



# Novel metaheuristic based on multiverse theory for optimization problems in emerging systems

Eghbal Hosseini<sup>1</sup> · Kayhan Zrar Ghafoor<sup>2,3</sup> · Ali Emrouznejad<sup>4</sup> · Ali Safaa Sadiq<sup>5,6</sup> · Danda B. Rawat<sup>7</sup> 

Accepted: 30 August 2020 / Published online: 11 November 2020  
© Springer Science+Business Media, LLC, part of Springer Nature 2020

## Abstract

Finding an optimal solution for emerging cyber physical systems (CPS) for better efficiency and robustness is one of the major issues. Meta-heuristic is emerging as a promising field of study for solving various optimization problems applicable to different CPS systems. In this paper, we propose a new meta-heuristic algorithm based on Multiverse Theory, named MVA, that can solve NP-hard optimization problems such as non-linear and multi-level programming problems as well as applied optimization problems for CPS systems. MVA algorithm inspires the creation of the next population to be very close to the solution of initial population, which mimics the nature of parallel worlds in multiverse theory. Additionally, MVA distributes the solutions in the feasible region similarly to the nature of big bangs. To illustrate the effectiveness of the proposed algorithm, a set of test problems is implemented and measured in terms of feasibility, efficiency of their solutions and the number of iterations taken in finding the optimum solution. Numerical results obtained from extensive simulations have shown that the proposed algorithm outperforms the state-of-the-art approaches while solving the optimization problems with large feasible regions.

**Keywords** Meta-heuristics · Constrained optimization · Multiverse algorithm (MVA) · Bi-level optimization

## 1 Introduction

Meta-heuristic algorithms can be applied for training neural networks in solving real-life problems, though each algorithm has its own limitations. For instance, there are recently some of the prominent meta-heuristic algorithms that been widely used in optimizing the neural network accuracy include Particle Swarm Optimisation (PSO) [5], Bat Algorithm (BA) [9], FireFly (FF) [12]. However, literature cannot identify a single algorithm to be the best for solving all optimization problems, this has also been proved by the well-known No Free Lunch (NFL) theorem [29]. In this theorem, there was a logical prove supporting the aforementioned claim that there is no such meta-heuristic best suited for solving all types of optimization problems. In another words, there is a group of meta-heuristic algorithms perform the best in solving a set of problems, while the same group might give poor performance in

solving different set of optimization problems. Hence, this NFL theorem has opened the door to researchers on keep developing new algorithms trying to achieve the best solution for different kind of problems. Besides, the challenges of heavy computational cost, existence of hastily convergence, mutation rate, crossover rate, time taken in fitness evaluation chiefs to boost current algorithm or develop new one.

Artificial Neural Networks (ANNs), which are normally used in pattern recognition, computer vision, solving real-world problems (liner or non-liner problems) and classification, are normally need to be trained or optimized using a wide range of meta-heuristics algorithms that mainly classified as a single-solution-based or population-based. Population-based meta-heuristics algorithms are widely used recently due to their ability in cooperatively finding the optimal solution over the course of training process. This kind of algorithms is mainly found as a concept of Swarm Intelligence (SI) that was proposed by [30]. The main process of meta-heuristics' behavior was formulated based on the evolutionary concept of the SI agents.

Evolutionary or nature-inspired algorithms like Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Bat Algorithm (BA), Fire Fly (FF), and Gray Wolfe Optimizer

---

✉ Danda B. Rawat  
db.rawat@ieee.org

(GWO) are widely used in various optimization problems in different fields. For instant, the feature selection is a vital process that mainly reflect on the accuracy of a classification model as well as optimizing the parameters via the use of meta-heuristic algorithms. The feature selection through this process is another extension of distinct research dimension. However, there are still some limitations with the widely used classifiers such as: computationally expensive, high algorithmic complexity, extensive memory requirements, and selection of appropriate kernel parameters, which are mostly tricky. Specifically, when a meta-heuristic algorithm well handles a problem with high accuracy may not produce the same inspiring results for another problem with different requirements.

Well-known meta-heuristic algorithms have been simulated mimicking animals/insects' behavior [1–11, 16–18, 21, 22, 32, 33]. Ant Colony Optimization (ACO) [2] simulated social life of ants, PSO [3] came from behavior of swarm life of animals such as birds, fishes and so on. Other famous meta-heuristics are as follows: Artificial Bee Colony Algorithm [4], krill herd algorithm [5], BA [6], social spider optimization [7], Chicken Swarm Optimization (CSO) [8], firefly algorithm [9], Multi-Verse Optimizer [26], Quantum multiverse optimization algorithm [27], Chaotic multi-verse optimizer [28]. However, the investigation of applying various heuristic algorithms and determine the best result is motivated through the need for the best tuning between the common feature of these algorithms that is the division of the search process into two phases: exploration and exploitation. Searching for appropriate weighing scale between these two phases is still considered as an open challenge job. This is due to the nature of stochastic behavior of meta-heuristics algorithms. One of the very good approaches that recently widely used is the evolutionary or nature-inspired algorithms, which originates from the meta-heuristic search algorithms family motivated by the theories and biological evolution and the actions of swarms of nature's creation. Table 1 shows the brief description of the nature-inspired algorithms.

Similarly, Cyber-physical systems (CPS) have tight integration of computation, communication, and control engineering with physical elements. CPS systems such as medical CPS, transportation CPS and energy CPS can benefit from proper design and optimization techniques. All CPS systems are emerging faster than before due to progress in real-time computing, communications, control, and artificial intelligence. Multi-objective design optimization approaches help to maximize the efficiency, capability, performance, and safety of CPS systems. The proposed approaches in this paper can be applied to above mentioned CPS systems for improved efficiency and performance where time-varying sampling patterns, sensor scheduling, real-time control, feedback scheduling, task and motion planning and resource sharing can be optimized.

Therefore, in this paper a new algorithm is proposed that has ability to solve various kinds of optimization problems. The proposed algorithm has been modelled and simulated by using inspiration of multiverse theory, named Multi-Verse Algorithm (MVA). Computational results show that MVA is successful to propose efficient and feasible solutions for different problems. MVA is constructed from easy conception of multiverse theory, which is implemented using MATLAB platform. MVA algorithm is organized based on initial population, explosion of solutions and principle concepts such as feasible and infeasible regions. Therefore, the algorithm has low computational complexity in comparison with the state of the arts approaches.

The MVA has some new remarkable features as it has been inspired from a scientific theory not based on behavior of animals or insects, which it gives more stable and accurate behavior. Moreover, MVA can solve difficult optimization problems such as bi-level programming problems. The algorithm is extensive according to different kinds of problem in computational results.

The MVA has been made from two conceptions: population of solution and the theory of parallel worlds. The algorithm starts from feasible and infeasible solutions and carries on using main conceptions of multiverse theory. In

**Table 1** Comparison of meta-heuristic algorithms

Algorithm	Approach
Genetic algorithm (GA)	Inspired by the process of natural selection
Particle swarm optimization (PSO)	Mimics Bird Flocking behavior
Ant Colony Optimization (ACO)	Mimics Foraging behaviour and ants colony structure
Artificial Bee Colony Algorithm (ABC)	Mimics foraging behavior of honey bees
Firefly Algorithm (FA)	Inspired by the flashing behavior tropical fireflies
Bat Algorithm (BA)	inspired by echolocation of bats
Grey Wolf Optimizer (GWO)	Inspired by the behavior of Gray Wolves
Multiverse Optimizer (MVO)	Mathematical formulation is based The white/black hole tunnels
The proposed MVA	Inspired from multiverse eruption

fact, details of theory are used in all steps of MVA such as: creation of the initial population, explosion of the solutions (big bangs) and movement of universes to find the optimal solution. MVA is compared with other classic and meta-heuristic approaches. Comparison confirms efficiency of the proposed meta-heuristic algorithm.

The rest of paper is organized as follows: Section 2 presents a source of inspiration of the proposed MVA. Section 3 details out a conceptual design and simulation of the proposed MVA algorithm from the multiverse theory. This is followed by Section 4, which presents computational results of MVA technique. Section 5 shows the comparison of our MVA and existing well-known meta-heuristic algorithms and Section 6 shows convergence behavior of the MVA. Finally, Section 7 concludes the paper.

### 1.1 Key differences of MVA and MVO

The proposed MVA in this paper is completely different with MVO [26]. In particular, the differences are categorized into three main aspects: concepts and inspiration, mathematical formulation, and steps of algorithms. Table 2 summarizes the key differences between the proposed MVA and MVO in [26].

#### 1.1.1 Concepts and inspiration

The main inspirations of MVO [26] are based on: white hole, black hole, and wormhole whereas the main inspirations of our algorithms, MVA, are based on: parallel and different big bangs.

