# Advanced Gravitational Search Algorithm with Modified Exploitation Strategy

Talha Ali Khan, Sai Ho Ling and Ananda Sanagavarapu Mohan

*Abstract*— **Gravitational search algorithm (GSA) is a novel technique as compared to other heuristic methods and depends upon the gravitational forces between masses. It showed better performance in terms of convergence but has slow exploitation ability due to the fitness function effect on masses; they are getting heavier after every iteration. Therefore, masses are getting closer to each other and nullify the gravitational forces on each other avoiding them from swiftly exploiting the optimum. In order to solve this problem in this paper, an advanced gravitational search algorithm (AGSA) with modified exploitation strategy is proposed. The reason for the modification is that the agents will reach the optimum point swiftly and the convergence is much faster as compared to the standard and other improved versions of GSA available in the literature. AGSA is also compared with the standard and modified Particle Swarm Optimization algorithm in this paper. Five benchmark functions have been implemented to assess the efficiency of the presented algorithm. In addition, a standard, constrained, design problem of a pressure vessel design is also used to examine the efficiency of the proposed technique. Simulation results empirically validated that the presented algorithm has remarkably better results in accordance with convergence and solution stability when compared to the other methods.**

## I. INTRODUCTION

Among the different nature-inspired algorithms, GSA is a novel technique that can solve advance and difficult problems. It depends upon gravity law and the law of motion [1] and comes under a population-based technique that are having different masses. Depends upon the gravitational force, agents are giving information directly for undeviating the search and to explore the finest solution in the search area [2]. GSA that is inspired and depends upon the laws of physics has improved performance and characteristics than any nature inspired or bio-inspired algorithms such as particle swarm optimization, cat swarm, Harmony search and others [3].

In GSA, each agent is replicated as a matter and the problem search area as the universe where they are experiencing gravitational force. The Einstein general theory of relativity states that the gravitational field is demonstrated as a curvature of space-time. Due to this reason, many opportunities are still in this area of research to utilize the concepts of gravity and introduce new search operators. As a result, GSA is evaluated as a physics-based metaheuristic search algorithm and population-based too. Newton's second law of motion defines some operators for GSA are agents' movement, mass allotment and calculation of the force acting on the objects. Since distance and mass influence the force of gravity, the agents collaborate and content through gravity. Dependence on the distance, the summation of the gravitational forces and the relationship between masses and fitnesses make this technique inimitable. To maintain stability between exploitation and exploration a new strategy in the velocity is proposed in this paper.

In AGSA, different agents are applying forces simultaneously on one another and therefore, the step-size is comparatively large. Due to this larger step size movement, the exploration of the AGSA is good. However, the bigger step size leads to poor exploitation in the final stage. As a result, the algorithm suffers from poor solution stability, convergence, and less accuracy. In order to solve this problem and to develop a bridge between exploration and exploitation, a new term is introduced in the velocity update equation in (11) with an exponentially decreasing value as the number of iterations is increasing; this helps to decrease the step size and thus improving exploitation.

Talha Ali Khan, Steve Ling and Ananda Mohan are with School of Biomedical Engineering in University of Technology Sydney, Australia.

## II. FUNDAMENTAL CONCEPTS OF GRAVITY LAW AND AGSA

Amongst the four other basic interactions in nature, one of the basic property is the force of gravity that is the propensity to accelerate the masses towards themselves. The gravitational force causes the particles to attract each other towards itself. The inexorableness and unavoidability of the gravitational force create it unique among the different natural forces. Newton has described the force of gravity as an action from a distance that explains gravity acts upon different distanced particles without interruption and without any mediator. Newton Law of gravity is defined as "A gravitational force enforced each particle to attract every other particle in this universe. This gravitational force $F$ is inversely proportional to the square of the distance they are separated from each other and directly proportional to the product of their masses [4]:

$$F = G\frac{M_a\ M_b}{R^2} \tag{1}$$

where $G$ is a gravitational constant and its value is $6.6738 \times 10^{-11}$ N·m²/kg², $M_a$ is the mass of the first particle and $M_b$ mass of the second particle, $R$ is the distance between the two particles they are separated from each other and $F$ is the force of gravity. Also, the second law of Newton describes as while force is exerted acceleration $a$ is produce that is dependent on its mass "$M$" and this force [5]:

