. The proposed CM-SFLA is executed in Dev C++. The population of frogs is generated using inbuilt *rand()* function.

Table 1. Parameterizations for test systems

Population Size of Frogs	50
Memplexes (m)	5
Local Explorations iterations in each Memplexes	10
Number of Function Evaluations (NFE)	50000
D_{max}	100% of variable range
F (Scaling Factor)	0.5

VI. STATISTICAL RESULTS

The simulated statistical results of four engineering design problems are compared with the results reported in the literature. The best values with the values of decision variables are for each engineering design problem are given in the Table 2 – Table 5.

We have performed 30 independent runs; where in each run 24,000 objective functions were evaluated.

We have compared the simulated results with that of available in the literature [14] [15]. [15] reached the best solution after 30,000 function evaluations, which is comparatively larger than taken by our proposal. The statistical results in terms of best values achieved are illustrated in Table 2 – Table 5. Comparative results in terms of mean, standard deviation (Std. Dev.) and best are presented in Table 6.

Table 2. Results of WBD

	Best Values (Solutions)
x_1	2.0573E-01
x_2	3.4705E+00
x_3	9.0366E+00
x_4	2.0573E-01
$g_1(x)$	-1.8190E-12
$g_2(x)$	-3.7210E-03
$g_3(x)$	0.0000E+00
$g_4(x)$	-3.4330E+00
$g_5(x)$	-8.0729E-02
$g_6(x)$	-2.3554E-01
$g_7(x)$	0.0000E+00
$f(x)$	1.7249E+00

Table 3. Results of PVP

	Best Values (Solutions)
x_1	8.1250E-01
x_2	4.3750E-01
x_3	4.2098E+01
x_4	1.7664E+02
$g_1(x)$	-4.5000E-15
$g_2(x)$	-3.5880E-02
$g_3(x)$	-1.1640E-10
$g_4(x)$	-6.3363E+01
$f(x)$	6.0597E+03

Table 4. Results of SRD

	Best Values (Solutions)
x_1	3.5000E+00
x_2	7.0000E-01
x_3	1.7000E+01
x_4	7.3000E+00
x_5	7.8000E+00
x_6	3.3502E+00
x_7	5.2867E+00
$g_1(x)$	-7.3915E-02
$g_2(x)$	-1.9800E-01
$g_3(x)$	-4.9917E-01
$g_4(x)$	-9.0147E-01
$g_5(x)$	0.0000E+00
$g_6(x)$	-5.0000E-16
$g_7(x)$	-7.0250E-01
$g_8(x)$	-1.0000E-16
$g_9(x)$	-5.8333E-01
$g_{10}(x)$	-5.1325E-02
$g_{11}(x)$	-1.0852E-02
$f(x)$	2.9963E+03

Table 5. Results of TP

	Best Values (Solutions)
x_1	5.1583E-02
x_2	3.5419E-01
x_3	1.1439E+01
$g_1(x)$	-2.0000E-16
$g_2(x)$	-1.0000E-16
$g_3(x)$	-4.0488E+00
$g_4(x)$	-7.2948E-01
$f(x)$	1.2665E-02

Table 6. Comparative Statistical Results

Problems		Optimal	SFLA	Bespoke – SFLA	COPSO	Mezura
WBD	Mean	0.724852	1.262611	1.72782	1.724800	1.777600
	Std. Dev.		1.21E-03	1.03E-02	1.20E-05	8.80E-02
	Best		1.72390	1.724852	1.724852	1.724852
PVP	Mean	6059.714335	6129.0732	6073.15	6071.013300	6379.938000
	Std. Dev.		1.54E+01	1.82E+01	1.51E+01	2.10E+02
	Best		6058.974722	6059.71583	6059.714335	6059.714300
SRD	Mean	NA	2997.1973	2996.382	2996.408500	2996.348000
	Std. Dev.		1.13E-01	1.0E-01	2.86E-02	0.00E+00
	Best		2996.46262	2996.39022	2996.372448	2996.348094
TP	Mean	0.012665	0.013301	0.01378	0.012600	0.013100
	Std. Dev.		1.78E-06	3.19E-04	1.20E-06	3.90E-04
	Best		0.0126751	0.0126691	0.012665	0.012689

VII. CNCLUSIONS AND FUTURE SCOPE

In this study we have presented a simple modification in the local searching process of Frogs in each

memeplexes. The searching process is updated using new search mechanism that is based and biased on best search and secondly embedding scaling factor enhances the convergence rate of the proposal. The proposal is named

as Bespoke – SFLA. Bespoke SFLA has shown a good performance on four constrained engineering design problems. We have compared the statistical results of our proposal with the basic SFLA and two algorithms results found in the literature. Our algorithm found to be capable in obtaining near optimal solutions with fewer number of function evaluations. This proposal could be an alternative for solving such kind of problems. In future we will try to study the theoretical concept of the proposal and applications in multi-objective optimization problems.