The mathematical models of concepts in MVO are: exploration, exploitation, and local search. The mathematical models of concepts in MVA are: creation of the initial population, explosion of the solutions, and rotation of universes to find the optimal solution. Following concepts of multiverse have never discussed in [26], and this is because inspirations of two algorithms are different:

1. In the multiverse theory all universes come from very small and dense particle. Thus, the proposed MVA is inspired from this idea to create the next population very near to the solutions of initial population.
2. In the next step, all solutions, which were very near to each other are distributed in the feasible region same as big bangs.
3. The best solution of each area will be found and these solutions construct next population. Each area corresponds a universe in multiverse theory.
4. In each iteration, better solutions will be surrounded by more new solutions of the next population. This is because in the multiverse theory universe with more dark energy has more galaxies and planets.
5. Likewise, the proposed algorithm focus on the beginning of the world and converting into the present complexity of the world. In the other words, MVA wants to answer this question: how the world was started and changed by passing time?"

#### 1.1.2 Mathematical formulation

In reference [26], formulation of the mathematical model is developed for:

1. The white/black hole tunnels and exchange the objects of universes.
2. Maintaining the diversity of universes and perform exploitation.

In our algorithm, formulation and mathematical model have been proposed for:

- a. Initial population:
  - For each solution, some solutions (based on the rank of that solution) will be created very near the solution randomly.
- b. Explosion of solutions:
  - Each solution would be changed in direction of the vector which connects that solution and the solution of previous population.

**Table 2** Key Differences of MVA and MVO in [26]

	MVO [26]	MVA
Concepts and inspiration	1. White hole 2. Black hole 3. Wormhole	1. Parallel worlds 2. Different big bangs
Mathematical models	1. Exploration 2. Exploitation 3. Local search	1. Creation of the initial population 2. Explosion of the solutions 3. Rotation of universes
The procedure	1. White holes 2. Black holes 3. White holes 4. Black holes 5. Wormholes	1. Initial population 2. Ranking of solutions 3. Making dense solutions 4. Big bangs 5. Finding best solutions 6. Termination

### 1.1.3 The procedure

The procedure of the proposed MVA is based on: Initial population, Ranking of solutions, Making dense solutions, Big bangs, Finding best solutions and Termination. Which is completely different with steps of algorithm in reference [26]:

1. The higher inflation rate, the higher probability of having white hole.
2. The higher inflation rate, the lower probability of having black holes.
3. Universes with higher inflation rate tend to send objects through white holes.
4. Universes with lower inflation rate tend to receive more objects through black holes.
5. The objects in all universes may face random movement towards the best universe via wormholes regardless of the inflation rate.

## 2 Source of inspiration

This section presents a simulation of multiverse theory as an optimizer, named multiverse algorithm (MVA), Here we explained the principle concepts of MVA, mathematical equations and process of the algorithm to find optimal solution in optimization problems. The basic idea of multiverse theory is developed from string theory. This theory states that there are several universes in the world. More particularly, multiverse theory more than one big bang are existing besides to the big bang of our universe [10].

Meta-heuristics have main concepts, which have been simulated from treatment of animals, insects or natural events. The most important concept of ant colony is pheromone of ants, particle swarm optimization has been based on the global best, while main concept of genetic algorithm is combination, warming of egg in laying chicken algorithm (LCA) and explosion in big bang algorithm are the most important concepts of these algorithms. In this paper, we have inspired MVA algorithm from mainly from the existence of several worlds and big bangs.

As mentioned previously according to the multiverse theory there are several universes. Thus, MVA algorithm starts with a set of solutions as initial population. In the multiverse theory all universes come from very small and dense particle, so this is great idea to create the next population very near to the solutions of initial population, which has been simulated by our MVA algorithm. In the next step all solutions, which were very near to each other, are distributed in the feasible region same as big bangs.

Eventually, the best solution of each area will be accordingly explored and found, afterwards, these solutions will construct the next population. Each area corresponds a universe in multiverse theory. In each iteration, better solutions will be surrounded by more new solutions of the next population. This is because in the multiverse theory universe with more dark energy has more galaxies and planets.

Likewise, the proposed algorithm focuses on the beginning of the world and converting into the present complexity of the world. In the other words, MVA intends to answer this question: “how the world was started and changed by passing time?” One of the features of MVA is based on population. Particularly, in each iteration the algorithm changes and modifies population as well as the set of solutions.

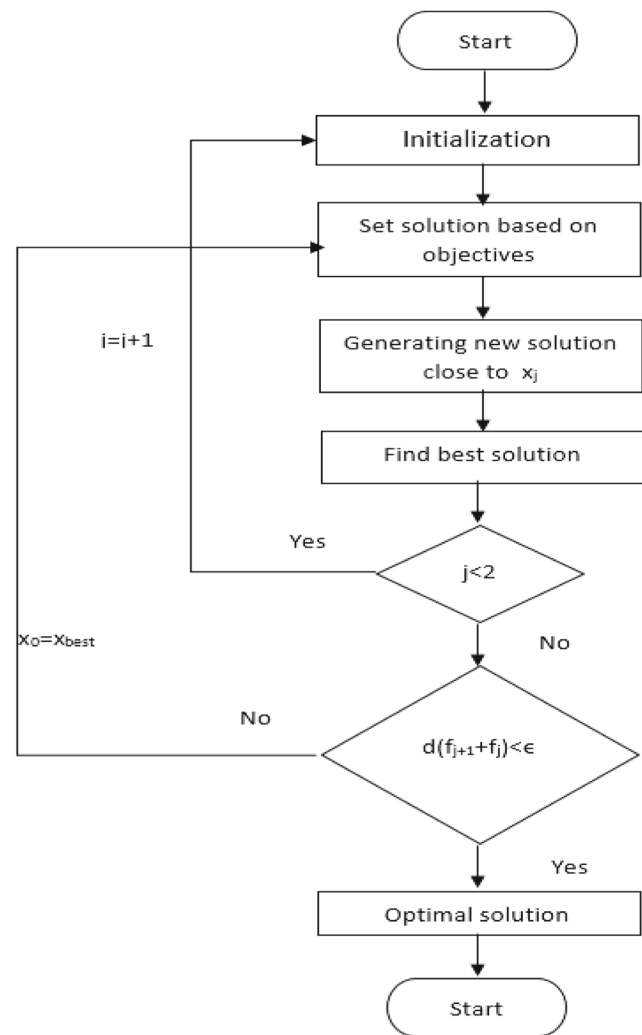


Fig. 1 The flowchart of proposed MVA

### 3 The proposed multiverse algorithm (MVA)

This section presents the detail of the proposed MVA technique in terms of its process of generating solutions and populations, followed by the process of explosion of solutions, then the main procedure of MVA is presented by this section accordingly. Figure 1 illustrates the procedure of the proposed MVA.

#### 3.1 The solutions and populations

Initial population is created in feasible region same as the first hypotheses of multiverse theory, which is stating that there are parallel worlds not just one. In fact, each solution in MVA displays a universe in multiverse theory. In multiverse theory, each universe is defined as a dark energy and this is great idea to sort solutions of populations and to correspond a rank to each solution based on their objective functions.

The number of random solutions is defined according to the rank of original solution. In fact, the algorithm tries to generate more solutions close to the relatively better solutions. This is exactly taken from the concept of multiverse theory, which states that a universe with large dark energy is larger and has more galaxies. For each solution  $x(i)$ , the random solutions  $x(j)$  are created according to non-equation  $||x_i - x_j|| \leq \epsilon$ , which is the famous Euclidean norm in mathematics [31].

In  $R^n$ ,  $i=1,2,\dots,n$  and  $j=1,2,\dots,m$ ,  $\epsilon$  is a small positive number,  $n$  is number of solutions in previous population and  $m$  is defined according to the rank of  $x_i$ , it is larger for solutions which have better rank. This procedure is illustrated in Algorithm 1. In the algorithm,  $i$  is defined as number of solutions that are generated randomly at the beginning of the MVA technique,  $j$  is defined as number of solutions for each previous solution and The value of  $j$  is calculated based on the objective function. Further,  $k$  is the number of iterations of the algorithm.

---

**Algorithm 1** Pseudo code of solutions and populations.

---

- 1: Number of solutions is  $n$
  - 2:  $\epsilon$  is a given small positive number
  - 3: Generate  $n$  feasible solutions
  - 4: Sort the solutions according their objective functions
  - 5:  $r_i$  is the rank of  $x_i$
  - 6:  $m$  is the number which is defined based on rank of  $x_i$
  - 7: **for**  $i = 1$  to  $n$  **do**
  - 8:     **for**  $j = 1$  to  $m$  **do**
  - 9:         Generate solution  $x_j$  randomly according the following
  - 10:          $||x_i - x_j|| \leq \epsilon$
  - 11:     **end for**
  - 12: **end for**
- 

As illustrated in Fig. 2a, 24 feasible solutions in the initial population (blue points) have been shown for a given problem and next population (green points) is distributed close to them according to the solution ranking in the initial population with  $\epsilon=0.2$ .