$$a = \frac{F}{M} \tag{2}$$

Due to the force of gravity particles of the universe attract all the other particles based on the principles (1) and (2). The influence of larger and nearer particles are greater. Therefore, if the distance increases the force of gravity also decreases between them. Moreover, due to the reducing in the force of gravity, the definite value of the "*G*" is based on the real time of life of the universe. (3) shows the curtailment of the "*G* "with the time:

$$G(t) = G(t_0) * \left(\frac{t_0}{t}\right)^{\beta} \quad , \beta < 1 \tag{3}$$

where at time *t*, $G(t)$ is the value of the gravitational constant and $G(t_0)$ is the value of the gravitational constant at the first cosmic Quantum-interval of time ($t_0$), as $\beta<1$ value of *G* is larger initially. It facilitates the exploration approach and ensures a swift convergence. To find out the whole search area in AGSA and at the end of the search exploitation fade in and exploration fade out for the faster convergence *G* asymptotically inclines to $G(t_0)$. The $\beta$ is used to adjust the value of *G*. For the same $G(t_0)$ a smaller value of $\beta$ guarantees a larger value of *G* initially. Thus, the convergence of GSA is controlled [6]. Value of $\beta=0.1$.

Theoretic physics defines masses in three different types these are as follows:

A. *Inertial mass*

When a force is applied, Inertial mass "$M_i$" is the amount of the object's confrontation to change its state of motion. Heavier inertial mass object varies its state of motion swiftly than an object with less inertial mass gradually varies its state of motion.

*B. Passive gravitational mass*

Passive gravitational mass "$M_p$" is the amount of the strength of an object's interaction with the gravitational field. The gravitational field is weak for smaller "$M_p$", as compared to the object with heavy "$M_p$".

*C.* Active gravitational mass

Active gravitational mass "$M_a$" is the amount of the strength of the gravitational field because of an individual object. For smaller values of "$M_a$", the gravitational field of an object is weak as compared to the object with larger values of "$M_a$".

According to the above characteristics, Newtown's Law of Gravity can be modified as follow:

$F_{ij}$ is the gravitational force that applied by mass *j* on mass *i* which is inversely proportional to the square of the distance between them and directly proportional to the product of the "$M_{pi}$" and "$M_{aj}$". "$a_i$" is inversely proportional to inertia mass of *i* and proportional to $F_{ij}$. Precisely (1) and (2) can be modified as follows:

$$F_{ij} = G\,\frac{M_{aj}\,M_{pi}}{R^2} \tag{4}$$

$$a_{ii} = \frac{F_{ij}}{M_{ii}} \tag{5}$$

In spite of the fact $M_a$, $M_p$ and $M_i$ are theoretically definite; so far, none of the observations have demonstrated any absolute dissimilarity amidst them. The general relativity's theory relies upon the supposition stating that passive gravitational mass and inertial mass are the same. This phenomenon is acknowledged as the weak equivalence principle.

## III. ADVANCED GRAVITATIONAL SEARCH ALGORITHM

In this method, the mass of the agents/particles is considered as the criteria of their performances. Objects (agents) are attracted to each other by the gravitational force that is responsible for the movement of these agents globally in the direction of the heavier masses agents. The objects having heavyweight matches with best results of the given problem. Since AGSA depends upon the physics laws, each object has four properties these are: "$M_p$", "$M_i$", "$M_a$" and position. The Problem solution is determined by the position of the object and by using a fitness function; inertial and gravitational masses are calculated. Gradually, objects (masses) are attracted towards the objects with heavier mass. The pseudo code of AGSA is as follows:

