

# On Four Metaheuristic Applications to Speech Enhancement

## — Implementing Optimization Algorithms with MATLAB R2018a

Su-Mei Shiue

Department of Computer Science and Information Engineering  
National Taipei University of Technology  
[t102599005@ntut.edu.tw](mailto:t102599005@ntut.edu.tw)

Lang-Jyi Huang

Department of Computer Science and Information Engineering  
National Taipei University of Technology  
[t8599003@ntut.edu.tw](mailto:t8599003@ntut.edu.tw)

Wei-Ho Tsai

Department of Electronic Engineering  
National Taipei University of Technology  
[whtsai@ntut.edu.tw](mailto:whtsai@ntut.edu.tw)

Yen-Lin Chen

Department of Computer Science and Information Engineering  
National Taipei University of Technology  
[ylchen@csie.ntut.edu.tw](mailto:ylchen@csie.ntut.edu.tw)

### Abstract

This study aims to firstly implement four well-known metaheuristic optimization algorithms which, among other things, are utilized on adaptive filter design, for dual-channel speech enhancement, with voice communication devices. The implementations are conducted in a simulation fashion using MATLAB code under its newly release version of R2018a. Lately, the study takes a closer look at these four optimization methods, based on learning from the literature of the research society, on their pros and cons while applying to speech enhancement. These four methods are, namely, (1) Accelerated Particle Swarm Optimization (APSO), (2) Gravitational Search Algorithm (GSA), (3) a hybrid algorithm of PSO (Particle Swarm Optimization) and GSA (called hybrid PSOGSA), and (4) Bat Algorithm (BA). This

study performs the said implementations successfully and obtains useful experimental results which contributes to building a solid foundation for more advanced research in the near future. Besides, the implementations made by the study confirm the correctness of many a previous research works which claim that these four algorithms show better performance on improved speech quality and intelligibility as compared to that of the existing standard PSO (SPSO) based speech enhancement approach.

Keywords: Speech Enhancement, APSO, GSA, PSO, GSA, BA.

## 1. Introduction

The concise characteristics and advantages of metaheuristic optimization methods as discussed in this paper for speech enhancement are listed in Table 1 below.

Table 1. Important metaheuristic optimization methods for speech enhancement

Name	Characteristics	
PSO	PSO does not require that the problem of optimization to be differentiated as required by classical optimization methods such as gradient descent and quasi Newton methods. Therefore, PSO can be used in optimization problems that are partly irregular, variable over time, etc. [5].	Simple conception, quick velocity, significant SNR ratio for higher accuracy. person will percept/ recognize the v
GSA	An algorithm inspired by Newton's laws of motion and gravitation force [1].	Updating is done by considering the performance than PSO [3].
PSO, GSA	A hybrid algorithm of PSO and GSA [3].	Combines the advantages of GSA and designing the adaptive filter in noise
APSO	Developed by Xin she Yang in 2010. APSO is simpler to implement and has faster convergence when compared to the standard PSO (SPSO) algorithm. [2]	APSO based speech enhancement improves speech quality and intelligibility m
BA	(1) Frequency tuning: BA uses echolocation and frequency tuning to solve problems. (2) Automatic zooming: BA has a distinct advantage over other metaheuristic algorithms. of automatically zooming into a region where promising solutions have been found [4].	BA is a bio-inspired algorithm developed diversity and provides a dynamic b search space [3].

## 2. Accelerated Particle Swarm Optimization (APSO) to Speech Enhancement

### 2.1 PSO: Advantages and Pseudo Code [7]

One of the most widely used swarm-intelligence-based algorithms owing to its simplicity and flexibility, particle swarm optimization, or PSO, was developed by Kennedy and Eberhart in 1995 [8]. The PSO algorithm searches the space of an objective function by adjusting the trajectories of individual agents, called *particles*, as the piecewise paths formed by positional vectors in a quasi-stochastic manner. The movement of a swarming particle is composed of two main components: a stochastic component and a deterministic component. Each particle is attracted toward the position of the current global best  $g^*$  and its own best location  $x_i^*$  in history, while at the same time it tends to move randomly.

When a particle finds a location that is better than any previously found locations, it updates that location as the new current best for particle  $i$ . There is a current best for all  $n$  particles at any time  $t$  during iterations. The aim is to find the global best among all the current best solutions until the objective no longer improves or after a certain number of iterations. The movement of particles is schematically represented in Figure 1, where  $x_i^*(t)$  is the current best for particle  $i$ , and  $g^* \approx \min\{f(x_i)\}$  for  $(i = 1, 2, \dots, n)$  is the current global best at  $t$ . moving toward the global best  $g^*$  and the current best  $x_i^*$  for each particle  $i$ .