#### 3.2 Explosion of solutions

In the explosion process of MVA, each solution of current population (green points) is going away from the solution of previous population (blue points) after the big bangs. This has been simulated from the big bang of each universe in multiverse theory. In fact, each solution would be changed in direction of the vector, which connects the current solution and the solution of previous population. These movements are according to  $x_j = x_i + \lambda d_{ij}$  known as the equation of movements, where,  $d_{ij}$  is distance between points  $x_i, x_j$  and  $\lambda$  is a constant.

Moreover, MVA tries to explode all solutions which are near to previous solutions. These solutions have been shown by black points in Fig. 2b. The best solution in each universe has been shown by red points. As illustrated in Fig. 2b, The algorithm will continue by tagging the red points to be categorised as new population. Then, the solutions in current generation (red points) are highlighted to be better than solutions obtained by the previous population (blue points initial population). Algorithm 2 shows the pseudo code of the stage of explosion of solutions.

---

**Algorithm 2** Pseudo code of explosion of solutions.

---

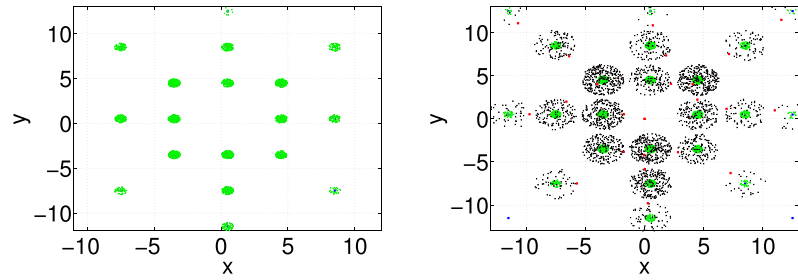
- 1:  $\alpha$  is given constant
  - 2:  $n$  is the number of initial population
  - 3:  $n_i$  is the number of solutions in  $i_{th}$  area
  - 4:  $f$  is the objective function
  - 5: **for**  $i = 1$  to  $n$  **do**
  - 6:     **for**  $j = 1$  to  $n_i$  **do**
  - 7:         **while**  $x_j$  is feasible **do**
  - 8:              $x_{j+1} = x_i + \alpha d_{ij}$
  - 9:         **end while**
  - 10:     **end for**
  - 11:     Evaluate new solutions and update  $f$
  - 12: **end for**
- 

In order to solve multi-objective problems, Algorithm 2 updates and evaluates the solutions based on Equation 1:

$$BS = \begin{cases} x_1 & \text{if } a < 0 \\ x_2 & \text{if } a > 0 \end{cases} \tag{1}$$

If we consider all objective functions are representing minimization problems and BS is defined as a better

**Fig. 2** Initial population and its changing process in simulation of big bangs



(a) Initial population has been distributed in space (b) Process of changing solutions in simulation of big bangs

solution between  $x_1, x_2$ , and  $f = (f_1, f_2, \dots, f_n)$  then  $a$  is defined in (2):

$$a = \sum_{i=1}^n (f_i(x_1) - f_i(x_2)) \tag{2}$$

and

$$BS = \begin{cases} x_1 & \text{if } a > 0 \\ x_2 & \text{if } a < 0 \end{cases} \tag{3}$$

Equation 3 for the cases where all objective functions are representing maximization problems.

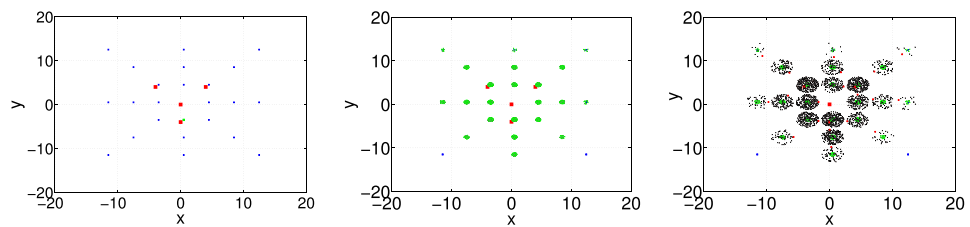
### 3.3 The procedure of the MVA

This section presents the procedure of the proposed MVA meta-heuristic technique. Algorithm 1 provides initial solution and population as stated in steps 1 to 3 while steps 4 to 6 are solved by Algorithm 2.

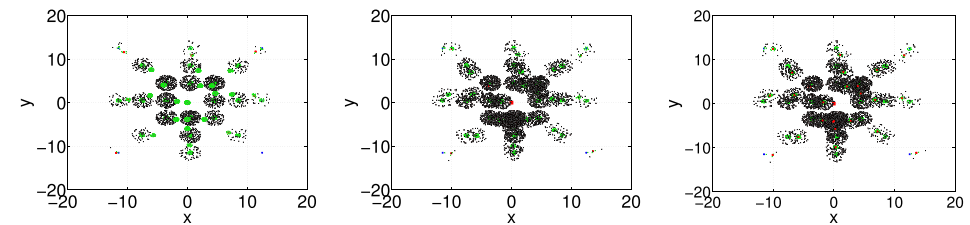
1. Initial population is generated in all feasible regions.  $N$  is number of solutions,  $k = 0$  and  $\epsilon$  is a given positive small number,  $i = 1$ . This step is illustrated in Fig. 3a.

2. All solutions will be sorted according to their objective function. In this step, a specific rank is assigned to each solution. This step is illustrated in Fig. 3b.
3. For each solution  $x_i$  from initial population, some solutions will be generated close to  $x_i$ . Number of these solutions depends on the rank of  $x_i$  in step two. For example, the number of solutions in the current population will be gathered near to the best solution in the previous population. In fact, current population is distributed among solutions of previous generation. This step is illustrated in Fig. 3c.
4. All solutions in current population are getting away from solutions of previous generation. Here, solutions will be exploded into the space as illustrated in 3d.
5. Find the best solution of current population. If  $j < 2$  let  $j=j+1$  and, then go to the step 2. This step is illustrated in Fig. 3e.
6. If  $d(f(x_{j+1}), f(x_j)) < \epsilon$  then the algorithm will be finished and  $x_{j+1}$  is the best solution by MVA  $x_j$  is the best solution in  $j^{th}$  iteration. Otherwise, let  $j=j+1$  and go to the step 2,  $d$  is defined in mathematical metric 4

**Fig. 3** Steps of the MVA to obtain optimal solution  $R^2$



(a) Step 1-Initial Population (b) Step 2-New Population (c) Step 3-Improving of Population



(d) Step 4- New Population (e) Step 5- Explosion of Population (f) Step 6- Optimal Solution

**Table 3** Results of MVA for Ackley Function - Example 2

Algorithms	N. Agents	N. Iterations	Optimal solution	F Min	$\epsilon$	Initial solution
MVA	24	1	(-3.2, -2.3)	12.2634	0.1	(-5, -8, 20.16)
MVA	24	2	(0,0)	0	0.1	(-5, -8, 20.16)

[31] and Fig. 3 shows the process of the algorithm to find optimal solution in  $R^2$  (2 Dimension).

$$\max_i |f(x_{j+1}^i) - f(x_j^i)| = d(f(x_{j+1}), f(x_j)) \quad (4)$$

### 4 Computational results

In this section, both kinds of continuous in small size and discrete in large size optimization problems are solved.

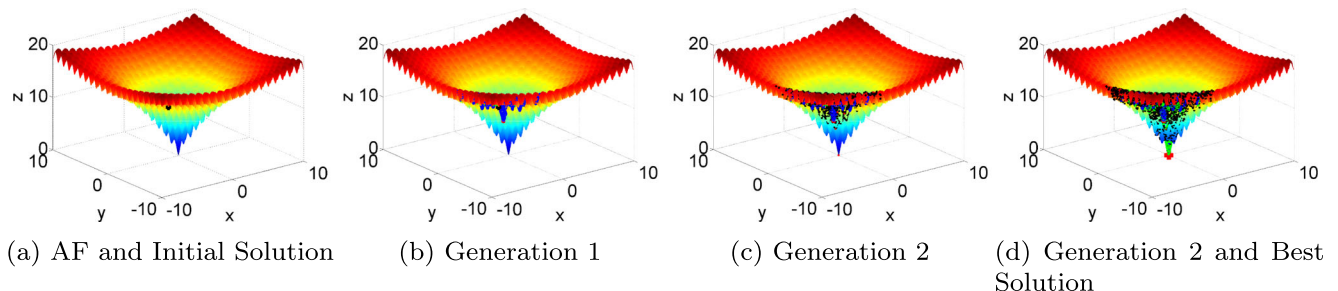
#### 4.1 Continuous problems

In this section, almost all kinds of continuous optimization problems: constrained, unconstrained, linear, non-linear, multi-level and multi-objective are solved.