***Pseudo Code of AGSA***

```
Initialization
for i= 1 to N
  for d= 1 to D
    initialize  X_i = ( X_i^1, X_i^2, ……………… X_i^d, X_i^N )
    initialize velocity
next d
compute the fitness of each agent
next i
for t= 1 to T
select best agent and record position
update  G(t) = G(t_0) * (t_0/t)^β
calculate each mass , update  X^w (t) and X^b (t)
for i= 1 to N
for d= 1 to D
calculate F_i^d(t) = Σ_{j∈K_best, j≠i} rand_j F_ij^d (t)
calculate acceleration  a_i^d(t) = F_i^d(t) / M_ii(t)
calculate position  X_i^d(t+1) = X_i^d + v_i^d(t+1)
calculate velocity  V_i^d(t+1) = (rand_i * v_i^d(t)+a_i^d(t)) * (1 - t/fE)^α
next d
compute the fitness value of the i^th agent
next i
next t
```

Let suppose a system of the objects with *L* objects, the $i^{th}$ object position can be denoted as:

$$X_i = \left( X_i^1, X_i^2, \ldots\ldots\ldots\ldots X_i^d, X_i^N \right) for\ i = (1,2,..N) \tag{6}$$

In the $d^{th}$ dimension, *N* represents the dimension of search space and $x_d$ shows the position of the $i^{th}$ object.

The force exerting on mass "*i*" from mass "*j*" at time *t* is represented as:

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t) * M_{aj}(t)}{R_{ij}(t) + \varepsilon}\ \left(X_j^d(t) - X_i^d(t)\right) \tag{7}$$

where $R_{ij}(t)$ is the distance between two objects *i* and *j*, $M_{pi}$ is the passive gravitational mass related to object *i*, $G(t)$ is a gravitational constant at time *t*, $\varepsilon$ is a small constant and $M_{aj}$ is an active gravitational mass of object *j*.

Let's suppose that the total force acted on object *i* in dimension *d* is a random weighted sum of all the components of the forces applied on the other objects then (7) can be rewritten as

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} rand_j\, F_{ij}^d(t) \tag{8}$$

At time *t*, the acceleration of the object *i* by the law of motion in the direction *d* can be represented as:

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \tag{9}$$

For the $i^{th}$ object, $M_{ii}$ represents the inertial mass. The velocity of the masses depends upon their present velocity as well as their acceleration. The updated object is given in (10) and (11) is the proposed velocity of the agent for next time *t*:

$$X_i^d(t+1) = X_i^d + v_i^d(t+1) \tag{10}$$

$$V_i^d(t+1) = (rand_i * v_i^d(t) + a_i^d(t)) * \left(\mathbf{1} - \frac{t}{fE}\right)^{\alpha} \quad (11)$$

where $rand_i$ is a uniform random number within [0-1], $t$ is current iteration, $fE$ is function evaluation which is the stopping criteria and $\alpha$ is an integer. The reason for choosing smaller values of $\alpha$ that it results in a faster convergence of the algorithm. Search accuracy is monitor by the gravitational constant $G$ and its value reduces with the passage of time. Inertial mass and the Gravitational constant will be calculated by the fitness evaluation. An agent with more mass is an efficient agent. That implies that agents with heavy masses have higher attractions and they move gradually. Assuming all the masses are same so the inertial mass and the gravitational masses can be updated as follows:

$$M_{ai} = M_{ii} = M_{pi} = M_i \qquad i = 1,2,3, \dots \dots \dots N \quad (12)$$

$$m_i(t) = \frac{fit_i(t) - X^w(t)}{X^b(t) - X^w(t)} \quad (13)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \quad (14)$$

At time $t$, $fit_i(t)$ is the fitness value of the object.

*Worst* $X^w(t)$ and best $X^b(t)$ for a maximization problem are defined as:

$$X^b(t) = max_{j\,\varepsilon\,\{1,\dots N\}} fit_j(t) \quad (15)$$

$$X^w(t) = min_{j\,\varepsilon\,\{1,\dots N\}} fit_j(t) \quad (16)$$

For the minimization problem, (15) and (16) are presented as follows:

$$X^b(t) = min_{j\,\varepsilon\,\{1,\dots N\}} fit_j(t) \quad (17)$$

$$X^w(t) = max_{j\varepsilon\,\{1,\dots N\}} fit_j(t) \quad (18)$$

By decreasing, the number of agents with the passage of time a balance can be obtained between exploitation and exploration. Therefore, simply the objects with heavier masses applying forces to the other objects are considered. A set of the agents that are known as Best agents "$K_{best}$" having heavier masses are applying forces and attract the other agents in the search space. Hence, the value of the $K_{best}$ is decreasing gradually as the iteration progresses until simply one object is applying force to the other agents. Therefore, (8) is modified as follows:

$$F_i^d(t) = \sum_{j\varepsilon K_{best}, j\neq i} rand_j\, F_{ij}^d(t) \quad (19)$$

## IV. RESULTS AND ANALYSIS

Five benchmark functions are applied to evaluate and compare the efficiency between the proposed AGSA, SPSO, MPSO and other methods available in the literature.