REFERENCES

- [1] Goldberg D., *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley, (1989).
- [2] Storn R. and Price K. V., “Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces,” *J. Global Optimization*, vol. 11, pp. 341–359, (1997).
- [3] J. Kennedy and R. C. Eberhart, “Particle Swarm Optimization”, *Proceeding of IEEE International Conference on Neural Networks, Perth, Australia*, IEEE Service Center, Piscataway, NJ (1995), pp. 1942–1948.
- [4] D. Karaboga, “An Idea based on Bee Swarm for Numerical Optimization,” *Technical Report*, TR-06, Erciyes University Engineering Faculty, Computer Engineering Department (2005).
- [5] Eusuff, M., Lansey, K.E.: “Optimization of water distribution network design using the shuffled frog leaping algorithm,” *Water Resources Planning and Management* 129(3), 210–225 (2003).
- [6] TK Sharma, M Pant. “Redundancy Level Optimization in Modular Software System Models using ABC,” *International Journal of Intelligent Systems and Applications* (IJISA) 6 (4), 40, 2014.
- [7] TK Sharma, M Pant. “Improvised Scout Bee Movements in Artificial Bee Colony,” *I.J. Modern Education and Computer Science*, 2014, 1, 1-16.
- [8] TK Sharma, M Pant. “Swarm Intelligence in Pulp and Paper Process Optimization. Applications of Metaheuristics in Process Engineering,” 123-151, 2014.
- [9] M Ali, CW Ahn, P Siarry. “Differential evolution algorithm for the selection of optimal scaling factors in image watermarking,” *Engineering Applications of Artificial Intelligence*, 31, 15-26, 2014.
- [10] Panchi Li, Hong Xiao “An improved quantum-behaved particle swarm optimization algorithm,” *Applied Intelligence*, April 2014, Volume 40, Issue 3, pp 479-496
- [11] Sunita Panda, Archana Sarangi, Siba Prasada Panigrahi. “A new training strategy for neural network using shuffled frog-leaping algorithm and application to channel equalization,” *AEU - International Journal of Electronics and Communications*. DOI: 10.1016/j.aeue.2014.05.005
- [12] Tarun K. Sharma, Millie Pant and Deepshikha Bhargava, “Impact of Modification Rate in Artificial Bee Colony for Engineering Design Problems,” *J. Information Engineering and Electronic Business*, 2013, 6, 55-63
- [13] Leticia C. Cagnina and Susana C. Esquivel, Carlos A. Coello Coello. “Solving Engineering Optimization Problems with the Simple Constrained Particle Swarm Optimizer,” *Informatica*, 32(2008) 319–326.
- [14] A. Hernandez Aguirre, A. Muñoz Zavala, E. Villa Diharce and S. Botello Rionda. “COPSO: Constrained Optimization via PSO Algorithm,” *Technical report No. I-*

07-04/22-02-2007, Center for Research in Mathematics (CIMAT), 2007.

- [15] E. Mezura and C. Coello. “Useful Infeasible Solutions in Engineering Optimization with Evolutionary Algorithms,” *Lect. Notes Comput. Sc.*, 3789:652–662, 2005.

Authors’ Profiles



Anurag Tripathi, B.Tech and pursuing his M.Tech in Electronics and Communication Engineering from Maharishi Ved Vyas Engineering College Jagadhari, Haryana, India. His area of interest is evolutionary algorithms and their engineering applications. He has several national and international publications

into his credit.



Tarun Kumar Sharma: PhD from IIT Roorkee and presently aligned with Amity Institute of Information Technology, Amity University Rajasthan, India. His research areas are evolutionary & Swarm intelligence algorithms and their applications in Software Engineering. He is in Editorial Board and

reviewer of many refereed Journals. He has published about 50 research papers in Journal of repute and in refereed international Journals.

Vipul Singh is an Engineering graduate in Computer Science discipline from Jaypee University of Engineering & Technology, Guna, MP, India. He is very keen into research in the field of engineering optimization. using memetic algorithms.

How to cite this paper: Anurag Tripathi, Tarun K. Sharma, Vipul Singh, "Bespoke Shuffled Frog Leaping Algorithm and its Engineering Applications", *International Journal of Intelligent Systems and Applications* (IJISA), vol.7, no.4, pp.41-46, 2015. DOI: 10.5815/ijisa.2015.04.06