A Novel Deluge Swarm Algorithm for Optimization Problems

Anahita Samadi[1,*] - Mohammad Reza Meybodi[2]
[1] Science and Research Branch, Islamic Azad University, Qazvin, Iran
[2] Soft Computing Laboratory, Computer Engineering and Information Technology Department, Amirkabir University of Technology, Tehran, Iran

[*] Corresponding author. E-mail addresses:Anahita.samadi@gmail.com.

ABSTRACT

In this study, a novel population based algorithm which is based on great deluge, is presented for solving optimization problems. Optimization problems due to the vast application in real life and science are so important. By applying more limitation they are classified as NP-hard problems. NP-hard problems, because of high complexity, can be solved by Meta-heuristic methods. In this regards population-based approaches are considered as a good option.

According to the proposed algorithm mechanism, local and global search can be done. Some of the advantages of this algorithm include simplicity, avoid trapping in local optima, appropriate balance between local and global search. The approach is examined on standard functions. Ackley and sphere, for instance are taking into account. The results are compared with G-PSO, G-HS, GSA, CM-AFSA and M-ABC which in all cases proposed algorithm shows better results.

Keywords: Deluge Swarm algorithm; Optimization; NP-hard problems

1. INTRODUCTION

In today's world, optimization plays crucial role for solving various problems especially scientific issues [1]. According to problem complexity, different solutions and algorithms have been suggested. The goal of all optimization algorithms is maximize or minimize the cost function [2].

Real world problems are most classified as NP-hard problems which need complicated computation [3]. Meta-heuristic techniques have shown great ability in solving this kind of problems. In this approach, optimal solution can be gained by iterating process which tries to enhance the result in each evaluation by different mechanisms [4].One of the most popular approaches in Meta-heuristic algorithms is population based which works according to swarm intelligent methods. Swarm intelligent is inspired by associative behavior of organisms such as fishes, insects and birds. The rules between members of the swarm lead to effective use of resources therefore optimal solution for special problem can be achieved [5].

This paper presents an efficient algorithm inspired by the great deluge that was introduced in 1993 by Dueck [6]. Great deluge algorithm inspired by people climbing up to avoid sinking. Great deluge algorithm initiates its parameters based on estimation of environment and consider them as a primary answer, in the next iterations it produces more answers based on primary answer, afterwards new parameter called “water level’’ engaged as a boundary for accepting result in which one with greater water level is accepted. "Water level'' rises gradually with new "rain speed" parameter [6]. Great deluge algorithm is a robust, easy and flexible algorithm. It has been used in solving different problems including, record-to-record travel [7], examination time tabling Problem [8], buffer allocation problem in unreliable production lines [9] and efficiently dynamic layout [10].

In the proposed algorithm to exploit swarm intelligence, agents just as members of swarm are engaged and follow the rules of great deluge algorithm. Moreover new mechanisms are used for agent's interaction to set them in search space. In this way high diversity of results are provided, thus proposed algorithm can find better peaks and avoids being trapped in local optima. Proposed algorithm which is called "Deluge swarm optimization" has simple structure and uses low numbers of parameters.

In this study, Deluge Swarm Optimization have been tested on eight standard benchmark functions, such as Zakharov, Schwefel, Sphere, Ackley then It's results compared to the well-known algorithms including global-best particle swarm optimization(GPSO) [11], modified artificial bee colony algorithm (M-ABC) [12], gravitational search algorithm(GSA) [13], modified artificial fish swarm algorithm with communication behavior(CM-AFSA) [14], and global-best harmony search [15].

The rest of the paper is organized as follows. Detail of the proposed algorithm and flow chart are presented in section 2. The results and comparison with other algorithms are presented in section 3. The conclusion is presented in sections 4.

2. DELUGE SWARM OPTIMIZATION

Proposed algorithm is a population- based namely Deluge Swarm Optimization. Similar to standard Great Deluge, in this novel algorithm the water level is supposed to increase due to probable floods and a group of people try to save their lives by climbing up to high altitudes. Generally, in suggested algorithm, the best position of individuals is the higher one, and all agents try to position themselves in suitable places near agents in higher altitudes. Simultaneously the water level is increasing according to raining rate; population at the same time should climb up higher than water levels.