### *A.Benchmark Functions*

The consistency, effectiveness, and performance tests of various optimization techniques are outlined by using different benchmark functions and standards. This is one of the methods by which the convergence, stability and solution quality are calculated. For the performance evaluation of the presented technique, the following benchmark functions are used. They are as follows:

*1) Sphere:*

The sphere is a unimodal function that is the symmetrical model with a single minimum. The key idea of using this function for testing is to find out the convergence rate of searching. It is most likely the commonly used benchmark function.

$$f_1(x) = \sum_{i=1}^{D} x_i^2 \quad (20)$$

*2) Ackley:*

Ackley is a multi-modal function with many local minima.

$$f_2(x) = -20exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right) + 20 + e \quad (21)$$

*3) Rastrigin:*

Rastrigin is also a multi-modal function with many local minima.

$$f_3(x) = \sum_{i=1}^{D}[x_i^2 - 10\cos(2\pi x_i) + 10] \quad (22)$$

*4) Griewank*

It is a multi-modal function with many local minima therefore; it is inclined to convergence in the wrong direction.

$$f_4(x) = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (23)$$

*5) Rosenbrock:*

It is a unimodal function with a single minimum. It is intensely nonseparable. Also, the optimal point is situated in a very narrow ridge. The tip of the ridge is very sharp, and it runs around a parabola.

$$f_5(x) = \sum_{i=1}^{D-1}[100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2] \quad (24)$$

### *B. Simulation Setup:*

The performance of the proposed AGSA, MPSO, the Particle Swarm Optimization Cuckoo Search Paralleled Algorithm (PSOCSPA) [7] , Standard Gravitational Search Algorithm (SGSA),Local Exploitation Based Gravitational search Algorithm (LEGSA) [8], Improved Particle Swarm Optimization Algorithm (IPSO) [9] and Plane Surface Gravitational Search Algorithm (PSGSA) [10] are evaluated on the five benchmark test functions.

The following simulation conditions for the algorithms used are as follows:

- No of Particles=30;
- Acceleration Constant $C_1 = 2.1$;
- Acceleration Constant $C_2 = 1.9$;
- Dimension D=30;
- $\alpha = 2$

### *C. Statistical Analysis:*

The results for the AGSA, MPSO are compared with the SPSO, SGSA and other variations of the improved PSO and GSA algorithms. The results are compared in terms of standard deviation and mean as shown in Table I.

One of the other main criteria to check the efficiency of the presented method and compare it with different techniques is to determine the *t*-test value. It is calculated by taking the standard deviation and mean values from the two algorithms. The positive t value corresponds to the good performance of the first algorithm as compared to the second algorithm and vice versa. Three terms are used in measuring the t-values: $\xi$ value of the degree of freedom, $\alpha_1$ and $\alpha_2$ are the mean values and, $\sigma_1$ and $\sigma_2$

are the values for the standard deviation of the two algorithms. The t-value can be measured as:

$$t = \frac{\alpha_1 - \alpha_2}{\sqrt{\left(\frac{\sigma_1^2}{\xi+1}\right) + \left(\frac{\sigma_2^2}{\xi+1}\right)}} \quad (25)$$

If the t-value is larger than 1.645, which means the first method has better performance than the second technique by 95%. The t-values between the AGSA, MPSO, SPSO and various other versions of PSO and GSA methods are shown in Table II. It is noticeable that the t-value for the proposed algorithm is higher than 1.645 for most of the functions.