The essential steps of the particle swarm optimization are summarized as the pseudo code shown in Figure 2.

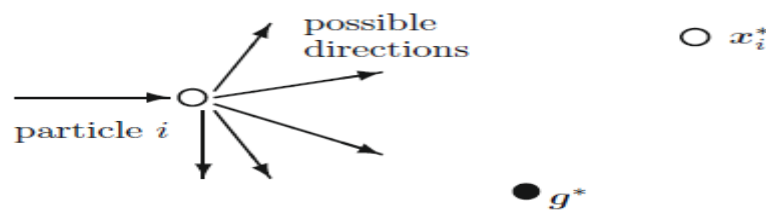


Figure 1. Schematic representation of the movement of a particle in PSO [7]

### Particle Swarm Optimization

---

```

Objective function  $f(x)$ ,  $x = (x_1, \dots, x_d)^T$ 
Initialize locations  $x_i$  and velocity  $v_i$  of  $n$  particles.
Find  $g^*$  from  $\min\{f(x_1), \dots, f(x_n)\}$  (at  $t = 0$ )
while ( criterion )
  for loop over all  $n$  particles and all  $d$  dimensions
  Generate new velocity  $v_i^{t+1}$  using equation (7.1)
  Calculate new locations  $x_i^{t+1} = x_i^t + v_i^{t+1}$ 
  Evaluate objective functions at new locations  $x_i^{t+1}$ 
  Find the current best for each particle  $x_i^*$ 
  end for
  Find the current global best  $g^*$ 
  Update  $t = t + 1$  (pseudo time or iteration counter)
end while
Output the final results  $x_i^*$  and  $g^*$ 

```

---

Figure 2. Pseudo code of Particle Swarm Optimization [7]

## 2.2 Implement the Simulated PSO using MATLAB

The simulated PSO is implemented by MATLAB code [9] and displays the following output.

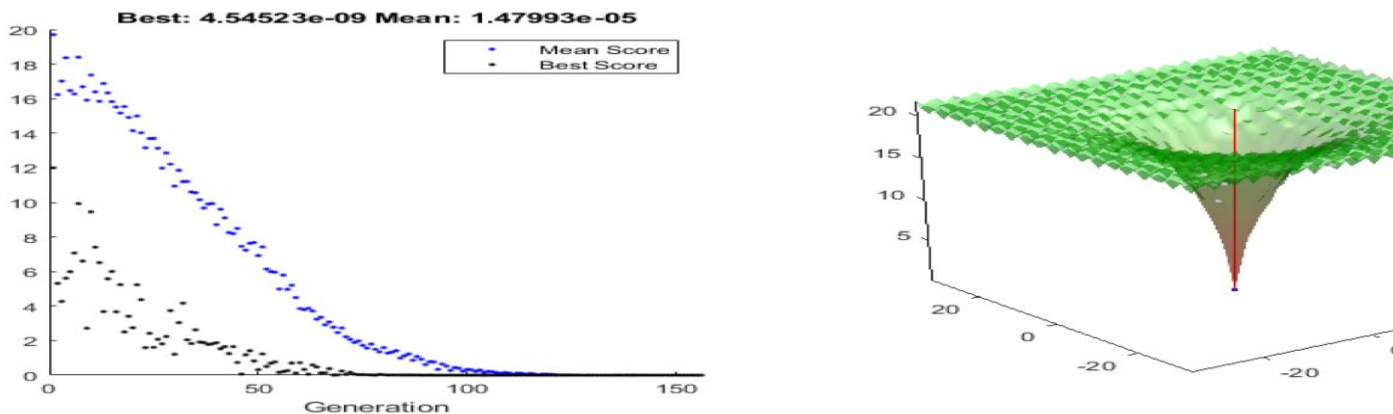


Figure 3. The PSO algorithm searches the space of the “ackleysfcn” objective function and reaches the convergence (generation = iteration)

## 2.3 Accelerated PSO (APSO)

The standard particle swarm optimization (PSO) uses both the current global best  $g^*$  and the individual best  $x_i^*(t)$ . However, there is no compelling reason for using the individual best unless the optimization problem of interest is highly nonlinear and multimodal.