In a standard Great Deluge, the rain's velocity parameter is a small constant value. In each iteration, raining with constant rate leads to an increase in the water level constantly. Determining the appropriate raining velocity amount is a challenging problem in the algorithm. In our suggested algorithm, the velocity value is considered equal to zero in initiation, from the second iteration the parameter's value is set to the difference of average fitness value of the population category in the current and previous iteration. Thus, the value of the parameter is determined based on the range of fitness. The determined agents' values in early iterations where they move with longer steps for a global search are considerable and gradually that with increasing the ability of local search after then agents' steps shortened so the fitness values decreases. This decrease in the values continues until the water level rises be in proportion to the population moving step in transforming a global search capability to local search during the process of optimization.

In real world, people getting stuck in a situation similar to Deluge algorithm, usually try to reach themselves to safer positions by evaluating the surroundings. This reaction is considered as a local search and simulated in proposed algorithm. For this aim, people being positioned in reasonable places (i.e. higher altitudes) search the surroundings for better positions. A search-space with the length "Vision-radius" is considered for agents and each agent can search this space for better positions. "Vision-radius" amount will be decreased gradually so that the capability of local search is increased and the steps are in proportion with the space, the algorithm is going to search in, for better positions.

To implement agents'search process, in proposed algorithm, each agent positioning above water level is set to assess neighboring different positions in "Vision-radius" domain equal to "try number" times. If the assessed position is more suitable than the current position, the agent moves to that position. Implementing the mentioned process is as follows:

1. First, the *ith* agent placed in *Xi* position considers a new position *Xtarget* by the following equation:

$$X_{Target,d} = X_{i,d} + \left(rand\left(-1,1\right) \times Vision\,Radius\right) \qquad (1)$$

In the above equation rand (-1,1) creates a random uniformly distributed number in the interval (-1,1). The fitness of calculated *Xtarget* in all dimensions is then assessed, if the fitness is proven to be better than the *Xi*, then the agent moves forward to it. The above mentioned process is repeated "try-number" times in each iteration.

2. In addition to individual search, group search is performed by the population except the agent with the best position. Individual agents try to reach themselves to a better location following the others with better locations. For this purpose, each agent randomly chooses the agent with better position. According to equation 2 the new location is:

$$X_{Temp,d} = X_{i,d} + \left(\left(X_{j,d} - X_{i,d}\right) \times rand\left(0,2\right)\right) \qquad (2)$$

Xtemp in all dimensions is calculated based on the position of the chosen random agent (i.e. *jth* agent). The new location is assessed for fitness. While its location is better than the current position of ith agent and placed above water level, the ith agent moves toward it. Pseudo code of proposed algorithm is presented in Table 1.

Table 1. *Pseudo code of proposed algorithm*

Deluge Swarm Optimization:

```
initialize N(population size),Vision Radius
for each Person_i do
    initialize X_i randomly in D-dimensional search space
Endfor
evaluate all People
Repeat
    Calculate Rain Speed based on difference of fitness average in itr_T and itr_T-1
    Water Level = Water Level + Rain Speed
    Update Vision Radius
    for each Person_i do
        if fitness of Person_i is better than Water Level then
            for i=1 to Try Number do
                obtain X_Target by Eq. (1)
                check search space boundary for X_Target
                If f(X_Target) is better than f(X_i) then
                    X_i = X_Target
                endif
            endfor
        Endif
    Endfor
    foreach  Person_i do
```

determine $Person_j$ randomly which $f(X_j)<f(X_i)$
Obtain X_{Temp} by Eq. (2)
check search space boundary for X_{Temp}
if $f(X_{Temp})<f(X_i)$ **&** $f(X_{Temp})<$*Water Level* **then**
$X_i = X_{Temp}$
endif
Endfor
Until stopping criterion is met

3. EXPERIMENTAL RESULTS

In this section, the proposed algorithm has been tested over 8 standard function benchmark which generally applied as acceptable test tools for optimization algorithm in continuous static environment. Function properties and dimension of the problem are defines in table 2 [16]. Optimal value for tested functions is equal to zero.