### *D. Graphical Analysis*

From Fig. 1 to Fig. 5, the comparison between the SPSO, AGSA, SGSA, and MPSO is presented. It is clear from the figures that the AGSA gives better performance than other algorithms and shows faster convergence for all tested benchmark functions except Rosenbrock

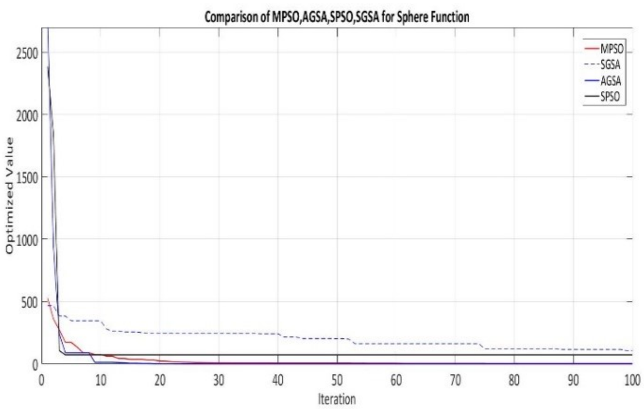


Fig 1. A comparison between MPSO, SPSO, GS, and AGSA for Sphere Function.

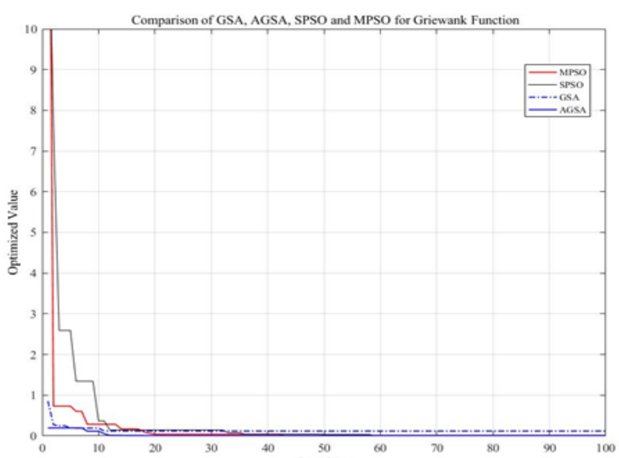


Fig. 2. A comparison between MPSO, SPSO, GSA, and AGSA for Griewank Function

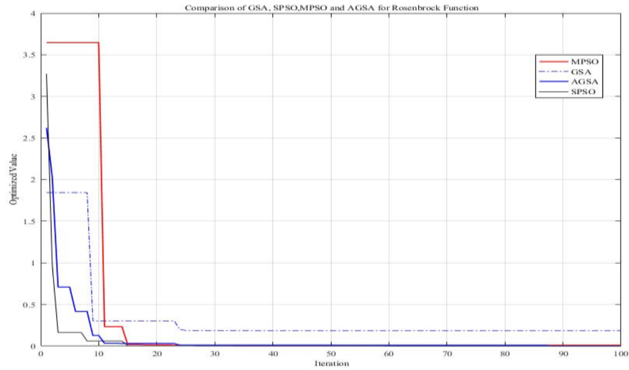


Fig. 3. A comparison between MPSO, SPSO, GSA, and AGSA for Rosenbrock Function

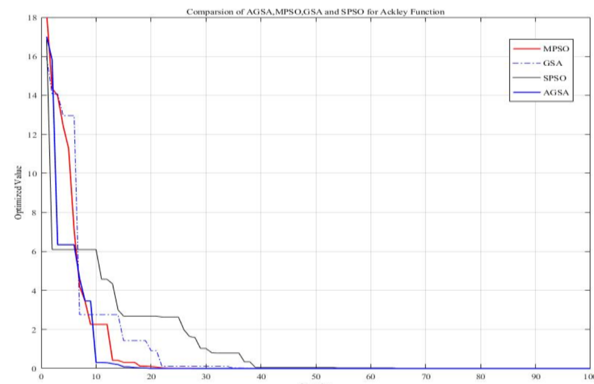


Fig. 4. A comparison between MPSO, SPSO, GSA, and AGSA for Ackley Function

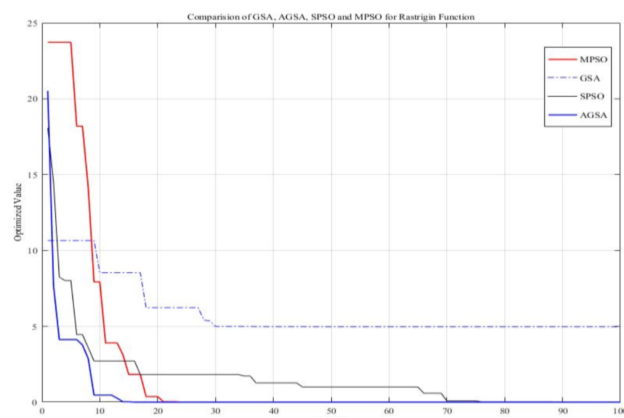


Fig. 5. A comparison between MPSO, SPSO, GSA, and AGSA for Rastrigin Function

TABLE I. The standard deviation and mean evaluation between AGSA and other algorithms
(All Results Are Averaged (Rank: 1—Best, 8—Worst

| Functions | | AGSA | MPSO | SPSO | SGSA | IPSO | PSOCSPA | PSGSA | LEGSA |
|---|---|---|---|---|---|---|---|---|---|
| $f_1(x)$ | Mean | $6.6e^{-172}$ | $1.50e^{-142}$ | 0.0057 | $9.01e^{-06}$ | $1.81e^{-52}$ | $1.33e^{+01}$ | $1.04\ e^{-02}$ | $8.81109e^{-06}$ |
| | Std. Dev | $5.21e^{-172}$ | $3.21e^{-142}$ | 0.0178 | $7.2e^{-07}$ | $3.52e^{-52}$ | $7.30e^{+01}$ | $1.91\ e^{-02}$ | $1.0968e^{-06}$ |