*Example 1* Consider Ackley Function (AF):

$$\min -20 \exp(-0.2 \sqrt{0.5(x^2 + y^2)}) - \exp(0.5(\cos(2\pi x) + \cos(2\pi y))) + \exp(1) + 20 \quad (5)$$

The proposed MVA is applied to solve the optimization problem in (5). Table 3 depicts the process of how the algorithm reaches to the optimal solution (0,0) after just two iterations. Further, the process of the algorithm, initial population, optimal solution of generations and constraints of the problems have been shown for two iterations in Fig. 4. As can be seen, the optimal solution, big red point in Fig. 4d, has surrounded by solution of generation 2.



**Fig. 4** Generations move to find optimal solution by MVA- Example 1

*Example 2* Consider Hölder Table Function (HTF):

The optimization problem that is represented by (6) has been solved by MVA. The results is shown in Table 4. The process of the algorithm, initial population, optimal solution of generations and constraints of the problems have been shown for two iterations in Fig. 5. Holder Table Function has four global optimal solutions, (8.05502, 9.66459), (-8.05502, 9.66459), (8.05502, -9.66459), (-8.05502, -9.66459), with -19.2085 objective function value. The proposed algorithm incredibly obtains (-8.05502, -9.66459) just after two iterations.

$$\min -|\sin(x)\cos(y)\exp(|1 - \sqrt{(x^2 + y^2)}/\pi)| \quad (6)$$

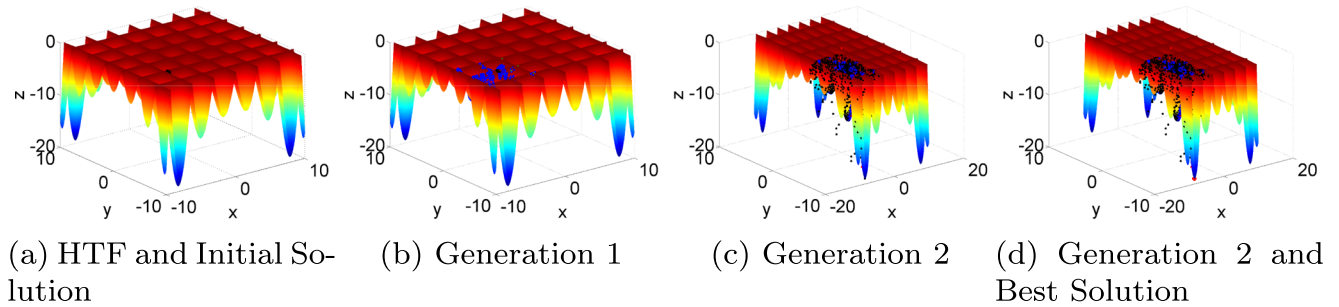
*Example 3* Consider Mishra’s Bird Function (MBF):

$$\min \sin(x)\exp((1 - \cos(y))^2) + \cos(y)\exp((1 - \sin(x))^2) + (x - y)^2 \quad (7)$$

The problem has been solved by MVA, results have been shown in Table 5 and also the process of the algorithm, initial population, optimal solution of generations in addition to the constraints of the problems have been shown for two iterations in Fig. 6. Global optimal of Mishra’s Bird Function is (-3.1302468, -1.5821422), with objective function value of -106.7645367. MVA could find the optimal solution during two populations, which has been shown in Table 5 and also in Fig. 6d as the large red point.

**Table 4** Results of MVA for Hölder Table Function - Example 3

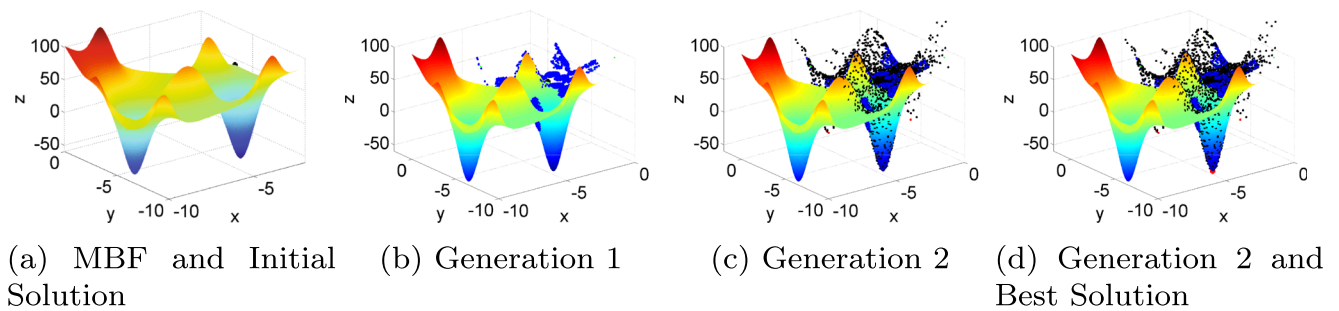
Algorithms	N. Agents	N. Iterations	Optimal solution	F Min	$\epsilon$	Initial solution
MVA	24	2	(-8.05, -9.66)	-19.20	0.1	(-5, -6, -0.35)



**Fig. 5** Generations move to find optimal solution by MVA- Example 2

**Table 5** Results of MVA for Mishra's Bird Function - Example 4

Algorithms	N. Agents	N. Iterations	Optimal solution	F Min	$\epsilon$	Initial solution
MVA	24	2	(-3.13,-1.58)	-106.7	0.1	(0,0,2.71)



**Fig. 6** Generations move to find optimal solution by MVA- Example 3



**Table 6** Comparison of MVA and other methods- Example 4

Algorithms	N. Agents	N. Iterations	Optimal solution	F Min	$\epsilon$	(x,y)
MVA	24	4	(4,4)	-12	0.1	(2,2)
Classic methods [13]	None	None	(4,4)	-12	None	None
LS and TM [13]	None	None	(3.9,4)	-12.1	None	None

Example 4 [13]:

Consider the following linear bi-level programming problem:

$$\begin{aligned}
 &\min x - 4y \\
 &\quad \min y \\
 &x + y \geq 3 \\
 &-2x + y \leq 0 \\
 &2x + y \leq 12 \\
 &3x - 2y \leq 4 \\
 &x, y \geq 0
 \end{aligned} \tag{8}$$

Using Search Results Web results

Karush–Kuhn–Tucker (KKT) conditions the problem will be converted to the following problem:

$$\begin{aligned}
 &\min x - 4y \\
 &-\lambda_1 + \lambda_2 + \lambda_3 - 2\lambda_4 = -1 \\
 &\lambda_1(-x - y + 3) = 0 \\
 &\lambda_2(-2x + y) = 0 \\
 &\lambda_3(2x + y - 12) = 0 \\
 &\lambda_4(3x - 2y - 4) = 0 \\
 &-x - y + 3 \leq 0 \\
 &-2x + y \leq 0 \\
 &2x + y - 12 \leq 0 \\
 &3x - 2y - 4 \leq 0 \\
 &x, y, \lambda_1, \lambda_2, \lambda_3, \lambda_4 \geq 0
 \end{aligned} \tag{9}$$

The bi-level programming problem is difficult, because two objective functions should be optimize in two different levels at the same time. So proposing a method, which can solve such kind of problems is significant. MVA proposed the optimal solution same as exact algorithms according

to Table 6. Number of iteration taken to find the optimal solution is completely low. Also, the proposed solution by LS and TM [13], (3.9,4), is feasible for all constraints of the second level of the problem, but it is infeasible for bi-level programming problem. Behavior of solutions, constraints of the problem and optimal solution are shown in Fig. 7.

More examples are solved by MVA and numerical results and behavior of populations are shown in Tables 7 and 8 for Examples 5 and 6.