Table 2. *Test functions that used in this study*

Name	Test Function	D	Search range	f_{Min}
Zakharov	$f_4(x)=\sum_{i=1}^{D} x_i^2 + \left(\sum_{i=1}^{D} 0.5 i x_i\right)^2 + \left(\sum_{i=1}^{D} 0.5 i x_i\right)^4$	10	$[-5,10]^D$	0
Schwefel 2.22	$f_6(x)=\sum_{i=1}^{D}\lvert x_i\rvert + \prod_{i=1}^{D}\lvert x_i\rvert$	30	$[-10,10]^D$	0
Hyper-ellipsoid	$f_8(x)=\sum_{i=1}^{D} i x_i^2$	30	$[-5.12,5.12]^D$	0
Sphere	$f_{11}(x)=\sum_{i=1}^{D} x_i^2$	30	$[-100,100]^D$	0
Schaffer	$f_{16}(x)=0.5+\dfrac{\sin^2\left(\sqrt{x_1^2+x_2^2}\right)-0.5}{\left(1+0.001\left(x_1^2+x_2^2\right)\right)^2}$	2	$[-100,100]^D$	0
Ackley	$f_{19}(x)=-20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right)-\exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos 2\pi x_i\right)+20+e$	30	$[-32,32]^D$	0
Penalized	$f_{22}(x)=\frac{\pi}{D}\left\{10\sin^2(\pi y_1)+\sum_{i=1}^{D-1}(y_i-1)^2\left(1+10\sin^2(\pi y_{i+1})\right)+(y_D-1)^2\right\} + \sum_{i=1}^{D} u(x_i,10,100,4)$ $y_i = 1+\frac{1}{4}(x_i+1)$ $u(x_i,a,k,m)=\begin{cases} k(x_i-a)^m, & x_i>a \\ 0, & -a\le x_i\le a \\ k(-x_i-a)^m, & x_i<-a \end{cases}$	30	$[-50,50]^D$	0

Rotated Penalized	$f_{27}(x)=\frac{\pi}{D}\left\{10\sin^2(\pi y_1)+\sum_{i=1}^{D-1}(w_i-1)^2\left(1+10\sin^2(\pi y_{i+1})\right)+(y_D-1)^2\right\}$ $+\sum_{i=1}^{D}u(x_i,10,100,4)$ $y_i=1+\frac{1}{4}(w_i+1)$ $u(w_i,a,k,m)=\begin{cases} k(w_i-a)^m, & w_i>a \\ 0, & -a\le w_i\le a \\ k(-w_i-a)^m, & w_i<-a \end{cases}$ $w=x\times M^*$ $^*M$ *is an orthogonal Matrix*	30	$[-50,50]^D$	0

Deluge Swarm Optimization and five well-known algorithms including G-PSO [11], G-HS [15], GSA [13], CM-AFSA [14] and M-ABC [12] are examined over functions that are mentioned in Table 2. Proposed algorithm configuration and other algorithms are shown in Table 3. Settings of other algorithms are taken from related references. Configuration of proposed algorithm is the result of different experiments which lead to best performance of this algorithm. Mentioned algorithms' Average and standard deviation from 30 replication are demonstrated in Table 4.

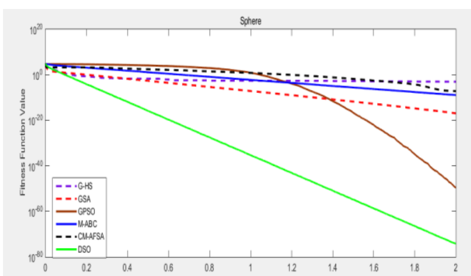
Table 3. *Parameter setting of algorithms*