| | Rank | **<u>1</u>** | 2 | 6 | 4 | 3 | 8 | 7 | 5 |
| $f_2(x)$ | Mean | $1.13e^{-20}$ | $7.21e^{-14}$ | 0.0209 | $9.4e^{-06}$ | $7.99e^{-15}$ | $3.841e^{-06}$ | $1.48\ e^{-02}$ | 0.00956612 |
| | Std. Dev | $7.16e^{-20}$ | $2.34e^{-14}$ | 0.629 | $3.2e^{-07}$ | $1.58e^{-15}$ | $5.964e^{-06}$ | $2.38\ e^{-02}$ | 0.00037639 |
| | Rank | **<u>1</u>** | 2 | 8 | 4 | 3 | 5 | 7 | 6 |
| $f_3(x)$ | Mean | $4.64e^{+01}$ | 3.634 | $1.94e^{+2}$ | $8.77\ e^{1}$ | $4.57e^{1}$ | 0.0 | $2.13\ e^{+01}$ | - |
| | Std. Dev | $8.30e^{+00}$ | 14.467 | $1.08e^{+2}$ | 9.51 | $1.23e^{1}$ | 0.0 | 5.34 | - |
| | Rank | **<u>4</u>** | 2 | 7 | 6 | 5 | 1 | 3 | 8 |
| $f_4(x)$ | Mean | $9.03e^{-04}$ | 0.0063 | $1.35e^{2}$ | 0.7589 | $7.63e^{-03}$ | $9.696e^{-07}$ | $1.01\ e^{+01}$ | - |
| | Std. Dev | $3.07e^{-04}$ | 0.02475 | $2.23e^{2}$ | $2.53e^{-16}$ | $8.69e^{-03}$ | $3.060e^{-6}$ | 2.47 | - |
| | Rank | **<u>2</u>** | 4 | 7 | 5 | 3 | 1 | 6 | 8 |
| $f_5(x)$ | Mean | $1.67e^{-03}$ | 2.465 | $2.30e^{+3}$ | 7.99 | $8.01e^{+01}$ | - | $5.53\ e^{+01}$ | - |
| | Std. Dev | $1.81e^{-04}$ | 8.452 | $2.12e^{+3}$ | 0.99 | $6.06e^{+01}$ | - | $6.22\ e^{+01}$ | - |
| | Rank | **<u>1</u>** | 2 | 6 | 3 | 5 | 7 | 4 | 8 |
| Overall Ranking (Average Ranking Number) | | **<u>1</u>** (1.8) | **2** (2.4) | **6** (6.8) | **4** (4.4) | **3** (3.8) | **4** (4.4) | **5** (5.4) | **7** (7.0) |