A simplified version that could accelerate the convergence of the algorithm is to use only the global best. This is the so-called *accelerated particle swarm optimization* (APSO) which was developed by Xin-She Yang in 2008 [7]. A further improvement to the accelerated PSO is to reduce the randomness as iterations proceed. That is to use a monotonically decreasing function such as

$$\alpha = \alpha_0 e^{-\gamma t}, \quad (1)$$

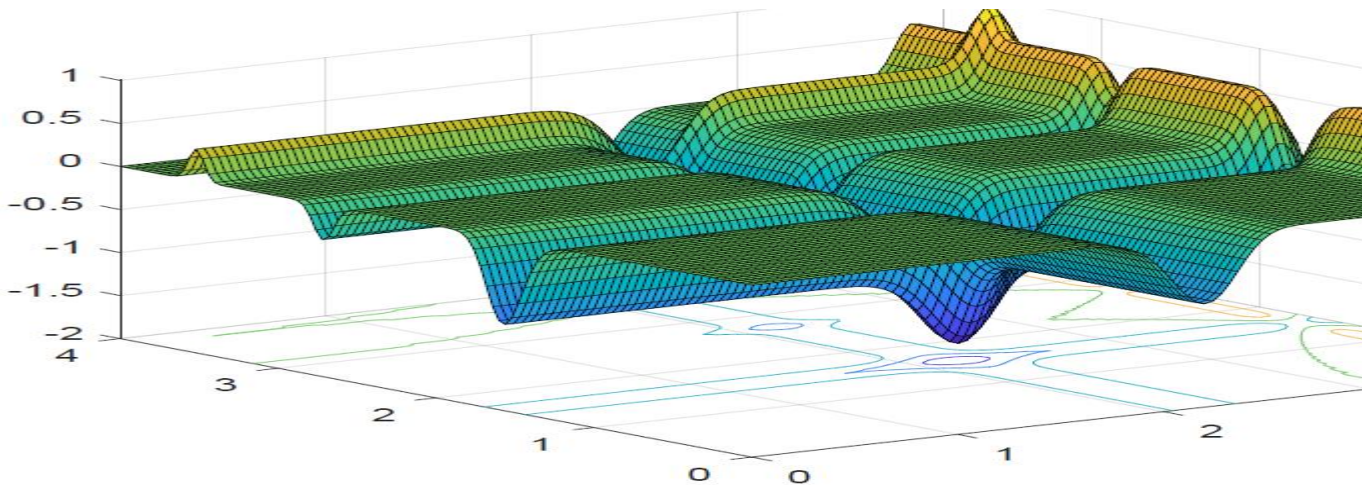
or

$$\alpha = \alpha_0 \gamma^t, \quad (0 < \gamma < 1), \quad (2)$$

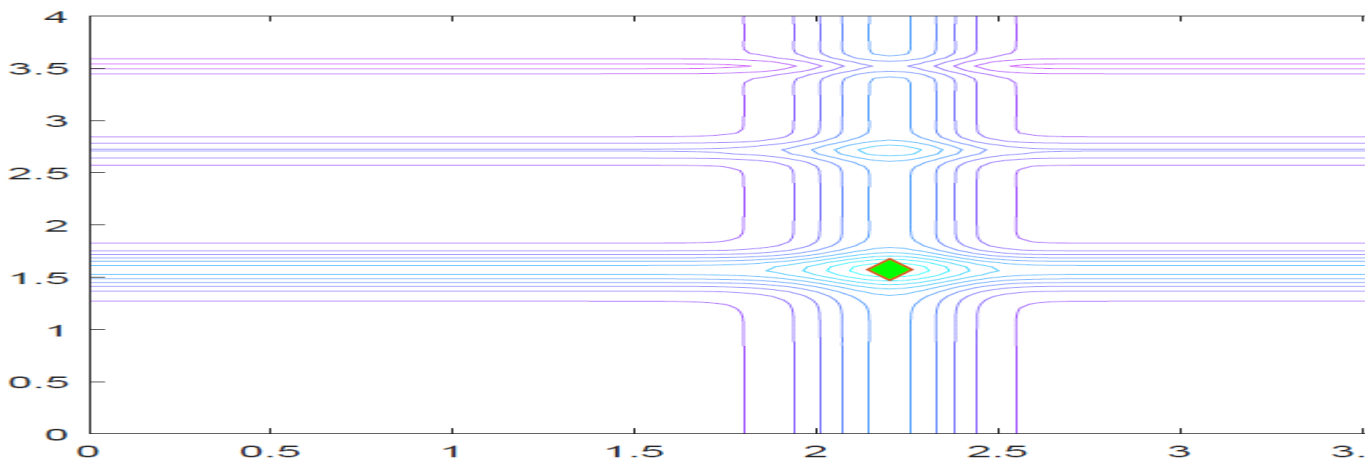
where  $\alpha_0 = 0.5$  to  $1$  is the initial value of the randomness parameter. Here  $t$  is the number of iterations or time steps.  $0 < \gamma < 1$  is a control parameter.