Algorithm	Parameter Setting
DSO	Population size = 20 try-number=5 W=0.98
G-PSO	Population size =20 $c_1=c_2=2$ $W_{min}=0.4$ $W_{max}=0.9$ neighbor topology=Global star Inertia weight = linear decrement
G-HS	Population size(HMS) =5 HMCR=0.9 $PAR_{min}=0.01$ $PAR_{max}=0.99$
CM-AFSA	Population size =50 try-number=5 crowd factor=0.75 Visual=Par[1] Step = Visual/5 $c_3=c_6=2$
GSA	Population size =50 α=20 G0=100 k0=50
M-ABC	Population size =40 Modification rate=0.8 Limit = 0.5 ×Dimension ×Population size Scaling factor = 1

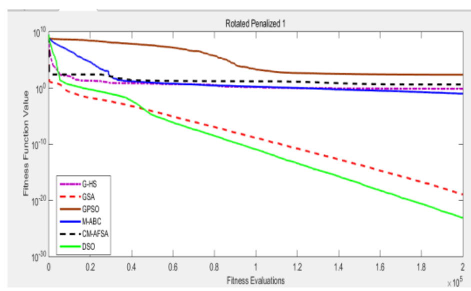
Table 4. *Average and standard deviation results of 6 algorithms of functions showed in table 2*

FUNCTION	G-PSO	G-HS	GSA	M-ABC	CM-AFSA	DSO
Zakharov	19.07 (37.72)	0.0001 (0.0003)	3.14e-18 (1.19e-18)	1.14e-15 (1.59e-15)	3.87e-14 (1.18e-13)	**2.60e-37** **(1.06e-37)**
Schwefel 2.22	2105.73 (4441.42)	0.01 (0.01)	1.81e-08 (2.09e-09)	1.43e-08 (5.07e-09)	0.0009 (0.0008)	**2.33e-11** **(2.58e-12)**
Hyper-ellipsoid	102.76 (82.19)	3.43e-05 (6.72e-05)	1.11e-16 (2.99e-17)	3.68e-11 (2.11e-11)	5.08e-08 (8.15e-08)	**8.00e-17** **(4.19e-17)**
Sphere	2.48e-50 (3.82e-51)	0.0009 (0.001)	1.29e-17 (2.61e-18)	1.35e-09 (6.55e-10)	7.19e-08 (4.57e-08)	**6.60e-75** **(3.84e-75)**
Schaffer	**0** **(0)**	0.01 (0.01)	0.01 (0.0036)	0.0001 (0.0001)	9.79e-5 (0.0004)	**0** **(0)**
Ackley	0.98 (2.79)	0.005 (0.004)	2.77e-09 (2.90e-10)	1.63e-05 (5.30e-06)	1.49 (0.92)	**5.56e-10** **(3.05e-10)**
Penalized	0.09 (0.22)	5.36e-06 (6.44e-06)	0.002 (0.01)	0.002 (0.008)	2.07 (1.83)	**1.39e-17** **(3.04e-18)**
Rotated Penalized	236.48 (2813.94)	0.73 (1.29)	1.10e-19 (3.12e-20)	0.09 (0.08)	4.26 (2.83)	**7.10e-24** **(2.33e-24)**

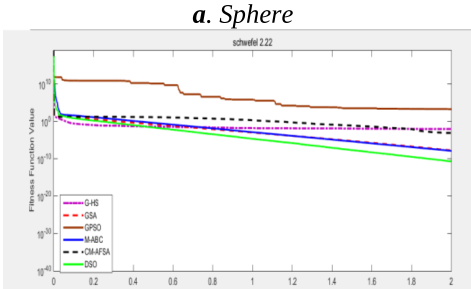
As it can be seen in Tables4, Deluge Swarm Optimization achieves best results in all functions except Schaffer one ,in that case proposed algorithm and G-PSO gain the best results among the tested algorithm.