TABLE II The t-values comparison between AGSA and other algorithms

| Functions | t-value between AGSA and SPSO | t-value between AGSA and MPSO | t-value between AGSA and SGSA | t-value between AGSA and PSGSA | t-value between AGSA and LEGSA | t-value between AGSA and ISPO | t-value between AGSA and PSOCSPA |
|---|---|---|---|---|---|---|---|
| $f_1(x)$ | 2.30 | 3.32 | 88.48 | 3.85 | 56.78 | 3.63 | 1.28 |
| $f_2(x)$ | 0.0234 | 21.78 | 207.7 | 4.39 | 181.66 | 35.72 | 4.39 |
| $f_3(x)$ | 9.63 | 18.13 | 23.13 | -17.98 | - | -.0333 | -3.95 |
| $f_4(x)$ | 4.28 | 1.54 | - | 28.91 | - | 5.47 | -20.77 |
| $f_5(x)$ | 7.67 | 2.06 | 57.05 | 6.28 | - | 9.34 | - |

## V. PROBLEM FOR PRESSURE VESSEL DESIGN

To illustrate the performance of the proposed AGSA, the problem for pressure vessel designing is presented. Fig. 6 illustrates a cylindrical pressure vessel being covered at both ends by spherical heads. The motivation is to decrease the overall cost, together with manufacturing and the material costs

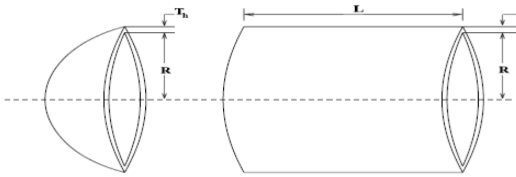


Fig. 6. Pressure Vessel Schematic

Four design parameters for this problem are inner radius *(r)*, the thickness of the shell ($T_s$), Cylindrical Length of section ($L$) and head thickness ($T_h$). The main purpose is to reduce the cost function. Coello et.al [11] defined these four parameter as $x_1$, $x_2$, $x_3$ and $x_4$ respectively. The cost function is represented as:

$$f(x) = 0.6224\, x_1 x_3 x_4 + 1.7781 x_2 x_3^2 + 3.1661\, x_4 x_1^2 + 19.84\, x_1^2 x_3 \quad (26)$$

The boundary conditions are defined as follows:

$$g_1(x) = -x_1 + 0.0193\, x_3 \le 0 \quad (27)$$

$$g_2(x) = -x_2 + 0.00954 x_3 \le 0 \quad (28)$$

$$g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^2 + 1296000 \le 0 \quad (29)$$

$$g_4(x) = x_4 - 240 \le 0 \quad (30)$$

The evaluation of the proposed and different methods are presented in Table III which shows the best solutions obtained from AGSA, MPSO, SPSO, an improved harmony search algorithm (IHS) [12] and augmented Lagrange multiplier algorithm combined with Powell's method (ALMAP) [13]. The results show that the presented technique gives improved results in terms of the mean and standard deviation of the cost function as compared to the other methods.

TABLE III: Optimal solution for the proposed and other algorithms

| **Variable(s)** | **Optimal Solutions** | | | | |
|---|---|---|---|---|---|
| | **AGSA** | **MPSO** | **SPSO** | **IHS** | **ALMAP** |
| $x_1$ | 1.125 | 1.125 | 1.125 | 1.125 | 1.125 |
| $x_2$ | 0.625 | 0.625 | 0.625 | 0.625 | 0.625 |
| $x_3$ | 57.1235 | 58.2749 | 58.2832 | 58.2901 | 48.97 |
| $x_4$ | 43.0152 | 43.6543 | 43.7864 | 43.6926 | 106.72 |
| $g_1(x)$ | 0.00001 | 0.0000 | 0.000 | 0.0000 | 0.179 |
| $g_2(x)$ | -0.0454 | -0.0675 | -0.06902 | -0.0689 | 0.1578 |
| $g_3(x)$ | -1.1006 | -2.2142 | -2.31629 | -2.0150 | 3.0 |
| $g_4(x)$ | -196.08 | -196.42 | -196.707 | -196.30 | 133.284 |
| $f(x)$ | 7195.649 | 7201.948 | 7217.494 | 7197.73 | 8129.800 |
| **Standard Deviation** | $7.4E^{-04}$ | $3.05E^{-03}$ | $5.04E^{01}$ | N/A | N/A |