## 2.4 APSO implementation

The implementation of the APSO was simulated using MATLAB code [10]. The total number of particles = 10 and that of iterations = 10. The 2D “Michalewicz” objective function is used. Two outputs are shown below.



**Figure 4.** Michalewicz objective function with the global optimality at (2.20319, 1.57049).



**Figure 5.** Initial and final locations of 20 particles after 10 iterations.

## 2.5 APSO implementation results for speech enhancement

The proposed algorithm does not use individual best position for updating the new locations and is capable of finding the solution in minimum number of iterations. Further, the proposed APSO algorithm is converging faster compared to SPSO algorithm [3].

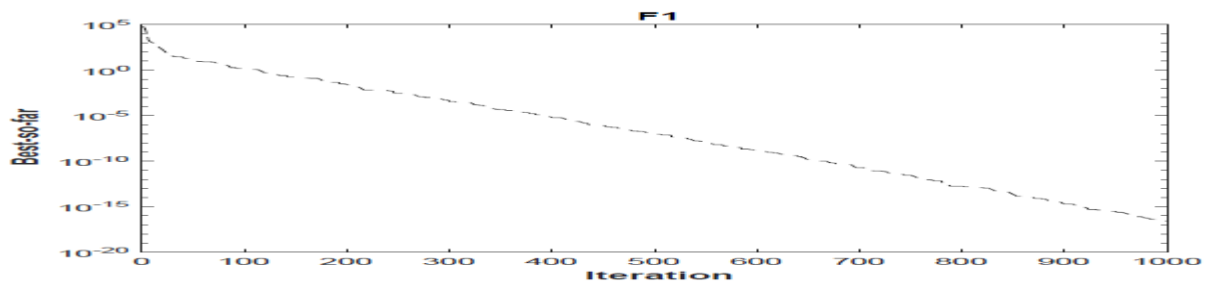
## 3. Gravitational Search Algorithm (GSA) for Speech Enhancement

### 3.1 Gravitational Search Algorithm (GSA)

Gravitational search algorithm (GSA) is an optimization algorithm based on the law of gravity and mass interactions. This algorithm is based on the Newtonian gravity: "Every particle in the universe attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them" [11].

### 3.2 Gravitational Search Algorithm (GSA) Implementation

The implementation of the GSA was simulated using MATLAB code [11]. This experiment uses Function 1 to calculate the objective function. The output is shown below.



**Figure 6.** This implementation of GSA uses Function 1 to compute the value of objective function

### 3.3 GSA implementation results for speech enhancement [3]

The simulation results concluded that the performance of GSA algorithm is better when compared to SPSO with respect to the speech quality and intelligibility.

Table 2. Comparison of quality objective measure PESQ

Input SNR for babble noise condition (dB)	Algorithm	PESQ scores
5	SPSO	2.0455
	GSA	2.1772
0	SPSO	2.2287
	GSA	2.3312
-10	SPSO	1.6485
	GSA	1.6697

Table 3. Comparison of intelligibility objective measure FAI

Input SNR for babble noise condition (dB)	Algorithm	FAI scores
5	SPSO	0.1602
	GSA	0.1735
0	SPSO	0.1419
	GSA	0.1545
-10 dB	SPSO	0.0689
	GSA	0.0834

## 4. Hybrid PSOGSA for Speech Enhancement

### 4.1 PSOGSA

The objective measures that improved SNR, PESQ, FAI and WSS have got improvement in hybrid PSOGSA algorithm when compared to the conventional GSA and standard PSO algorithms. And it can be concluded that PSOGSA can more effectively reduce the background noise of the noisy input speech [3]. The main idea of PSOGSA is to integrate the ability of exploitation in PSO with the ability of exploration in GSA to synthesize both algorithms' strength. Some benchmark test functions are used to compare the hybrid algorithm with both the standard PSO and GSA algorithms in evolving best solution [12].