Example 5 [15]

Consider the following linear programming problem (Fig. 8):

$$\begin{aligned}
 &\min -3x_1 + x_2 \\
 &x_1 + 2x_2 \leq 4 \\
 &-x_1 + x_2 \leq 1 \\
 &x_1, x_2 \geq 0
 \end{aligned} \tag{10}$$

Example 6 [9] ( Non-linear)

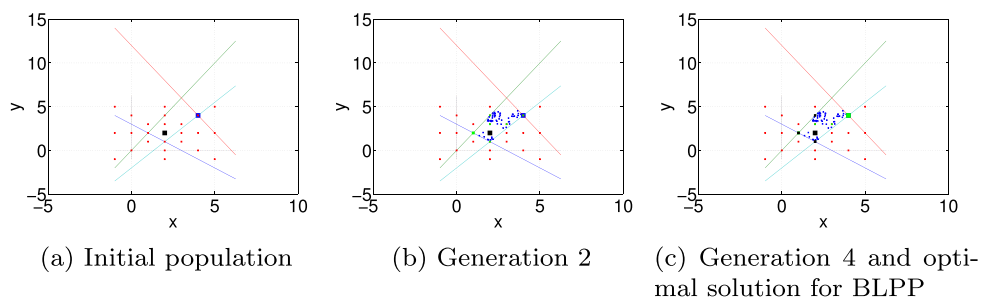
Consider the following non-linear unconstraint optimization problem:

$$\max e^{-(x-4)^2-(y-4)^2} + e^{-(x+4)^2-(y-4)^2} + 2e^{-x^2-y^2} + 2e^{-x^2-(y+4)^2} \tag{11}$$

Example 7 [23] (Multi-Objective):

Here MVA is used for solving Deb, Thiele, Laumanns and Zitzler (DTLZ) benchmark problems. Behavior of the algorithm to find Pareto optimal for DTLZ1 problem has been shown in Fig. 9. Feasibility of the algorithm is clear based on Fig. 9c to control sequence, which shows initial population, because some of solutions in the population have been reached to Pareto optimal. Moreover, efficiency of the algorithm is obvious by comparison of Fig. 9a and c. Most of solutions are completely far from Pareto optimal

**Fig. 7** Process of finding optimal solution by MV- Example 4



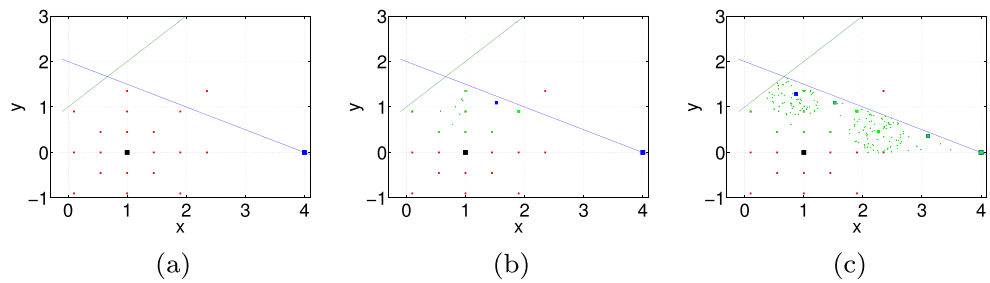
**Table 7** Comparison of MV and other methods in examples 5 and 6

Examples	Optimal solution (OS)	Iteration	Objective function	(x,y)
Example 5	2	3	1.9998	(0,0)
Example 6	-12	2	-10.14	(3.42,0.12)

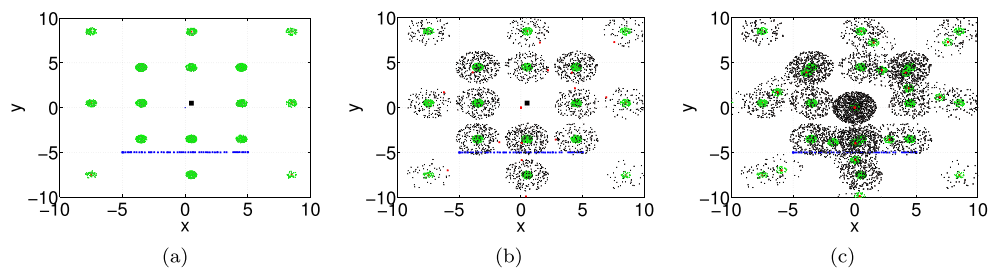
**Table 8** Comparison of MVA and other methods for DTLZ problems

Problems	k	ParEGO			MVA		
		min	mean	max	min	mean	max
DTLZ1	3	13.42	52.47	112.7	8.54	29.63	76.19
DTLZ1	10	NA	NA	NA	1.05	1.45	1.76
DTLZ2	3	0.151	0.191	0.243	0.136	0.195	0.203
DTLZ2	10	NA	NA	NA	0.098	0.154	0.194
DTLZ3	3	81.15	145.5	261.6	46.98	111.67	200.43
DTLZ3	10	NA	NA	NA	0.79	1.03	1.87

**Fig. 8** Behavior of populations to get optimal solution by applying proposed MVA for example 5



**Fig. 9** Behavior of populations to get Pareto optimal solution by MVA for DTLZ1 with k=2



**Table 9** HV Results of MVA and other algorithm over DTLZ1-DTLZ5

		MVA	MaOEA/IGD	NSGA-III	MOEA/D	HypE
DTLZ1	8	0.9932(6.84E-4)	<b>0.9998(2.93E-4)</b>	0.9964(6.12E-4)	0.9996(3.52E-5)	0.7213(4.31E-1)
	15	0.9914(8.17E-4)	<b>0.9990(3.18E-3)</b>	0.9984(7.23E-4)	0.9987(3.20E-4)	0.6922(5.45E-1)
	20	<b>0.9991(2.28E-3)</b>	0.9990(2.32E-3)	0.9983(3.82E-4)	0.9977(7.24E-4)	0.7672(3.88E-1)
DTLZ2	8	<b>0.8745(2.13E-3)</b>	0.7174(3.96E-3)	0.8132(2.78E-3)	0.5221(3.83E-3)	0.1121(3.34E-2)
	15	0.9262(2.86E-3)	<b>0.9268(2.62E-3)</b>	0.8832(9.11E-3)	0.3329(1.73E-2)	0.0892(4.12E-2)
	20	<b>0.9750(2.81E-3)</b>	0.8905(6.80E-3)	0.9660(3.23E-3)	0.3298(2.10E-2)	0.0633(5.32E-2)
DTLZ3	8	<b>0.5863(4.23E-3)</b>	0.4664(9.25E-2)	0.0055(3.80E-4)	0.5169(5.68E-3)	0.0085(0.76E-5)
	15	0.3028(4.40E-3)	<b>0.6984(6.68E-2)</b>	0.0091(0.78E-5)	0.3030(4.43E-3)	0.0133(1.07E-5)
	20	<b>0.7547(7.73E-2)</b>	0.7476(7.52E-2)	0.0002(6.48E-4)	0.2162(4.51E-4)	0.0065(5.47E-4)
DTLZ4	8	0.8335(3.41E-3)	<b>0.8338(3.31E-3)</b>	0.8187(6.22E-4)	0.5322(5.87E-2)	0.2537(2.08E-4)
	15	<b>0.9665(1.42E-3)</b>	0.9548(1.66E-3)	0.9537(4.24E-4)	0.3150(5.08E-3)	0.1957(0.86E-4)
	20	0.9854(1.35E-3)	0.9824(1.33E-3)	<b>0.9947(1.37E-3)</b>	0.2755(7.21E-5)	0.2101(1.07E-4)
DTLZ5	8	<b>0.4832(0.53E-3)</b>	0.4190(0.64E-3)	0.3908(7.67E-3)	0.3174(7.15E-5)	0.0451(2.24E-5)
	15	<b>0.3812(8.02E-3)</b>	0.2677(9.71E-3)	0.2178(5.34E-5)	0.1821(8.85E-2)	0.0418(8.99E-5)
	20	0.3157(0.47E-2)	0.2101(5.57E-3)	<b>0.3390(0.44E-2)</b>	0.1790(0.99E-4)	0.0423(4.90E-2)

at first, but during the process of applying our proposed algorithm, solutions have achieved Pareto optimal. Further, Fig. 9c shows that the last population has surrounded Pareto optimal solutions.

For single objective problems, we have used the same procedure of the proposed MVA technique while multi-objective problems, a set of solutions have been generated in the feasible region. Additionally, applying the procedure of the MVA on solving singly objective problems set, till the algorithm reaches the Pareto optimal solutions as shown in Fig. 9.

Table 8 shows the comparison of best solutions in obtaining Pareto optimal of DTLZ problems by MVA and ParEGO which is a method that has used in reference [24].

To evaluate the performance of the proposed algorithm, Hyper-Volume (HV) is used as a performance metric in Table 9. HV metric simultaneously measures the convergence of many-objective optimization problems. In Table 9, the HV values are normalized between [0,1] by dividing the HV value of the origin with the corresponding reference point. Thus, higher value of HV interprets better performance of the corresponding many-objective optimization problem. In the simulation, the number of population is set to 240, the maximum number of iterations

is equal to 100, epsilon value is 0.1, and the number that each algorithm has been carried out is 30 times. Table 9 illustrates the performance of the proposed MVA as compared with the existing algorithms for solving test problems with specific objective numbers. Here, we used HV as a performance metric to fairly judge the efficiency of the algorithms. Further, the best mean values of the corresponding test problem has been shown in bold, based on the results of HV on DTLZ1-DTLZ5 test problems. It is worth to highlight that MVA achieves the best performance as compared with its peer competitors.