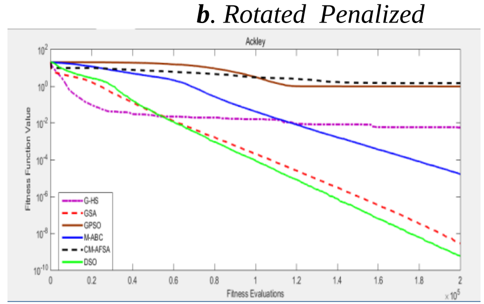
***a**. Sphere*



***b**. Rotated Penalized*



***c.** Schwefel 2.22*



***d.** Ackley*

Fig. 1. *Convergence behavior of Sphere, Rotated Penalized, Schwefel 2.22 and Ackley functions*

4. CONCLUSION

In this study, novel algorithm namely Deluge Swarm Optimization based on great Deluge algorithm has been proposed. This new algorithm taking advantage of the multi agents, enables proposed algorithm to make diversity in search space. Agents in proposed algorithm improve their result by comparison to water level parameter individually. On the other hand with the ability to comparison between agents, Deluge Swarm Optimization is capable to local and global search leading to reasonable results, and acceptable convergence rate. Some of the privilege of this algorithm is including simplicity, avoiding local optima, appropriate balance between local and global search. To prove the efficiency of proposed algorithm the results were compared with G-PSO, G-HS, GSA, CM-AFSA, M-ABC using some standard functions including Ackley and sphere. Comparing results showed that the proposed algorithm performs better than all other algorithms with mentioned functions.

REFERENCES

[1] Cheng, H., & Yang, S. (2010, April). Multi-population genetic algorithms with immigrants scheme for dynamic shortest path routing problems in mobile ad hoc networks. In *European Conference on the Applications of Evolutionary Computation* (pp. 562-571). Springer Berlin Heidelberg.

[2] Lung, R. I., & Dumitrescu, D. (2007, July). A new collaborative evolutionary-swarm optimization technique. In *Proceedings of the 9th annual conference companion on Genetic and evolutionary computation* (pp. 2817-2820). ACM.

[3] Li, X., Branke, J., & Blackwell, T. (2006, July). Particle swarm with speciation and adaptation in a dynamic environment. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation* (pp. 51-58). ACM.

[4] Yang, S., & Li, C. (2010). A clustering particle swarm optimizer for locating and tracking multiple optima in dynamic environments. *IEEE Transactions on Evolutionary Computation*, *14*(6), 959-974.

[5] Yazdani, D., Sadeghi-Ivrigh, S., Yazdani, D., Sepas-Moghaddam, A., & Meybodi, M. R. (2015). Fish Swarm Search Algorithm: A New Algorithm for Global Optimization. *International Journal of Artificial Intelligence™*, *13*(2), 17-45.

[6] Dueck, G. (1993). New optimization heuristics: The great deluge algorithm and the record-to-record travel. *Journal of Computational physics*, *104*(1), 86-92.

[7] Dueck, G. (1993). New optimization heuristics: The great deluge algorithm and the record-to-record travel. *Journal of Computational physics*, *104*(1), 86-92.

[8] Burke, E., Bykov, Y., Newall, J., & Petrovic, S. (2004). A time-predefined local search approach to exam timetabling problems. *Iie Transactions*, *36*(6), 509-528.

[9] Nahas, N., Ait-Kadi, D., & Nourelfath, M. (2006). A new approach for buffer allocation in unreliable production lines. *International journal of production economics*, *103*(2), 873-881.

[10] Nahas, N., Kadi, D. A., & El Fath, M. N. (2010). *Iterated great deluge for the dynamic facility layout problem*. CIRRELT.

[11] Shi, Y., & Eberhart, R. (1998, May). A modified particle swarm optimizer. In *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on* (pp. 69-73). IEEE.

[12] Akay, B., & Karaboga, D. (2012). A modified artificial bee colony algorithm for real-parameter optimization. *Information Sciences*, *192*, 120-142.

[13] Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). GSA: a gravitational search algorithm. *Information sciences*, *179*(13), 2232-2248.

[14] Tsai, H. C., & Lin, Y. H. (2011). Modification of the fish swarm algorithm with particle swarm optimization formulation and communication behavior. *Applied Soft Computing*, *11*(8), 5367-5374.

[15] Omran, M. G., & Mahdavi, M. (2008). Global-best harmony search. *Applied mathematics and computation*, *198*(2), 643-656.

[16] Zhan, Z. H., Zhang, J., Li, Y., & Chung, H. S. H. (2009). Adaptive particle swarm optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(6), 1362-1381.