### 4.2 PSOGSA Implementation

The implementation of the GSA was simulated using MATLAB code [13]. This experiment uses Function 1 to calculate the objective function. The output is shown below.

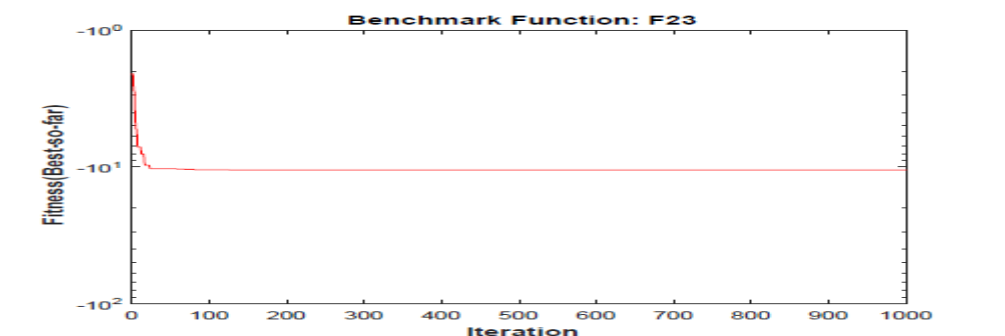


Figure 7. This implementation of PSOGSA uses Function 23 to compute the value of objective function

### 4.3 PSO-GSA Implementation Results for Speech Enhancement

The SNR levels of the input noisy signal for babble-type noise is set at  $-10$ ,  $-5$ ,  $0$  and  $5$  dB. Results are averaged over 10 trail runs. The performance of the algorithms is evaluated by computing SNRI measure, and the results are tabulated. There is  $10$  dB improvement in PSO-GSA when compared with that of SPSO algorithm. At all other noise input SNR levels of noisy speech ( $-10$ ,  $0$  and  $5$  dB), there is almost  $6$  dB improvement in SNRI with PSO-GSA when compared with that of SPSO. Throughout input SNR level, there is almost  $2$  dB improvement in PSO-GSA when compared with that of conventional GSA. The performance of the algorithms is also evaluated by computing intelligibility objective measure FAI [3],

## 5. Bat Algorithm (BA) for Speech Enhancement

### 5.1 Bat Algorithm (BA)

Bat algorithm (BA) is one of the recently proposed heuristic algorithms imitating the echolocation behavior of bats to perform global optimization. The superior performance of this algorithm has been proven among the other most well-known algorithms such as genetic algorithm (GA) and particle swarm optimization (PSO) [14, 15].

### 5.2 Bat Algorithm (BA) Implementation

The original version of this algorithm is suitable for continuous problems, so it cannot be applied to binary problems directly. Here [14, 15], a binary version of this algorithm is available.

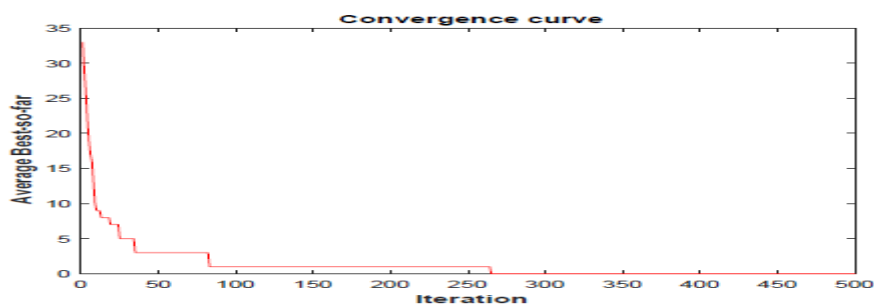


Figure 8. Implementation of the Binary version BA

### 5.3 Bat Algorithm (BA) Implementation results for speech enhancement

Other related simulation discussion on BA [3]: BA has yielded best solutions to the problem of adaptive filtering in speech enhancement by providing a dynamic balance between the exploration and exploitation of search space with an automatic zooming effect. With respect to quality and intelligibility of enhanced speech, simulation results proved that the proposed



BA-based enhancement algorithm is superior to SPSO-based enhancement algorithm and all other algorithms proposed in this research work.