### 4.2 Large size practical problems

To show efficiency of the algorithm for real life problems, this section presents three kinds of practical problems: large size of real linear programming problems, transportation problems and internet of vehicles problems. Then, the proposed MVA is applied to solve the aforementioned real life problems.

Some benchmark of linear programming can be found in NetLib repository such as aggregate function (agg), Quadratic assignment problem8 (qap8), SC50A, AFIRO. Table 10 confirms that the MVA can solve large size problems. Note that the agg, qap8, SC50A, AFIRO are

**Table 10** Results of MVA for more test problems

Name	Size	Optimal	Linprog	MVA	N. Iterations
agg	489 163	-3.5991767287E+07	-3.9217e+16	- 3.599173e+07	15
qap8	913 1632	2.0350000000E+02	-1.6987e+16	2.378e+02	25
SC50A	51 48	-6.4575077059E+01	-6.5313e+20	-6.5890e+01	10
AFIRO	28 32	-4.6475314286E+02	-1.4505e+29	-4.8741e+02	10

**Table 11** Comparison among MVA and other algorithms for large size problems

Problem	Size	North-West	Vogel	MVA	Improvement using MVA
Transportation 1	80 20	132804	30123	21345	0.29
Transportation 2	100 25	177666	26462	25387	0.04
Transportation 3	160 40	185366	85456	60459	0.29
Transportation 4	200 50	297629	26566	20345	0.23
Transportation 5	210 70	322356	27619	24897	0.10
Transportation 6	261 87	245311	152930	120754	0.21

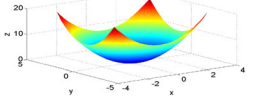
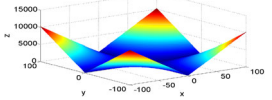
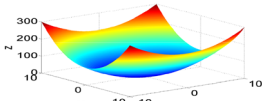
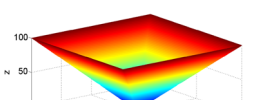
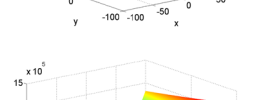
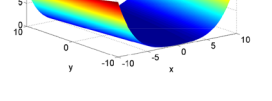
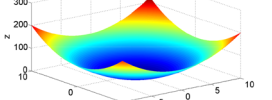
linear programming test problems in the “NETLIB Linear Programming test set” which is a collection of real-life linear programming examples.

Finding a suitable feasible solution of transportation problem is remarkable, so MVA has been applied to some random transportation problems [19]. The obtained results have been listed in Table 11.

North-West and Vogel are two famous algorithms are used in finding feasible solution of transportation problems. Comparison with Vogel algorithm, the best algorithm, in Table 11 ascertains the preference of MVA.

Finally, MVA is applied for solving route optimization problem in IoV scenario as illustrated in [25]. Table 14 shows the higher efficiency has been obtained by deploying

**Table 12** Optimization test functions

Functions	Equations	Figures
F1-Sphere function	$f(\mathbf{x}) = f(x_1, \dots, x_n) = \max_{i=1, \dots, n}  x_i $	
F2-Schwefel 2.22 function	$f(\mathbf{x}) = f(x_1, \dots, x_n) = \sum_{i=1}^n  x_i  + \prod_{i=1}^n  x_i $	
F3-Sum squares function	$f(\mathbf{x}) = f(x_1, \dots, x_n) = \sum_{i=1}^n ix_i^2$	
F4-Schwefel 2.21 function	$f(\mathbf{x}) = f(x_1, x_2, \dots, x_n) = \sum_{i=1}^n x_i^2$	
F5-Rosenbrock function	$f(\mathbf{x}) = f(x_1, \dots, x_n) = \sum_{i=1}^n x_i^2 + \left( \sum_{i=1}^n x_i + n/4 \right)$	
F6-Zakharov function	$f(\mathbf{x}) = f(x_1, \dots, x_n) = -a \cdot \exp\left(-b \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(cx_i)\right) + a + \exp(1)$	
F7-Quartic function	$f(x, y) = \sin^2(3\pi x) + (x - 1)^2(1 + \sin^2(3\pi y)) + (y - 1)^2(1 + \sin^2(2\pi y))$	

**Table 13** Optimization test functions

Functions	Equations	Figures
F8-Schwefel function	$f(\mathbf{x}) = f(x_1, \dots, x_n) = \sum_{i=1}^n ix_i^4 + \text{random}[0, 1)$	
F9-Rastrigin function	$f(x, y) = 10n + \sum_{i=1}^n (x_i^2 - 10\cos(2\pi x_i))$	
F10-Ackley function	$f(x, y) = \sum_{i=1}^n [b(x_{i+1} - x_i^2)^2 + (a - x_i)^2]$	
Function 11 salomon function	$f(\mathbf{x}) = f(x_1, \dots, x_n) = 1 - \cos(2\pi \sqrt{\sum_{i=1}^D x_i^2}) + 0.1 \sqrt{\sum_{i=1}^D x_i^2}$	
F12-Levi N. 13 function	$f(\mathbf{x}) = f(x_1, x_2, \dots, x_n) = 418.9829d - \sum_{i=1}^n x_i \sin(\sqrt{ x_i })$	
Function 13 Alpine N. 1 function	$f(\mathbf{x}) = f(x_1, \dots, x_n) = \sum i = 1^n  x_i \sin(x_i) + 0.1x_i $	

the proposed MVA as compared with the the benchmark LCA algorithm (Tables 12 and 13).

For each problem initial solutions have been generated randomly and they are different for both LCA and MVA algorithms. Table 14 shows improvement of their initial solutions after five iterations.

### 5 Comparison with other optimization algorithms

MVA is used to solve two different test functions: unimodal and multi-modal. Unimodal test functions have one global optimum and multi-modal test functions have a

**Table 14** Comparison of LCA and MVA for internet of vehicles

Problems	Size	Best Solution LCA	Best Solution MVA	Improvement by MVA
IoV 1	100 100	775.8550	856.4378	0.17
IoV 2	200 200	9.9319e+03	1.4698e+04	0.49
IoV 3	500 500	5.8147e+04	6.7408e+04	0.17
IoV 4	1000 1000	2.5991e+05	2.9631e+05	0.14
IoV 5	2000 2000	9.8622e+05	1.4790e+06	0.50
IoV 6	5000 5000	6.2266e+06	6.51231e+06	0.05
IoV 7	10000 10000	2.4950e+07	2.8547e+07	0.11

**Table 15** Comparison of MVA and existing metaheuristic methods

F	MVA		MVO		GWO		PSO		GA	
	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
F1	1.0589	0.4698	2.08583	0.648651	2319.19	1237.109	3.552364	2.85373	27,187.58	2745.82
F2	5.3647	3.4279	15.92479	44.7459	14.43166	5.923015	8.716272	4.929157	68.6618	6.062311
F3	123.4689	85.7954	453.2002	177.0973	7278.133	2143.116	2380.963	1183.351	48,530.91	8249.75
F4	1.9361	1.3689	3.123005	1.582907	13.09729	11.3469	21.5169	6.71628	62.99326	2.535643
F5	836.279	756.148	1272.13	1479.477	3425,462	3304,309	1132.486	1357.967	65,361,620	29,714,021
F6	1.5824	0.6843	2.29495	0.630813	5009.442	3028.875	86.62074	147.3067	49,574.1	8545.149
F7	0.01478	0.0115	0.051991	0.029606	0.408082	0.119544	0.577434	0.318544	18.72524	4.935256

global optimum as well as multiple local optima. For the verification of the results, proposed algorithm is compared with MVO [26], PSO [2], GA [1], GWO [28]. Note that the number of agents is set to 24, the maximum number of iterations is equal to 100, epsilon value is  $\epsilon$  0.1, and the number that each algorithm has been carried out is 20 times. The results of Tables 15 and 16 show that the proposed algorithm is able to provide very competitive and efficient results on both the unimodal and multi-modal test functions. Low standard deviation of MVA is remarkable, which indicates that the values tend to be close to the mean of the set of solutions.