## VI. CONCLUSION

An advanced form of the gravitational search algorithm is presented in this paper. The AGSA that is modeled depends upon the mass and gravity law. The algorithm depends upon the concepts of the theory of physics presented by Newton and each member of the search area in AGSA are the masses. AGSA consists of the individual system of masses in which the force of gravity is used to convey the information between masses.

The effectiveness of the presented method is illustrated by using five benchmark functions. Results achieved are analyzed with the Modified PSO, SPSO, SGSA and different other algorithms. It has been observed that the AGSA performed significantly better as compared to the SPSO, MPSO and other improved versions with respect to different parameters such as t-values, mean and standard deviation. Moreover, one other example is also taken into account to verify the implementation of the presented method and the results achieved show better mean and standard deviation as compared to the other algorithms. AGSA converges faster and effectively to the optimal solution. The AGSA also gives better results almost on all the benchmark functions.

Gravitational Search Algorithm is a prospective research topic and still has the capacity for new improvements and variations in the original algorithm that present excellent performance on the different types of problems. GSA families can be developed by determining advanced operators, or by proposing enhanced versions.

## VII. REFERENCES


[1] E. Rashedi, E. Rashedi, and H. Nezamabadi-pour, "A comprehensive survey on the gravitational search algorithm," *Swarm and Evolutionary Computation,* vol. 41, pp. 141-158, 2018/08/01/ 2018.

[2] A. Hatamlou, S. Abdullah, and Z. Othman, "Gravitational search algorithm with the heuristic search for clustering problems," in *2011 3rd Conference on Data Mining and Optimization (DMO)*, 2011, pp. 190-193.

[3] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A Gravitational Search Algorithm," *Information Sciences,* vol. 179, no. 13, pp. 2232-2248, 2009/06/13/ 2009.

[4] F. Wilczek, "Newton rules (for now)," *Nature,* vol. 410, no. 6831, p. 881, 04/19/online 2001.

[5] H. Dingle, "The Laws of Nature*," *Nature,* vol. 153, p. 758, 06/24/online 1944.

[6] K. Pal, C. Saha, S. Das, and C. A. C. Coello, "Dynamic Constrained Optimization with offspring repair based Gravitational Search Algorithm," in *2013 IEEE Congress on Evolutionary Computation*, 2013, pp. 2414-2421.

[7] Y. Xiaodong and C. Zefan, "Particle swarm optimization and cuckoo search paralleled algorithm," in *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, 2017, pp. 2236-2240.

[8] P. Rawal, H. Sharma, and N. Sharma, "A Local Exploitation Based Gravitational search algorithm," in *2017 International Conference on Computer, Communications, and Electronics (Comptelix)*, 2017, pp. 573-579.

[9] M. Alhussein and S. I. Haider, "Improved Particle Swarm Optimization Based on Velocity Clamping and Particle Penalization," in *2015 3rd International Conference on Artificial Intelligence, Modelling, and Simulation (AIMS)*, 2015, pp. 61-64.

[10] D. Pelusi, R. Mascella, L. Tallini, J. Nayak, B. Naik, and A. Abraham, "Neural network and fuzzy system for the tuning of Gravitational Search Algorithm parameters," *Expert Systems with Applications,* vol. 102, pp. 234-244, 2018/07/15/ 2018.

[11] C. A. Coello Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer Methods in Applied Mechanics and Engineering,* vol. 191, no. 11, pp. 1245-1287, 2002/01/04/ 2002.

[12] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Applied Mathematics and Computation,* vol. 188, no. 2, pp. 1567-1579, 2007/05/15/ 2007.

[13] E. Sandgren, "Nonlinear Integer and Discrete Programming in Mechanical Design Optimization," *Journal of Mechanical Design,* vol. 112, no. 2, pp. 223-229, 1990.