Bat algorithm (BA) is one of the recently proposed heuristic algorithms imitating the echolocation behavior of bats to perform global optimization. The superior performance of this algorithm has been proven among the other most well-known algorithms such as genetic algorithm (GA) and particle swarm optimization (PSO). This version of this algorithm can be applied to binary problems directly. This study run the BA simulation successfully which was executed under MATLAB R2018a and the simulation output is displayed as follows:

## 6. Conclusions

Since the implementations of the four metaheuristic optimization algorithms (APSO, GSA, PSOGSA, and BA) are well done, which provides a good base for more advanced study in the near future, such as to create a much powerful algorithm by making a variant of a current algorithm or combining multiple algorithms' advantages.

Besides, with the work presented in [3], it is proved that the performances of the proposed algorithms are compared with the performance of the existing standard PSO based speech enhancement approach. From the results it is observed that each of the proposed algorithms achieved better performance when compared with that of standard PSO based speech enhancement approach with improved speech quality and intelligibility scores.

## References

- [1] Poonam, S. Ratnoo, Gravitational Search Algorithms in Data Mining: A Survey, IJARCCCE ISSN (Online) 2278-1021 ISSN (Print) 2319 5940 International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified Vol. 6, Issue 6, June 2017.
- [2] K. Prajna, G. S. B. Rao, K. V.V. S. Reddy, R. U. Maheswari, A New Dual Channel Speech Enhancement Approach Based on Accelerated Particle Swarm Optimization (APSO), *I.J. Intelligent Systems and Applications*, 2014, 04, 1-10 Published Online March 2014 in MECS (<http://www.mecs-press.org/>) DOI: 10.5815/ijisa.2014.04.01.
- [3] P. Kunche and K.V.V.S. Reddy, *Metaheuristic Applications to Speech Enhancement*, Springer Briefs in Speech Technology, Springer International Publishing AG Switzerland, 2016.
- [4] X.-S. Yang, Bat Algorithm: Literature Review and Applications, J. Bio-Inspired

- Computation, Vol. 5, No. 3, pp. 141–149 (2013). DOI: 10.1504/IJBIC.2013.055093.
- [5] M. P. KUMAR, R. P. DAS, Speech Enhancement Analysis using PSO and ANFIS Methods, International Journal of Scientific Engineering and Technology Research (IJSETR), ISSN 2319-8885 Vol.05, Issue.49, December-2016, Pages:10059-10065.
- [6] S. Mirjalili, S. M. Mirjalili, X. Yang, Binary Bat Algorithm, Neural Computing and Applications, In press, 2014, Springer DOI:<http://dx.doi.org/10.1007/s00521-013-1525-5>
- [7] X.-S. Yang, Nature-Inspired Optimization Algorithms, Elsevier Inc., 2014.
- [8] J. Kennedy, RC Eberhart. Particle swarm optimization. In: Proceedings of the IEEE international conference on neural networks, Piscataway, NJ, USA; 1995. p. 1942–48.
- [9] A. Cirillo,  
<https://www.mathworks.com/matlabcentral/fileexchange/30660-simple-example-of-pso-algorithm>.
- [10] X-S. Yang,  
<https://www.mathworks.com/matlabcentral/fileexchange/29725-accelerated-particle-swarm-optimization>.
- [11] E. Rashedi, H. Nezamabadi-pour and S. Saryazdi, GSA: A Gravitational Search Algorithm,  
<https://www.mathworks.com/matlabcentral/fileexchange/27756-gravitational-search-algorithm-gsa?focused=5194754&tab=function>
- [12] Paper: A New Hybrid PSOGSA Algorithm for Function Optimization, in IEEE International Conference on Computer and Information Application (ICCIA 2010), China, 2010, pp.374-377, DOI: <http://dx.doi.org/10.1109/ICCIA.2010.6141614>
- [13] S. Mirjalili,  
<https://www.mathworks.com/matlabcentral/fileexchange/35939-hybrid-particle-swarm-optimization-and-gravitational-search-algorithm-psogsa>
- [14] S. Mirjalili,  
<https://www.mathworks.com/matlabcentral/fileexchange/35939-hybrid-particle-swarm-optimization-and-gravitational-search-algorithm-psogsa>
- [15] S. Mirjalili, S. M. Mirjalili, X.-S. Yang, Binary Bat Algorithm, Neural Computing and Applications, In press, 2014, Springer DOI:<http://dx.doi.org/10.1007/s00521-013-1525-5>