## 6 Convergence behavior of MVA

In MVA, each solution of population will be exploded in space, so it needs a large space to improve the obtained solutions. Therefore, problems with large feasible region, the algorithm improves population very fast and gets appropriate solution. Thus, MVA is completely efficient in

solving unconstrained and unbounded types of problems. Also, it proposes suitable solutions for constrained problems with large feasible region. However, MVA is not very efficient in solving problems with small feasible region. In this case, if MVA is starting from infeasible solutions, better results can be found. For example, by changing initial population in Example 5 much better result will be found according to Table 17 and Fig. 10:

Figure 10a shows initial population by only one feasible solution. In Fig. 10b just the feasible solution of the previous population will be exploded (green point).

In this paper, we introduced generic optimization problems and solutions with the help of developed MVA. However, when we consider a typical CPS system, we need to consider the domain specific parameters while optimizing the overall performance. For instance, low latency is required in almost all CPS systems such as transportation CPS and energy CPS, information should be propagated in fraction of second (10 ms to 500 ms depending on the types of messages in the systems) [34, 35]. Energy CPS does not have to deal with mobility that much since most

**Table 16** Comparison of MVA and existing metaheuristic methods

F	MVA		MVO		GWO		PSO		GA	
	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
F8	-8932	526.78	-11, 720	937.1975	-10, 739	1162.793	-6727	1352.882	-10, 698	602.3045
F9	13.6397	8.4825	118.046	39.34364	89.13475	37.95765	99.83202	24.62872	273.2519	29.55218
F10	1.6479	0.8579	4.074904	5.501546	9.452571	3.467608	4.295044	1.308386	18.59657	0.351737
F11	0.0258	0.0085	0.938733	0.059535	22.51942	26.68168	624.3092	105.3874	353.3655	77.26729
F12	0.7462	0.1036	2.459953	0.791886	3,200,008	6,746,208	13.38384	8.969122	2.21e+08	1.1e+08
F13	0.1578	0.00146	2.459953	0.086407	7,815,082	16,475,640	21.11298	12.83179	4.49e+08	2.26e+08

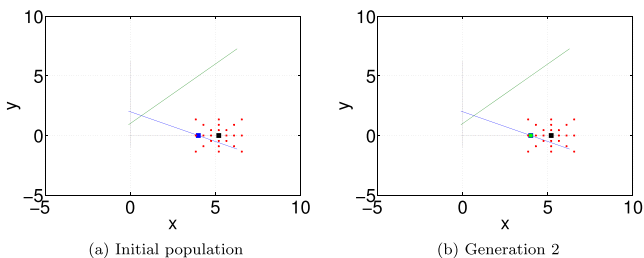
**Table 17** Comparison of MVA and exact methods by changing initial population

Examples	OS	Iteration	Objective function	(x,y)
Example 5	-12	2	-11.55	(3.85,0.00)

of the energy assets are fixed. However, for transportation CPS, most of the CPS nodes are mobile [36, 37]. While optimizing, we need to consider mobility on top of other parameters such as delay and high throughput. So, future research can focus on generic optimization that can be fine-tuned to a domain specific problem such as optimization problem with mobility constraint can be relaxed when we consider speed or velocity of the node equal to 0. Further research could focus on time-varying sampling patterns, sensor scheduling, real-time control, feedback scheduling, task and motion planning and resource sharing for different CPS systems.

## 7 Conclusion

In this paper, we developed a novel meta-heuristic algorithm named MVA, which is inspired from a scientific theory of multiverse. MVA is a naive optimizer, which optimizes most kinds of optimization programming problems. The proposed algorithm is applicable for unconstrained and constrained with small and large feasible regions. In particular, several types of complex Engineering problems, including problems in CPS, can be solved by our proposed MVA because of its fast convergence and lower appropriate complexity. In this paper, extensive simulations have been carried out to get numerical results which show the feasibility of our proposed MVA. We observed that the MVS outperforms the existing well known meta heuristic algorithms, especially for large size real problems.

**Fig. 10** Example 6 by infeasible initial population

## Compliance with Ethical Standards

**Conflict of interests** The authors declare that they have no conflict of interest. Moreover, this research was not funded by any funding agency.

## References

- Holland J (1975) Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor
- Fogel LJ, Owens AJ, Walsh MJ (1966) Artificial intelligence through simulated evolution. Wiley
- Kirkpatrick S, Gellat CD, Vecchi MP (1983) Optimization by simulated annealing. *Science* 220:671–680
- Dorigo M (1992) Optimization, Learning and Natural Algorithms PhD thesis, Politecnico di Milano, Italy
- Kennedy J, Eberhart RC (1995) Particle swarm optimization. In: Proceedings of IEEE international conference on neural networks, Perth, Australia, pp 1942–1948
- Storn R, Price K (1997) Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim* 11:341–359
- Karaboga D, Basturk B (2007) A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, Springer Netherlands 39(3):459–471
- Gandomi AH, Alavi AH (2012) Krill herd: A new bio-inspired optimization algorithm. *Commun Nonlinear Sci Numer Simulation* 17:4831–4845
- Yang XS (2013) Bat algorithm: literature review and applications. *Int J Bio Inspired Comput* 5(3):141–149
- Cuevas E, Cienfuegos M, Zaldivar D, Cisneros M (2013) A swarm optimization algorithm inspired in the behavior of the social-spider. *Expert Syst Appl* 40:6374–6384
- Meng X, Liu Y, Gao X, Zhang H (2014) A New Bio-inspired Algorithm: Chicken Swarm Optimization. ICSI 2014, Part I, LNCS 8794, pp 86–94
- Yang XS (2010) Nature- Inspired Meta-Heuristic Algorithms. University of Cambridge
- Wallace D (2012) The emergent multiverse: Quantum theory according to the Everett interpretation. Oxford University Press
- Bazzara M (2007) Non-Linear Programming Theory and Algorithms. Wiley Inc, New York
- Bazzara M (2010) Linear Programming and Network Flows. Wiley Inc., New York
- Hosseini E, Kamalabadi IN (2014) Solving linear bi-level programming problem using two new approaches based on line search and Taylor methods. *Int J Manage Sci Education* 2(6): 243–252
- Hosseini E (2017) Laying chicken algorithm: a new meta-heuristic approach to solve continuous programming problems. *J Appl Comput Math* 6(1):1–8
- Hosseini E (2017) Big bang algorithm: a new meta-heuristic approach for solving optimization problems. *Asian J Appl Sci* 10(4):334–344
- Hosseini E (2017) Three new methods to find initial basic feasible solution of transportation problems. *Appl Math Sci* 11:1803–1814
- Hosseini E (2017) Solving linear tri-level programming problem using heuristic method based on bi-section algorithm. *Asian Journal of Scientific Research* 10(4):227–235

21. Hosseini E (2018) Presentation and solving Non-Linear Quad-Level programming problem utilizing a heuristic approach based on Taylor theorem. *J Optim Industrial Eng (JOIE)* 1:91–101
22. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. *Adv Eng Softw* 69:46–61
23. Deb K, Thiele L, Laumanns M, Zitzler E (2002) Scalable multi-objective optimization test problems. In: *Proc IEEE Congr Evol Comput*, Honolulu, HI, USA, pp 825–830
24. Chugh T, Jin Y, Miettinen K, Hakanen J, Sindhya K (2018) A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization. *IEEE Trans Evol Comput* 22(1):129–142
25. Kayhan G, Linghe K, Rawat D, Eghbal H, Ali S (2018) Quality of service aware routing protocol in Software-Defined internet of vehicles. *IEEE Internet J* 6:2817–2828
26. Mirjalili S, Mirjalili SM (2016) Multi-Verse Optimizer: a nature-inspired algorithm for global optimization. *Neural Comput Appl* 27(2):495–513
27. Sayed GI, Darwish A, Hassanien AE (2017) Quantum multi-verse optimization algorithm for optimization problems. *Neural Computing and Applications* 1:1–18
28. Ewees AA, Abd El Aziz M, Hassanien AE (2019) Chaotic multi-verse optimizer-based feature selection. *Neural Computing and Applications* 31(4):991–1006
29. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. *Evolut Comput IEEE Trans* 1:67–82
30. Beni G, Wang J (1993) Swarm intelligence in cellular robotic systems. In: *Robots and biological systems: towards a new bionics?*, ed. Springer, pp 703–12
31. Rudin W (1976) *Principles of Mathematical Analysis*. McGraw-Hill Education; 3rd edition
32. Eghbal H, Zrar GK, Ali S, Mohsen G, Ali E (2020) COVID-19 Optimizer Algorithm, Modeling and Controlling of Coronavirus Distribution Process. *IEEE Journal of Biomedical and Health Informatics*, July, 27, 2, IEEE
33. Eghbal H, Safaa SA, Zrar GK, Danda R, Mehrdad S, Xinan Y (2020) Volcano eruption algorithm for solving optimization problems. *Neural Computing and Applications*, Springer 2:1–17
34. Danda R, Ghafoor KZ (2018) *Smart Cities Cybersecurity and Privacy*. Elsevier; 1st edition
35. Ghafoor KZ, Guizani M, Kong L, Maghdid HS, Jasim KF (2019) Enabling efficient coexistence of dsrc and c-v2x in vehicular networks. *IEEE Wireless Communications*, IEEE 27(2):134–140
36. Ghafoor KZ, Kong L, Zeadally S, Sadiq AS, Epiphaniou G, Hammoudeh M, Bashir AK, Mumtaz S (2020) Millimeter-Wave Communication for Internet of Vehicles: Status, Challenges and Perspectives. *IEEE Internet of Things Journal*, IEEE. <https://doi.org/10.1109/JIOT.2020.2992449>
37. Maghdid HS, Ghafoor KZ, Al-Talabani A, Sadiq AS, Singh PK, Rawat DB (2020) Enabling Accurate Indoor Localization for Different Platforms for Smart Cities Using a Transfer Learning Algorithm. *Internet Technology Letters*, Wiley Online Library. <https://doi.org/10.1002/itl2.200>

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Eghbal Hosseini** is currently working as senior researcher with Erbil Polytechnic University. Before that, he was a lecturer at University of Raparin. He received B.Sc. degree in applied mathematics from University of Razi, in 2005 and M.Sc. degree in operations research at University of Kurdistan in 2007. He received Ph.D. degree in optimization from Tehran Payame Noor University. His research interests are meta-heuristic approaches, algorithms, multi-

level programming problems and machine learning. From 2017 he has proposed four other new meta-heuristics Laying Chicken Algorithm (LCA), Big Bang Algorithm (BBA), Volcano Eruption Algorithm (VEA), and Covid-19 optimizer algorithm (CVA).



**Kayhan Zrar Ghafoor** is currently working as an associate professor at the Salahaddin University-Erbil and visiting scholar at the University of Wolverhampton. Before that, he was a postdoctoral research fellow at Shanghai Jiao Tong University, where he contributed to two research projects funded by National Natural Science Foundation of China and National Key Research and Development Program. He is also served as a visiting researcher at

University Technology Malaysia. He received the B.Sc. degree in electrical engineering, the M.Sc. degree in remote weather monitoring and the Ph.D. degree in wireless networks in 2003, 2006, and 2011, respectively. He is the author of 2 technical books, 7 book chapters, 65 technical papers indexed in ISI/Scopus. He is the recipient of the UTM Chancellor Award at the 48<sup>th</sup> UTM convocation in 2012.





**Ali Emrouznejad** is a Professor and Chair in Business Analytics at Aston Business School, UK. His areas of research interest include performance measurement and management, efficiency and productivity analysis as well as data mining and big data. He holds an MSc in applied mathematics and received his PhD in operational research and systems from Warwick Business School, UK. Having got PhD in the area of DEA, in 1998, he has joined to the

development team of Performance Indicators (PI) in Higher Education (HE) at HEFCE (Higher Education Funding Council for England). The PI in HE has been widely published and it is now an annual publication of HESA (Higher Education Statistical Agency). He has also collaborated on research project titled “assessing cost efficiencies in higher education” funded by the Department for Education and Skills (DFES). His most research project is on “Analysis of efficiencies and productivity evolution in manufacturing industries with CO<sub>2</sub> emissions” funded by Royal Academy of Engineering.



**Ali Safaa Sadiq** is a senior IEEE member and currently a faculty member at Faculty of Science and Engineering, School of Mathematics and Computer Science, University of Wolverhampton, UK; he is also an adjunct staff at Monash University and Centre for Artificial Intelligence Research and Optimisation, Torrens University Australia. Ali has served as a lecturer at the School of Information Technology, Monash University, Malaysia. Previously he


has also served as a senior lecturer at the Department of Computer Systems & Networking Department, Faculty of Computer Systems & Software Engineering, University Malaysia Pahang, Malaysia. Ali has received his PhD, M.Sc, and B.Sc degrees in Computer Science in 2004, 2011, and 2014 respectively. Ali has been awarded the Pro-Chancellor Academic Award as the best student in his batch for both Masters and PhD. Ali has also been awarded the UTM International Doctoral Fellowship (IDF). He has published several scientific/research papers in well-known international journals and conferences. He was involved in conducting 5 research grants projects, whereby 3 of them are in the area of network and security and the others in analyzing and forecasting floods in Malaysia. He has supervised 3 PhD students and 3 Masters students as well as some other undergraduate final year projects. His current research interests include Wireless Communications, Network security and AI applications in networking.



**Danda B. Rawat** is a Full Professor in the Department of Electrical Engineering & Computer Science (EECS), Founder and Director of the Howard University Data Science and Cybersecurity Center, Director of Cyber-security and Wireless Networking Innovations (CWiNs) Research Lab, Graduate Program Director of Howard CS Graduate Programs and Director of Graduate Cybersecurity Certificate Program at Howard

University, Washington, DC, USA. Dr. Rawat is engaged in research and teaching in the areas of cybersecurity, machine learning, big data analytics and wireless networking for emerging networked systems including cyber-physical systems, Internet-of-Things, multi domain battle, smart cities, software defined systems and vehicular networks. His professional career comprises more than 18 years in academia, government, and industry. He has secured over \$16 million in research funding from the US National Science Foundation (NSF), US Department of Homeland Security (DHS), US National Security Agency (NSA), US Department of Energy, National Nuclear Security Administration (NNSA), DoD and DoD Research Labs, Industry (Microsoft, Intel, etc.) and private Foundations. Dr. Rawat is the recipient of NSF CAREER Award in 2016, Department of Homeland Security (DHS) Scientific Leadership Award in 2017, Researcher Exemplar Award 2019 and Graduate Faculty Exemplar Award 2019 from Howard University, the US Air Force Research Laboratory (AFRL) Summer Faculty Visiting Fellowship in 2017, Outstanding Research Faculty Award (Award for Excellence in Scholarly Activity) at GSU in 2015, the Best Paper Awards (IEEE CCNC, IEEE ICII, BWCA) and Outstanding PhD Researcher Award in 2009. He has delivered over 20 Keynotes and invited speeches at international conferences and workshops. Dr. Rawat has published over 200 scientific/technical articles and 10 books. He has been serving as an Editor/Guest Editor for over 50 international journals including the Associate Editor of IEEE Transactions of Service Computing, Editor of IEEE Internet of Things Journal, Associate Editor of IEEE Transactions of Network Science and Engineering and Technical Editors of IEEE Network. He has been in Organizing Committees for several IEEE flagship conferences such as IEEE INFOCOM, IEEE CNS, IEEE ICC, IEEE GLOBECOM and so on. He served as a technical program committee (TPC) member for several international conferences including IEEE INFOCOM, IEEE GLOBECOM, IEEE CCNC, IEEE GreenCom, IEEE ICC, IEEE WCNC and IEEE VTC conferences. He served as a Vice Chair of the Executive Committee of the IEEE Savannah Section from 2013 to 2017. Dr. Rawat received the Ph.D. degree from Old Dominion University, Norfolk, Virginia. Dr. Rawat is a Senior Member of IEEE and ACM, a member of ASEE and AAAS, and a Fellow of the Institution of Engineering and Technology (IET).

## Affiliations

Eghbal Hosseini<sup>1</sup> · Kayhan Zrar Ghafoor<sup>2,3</sup> · Ali Emrouznejad<sup>4</sup> · Ali Safaa Sadiq<sup>5,6</sup> · Danda B. Rawat<sup>7</sup> 

Eghbal Hosseini  
kseghbalhosseini@gmail.com

Kayhan Zrar Ghafoor  
kayhan@ieee.org

Ali Emrouznejad  
a.emrouznejad@aston.ac.uk

Ali Safaa Sadiq  
ali.sadiq@wlv.ac.uk

- <sup>1</sup> Mechanical and Energy Engineering Department, Erbil Technical Engineering College, Erbil Polytechnic University, Kurdistan Region, Iraq
- <sup>2</sup> Department of Software, Informatics Engineering, Salahaddin University-Erbil, Erbil, 44001, Iraq
- <sup>3</sup> School of Mathematics and Computer Science, University of Wolverhampton, Wulfruna Street, Wolverhampton, WV1 1LY, UK
- <sup>4</sup> Operations and Information Management, Aston Business School, Aston University, Birmingham, UK
- <sup>5</sup> Wolverhampton Cyber Research Institute, School of Mathematics and Computer Science, University of Wolverhampton, Wolverhampton, WV1 1LY, UK
- <sup>6</sup> Centre for Artificial Intelligence Research and Optimisation, Torrens University Australia, Fortitude Valley, Brisbane, 4006, QLD, Australia
- <sup>7</sup> Department of Electrical Engineering and Computer Science at Howard University, Washington, DC 20059, USA