Improved Gazelle Optimisation Algorithm

Prof. Vijay Kumar Bohat, Ayush Arora,Saaz Gupta,Aayush Bagga

Netaji Subhas University of Technology, Dwarka, Delhi, 110078, India

Abstract

In this study, we present an enhanced version of the Gazelle Optimization Algorithm (GOA) by integrating predators into the optimization framework. Inspired by the dynamic interplay between predators and prey in natural ecosystems, our Improved Gazelle Optimization Algorithm (IGOA) introduces a novel predator-prey dynamic, where both predators and prey engage in exploration and exploitation behaviors. During the exploration phase, predators actively pursue prey, simulating the predatory chase observed in nature. Conversely, in the exploitation phase, predators adopt a Brownian motion-like movement pattern, mimicking the wandering behavior as they exploit local resources. This predator-prey interaction enriches the algorithm's exploration-exploitation balance, enhancing its ability to navigate complex optimization landscapes. Through rigorous experimentation on benchmark functions and engineering design problems, we assess the performance of IGOA against state-of-the-art algorithms. Our results demonstrate the efficacy of IGOA in achieving superior solution quality, convergence speed, and robustness, showcasing its potential as a versatile optimization tool across diverse domains. This research extends the frontier of metaheuristic optimization techniques by integrating predator-prey dynamics, providing practitioners with a powerful approach to address complex optimization challenges effectively.

Keywords: Gazelle optimization algorithm; GOA ; IGOA ; Metaheuristics; Optimization problems; ; Nature-inspired ; Gazelle ; Population-based

1. Introduction

Optimization problems are pervasive across numerous fields, ranging from engineering and medicine to computer science and supply chains. These problems entail the quest for optimal solutions within specified constraints, a task compounded by their inherent complexity, nonlinearity, and multimodality. Traditional optimization approaches often fall short in addressing these challenges, prompting the adoption of metaheuristic algorithms, particularly those inspired by nature.

Nature-inspired metaheuristic algorithms, leveraging evolutionary or exploratory principles, have gained popularity for their ability to tackle optimization problems. Drawing inspiration from natural phenomena such as hunting, foraging, mating, and survival strategies observed in various species, these algorithms offer a diverse toolkit for problem-solving. They are classified into different categories, including evolutionary algorithms, swarm intelligence algorithms, physical algorithms, and those based on human behavior, reflecting the breadth of their inspirations.

Despite their widespread adoption and success in applications such as disease classification and clustering, questions linger regarding the fidelity of these algorithms to natural processes and their practical effectiveness. The rapid proliferation of new algorithms further complicates matters, as claims of novelty and superior performance often outpace empirical validation. Moreover, the No Free Lunch theorem underscores the absence of a one-size-fits-all optimization solution, highlighting the need for tailored methods to address specific problem domains.

Acknowledging the efficacy of existing optimization methods within certain contexts, it is essential to recognize their limitations and the imperative of ongoing innovation. This study contributes to this discourse by presenting an enhanced metaheuristic algorithm, aiming to bridge the gap between nature-inspired principles and practical optimization challenges. By advancing

Email address: vijay.bohat@nsut.ac.in (Ayush Arora,Saaz Gupta,Aayush Bagga)

the field of metaheuristic optimization, this research seeks to provide more robust and effective problem-solving strategies for diverse real-world applications.

The Gazelle Optimization Algorithm (GOA) is one such population-based optimization algorithm inspired by gazelles' survival ability in their predator-dominated environment. Introduced in 2022 by Jeffrey O. Agushaka, Absalom E. Ezugwu, and Laith Abualigah. The technical contributions of this research are summarized as follows:

- This study introduces the Improved Gazelle Optimization Algorithm (IGOA), a novel population-based metaheuristic algorithm designed to emulate the survival instincts of gazelles in their natural environment.
- Drawing inspiration from the gazelle's ability to detect and evade predators, the IGOA incorporates two distinct phases representing exploration and exploitation.
- During the exploitation phase, the algorithm emulates the calm grazing behavior of gazelles, occurring either when predators are stalking them or have yet to be detected. Upon identification of a predator, the IGOA seamlessly transitions into the exploration phase, where the gazelle actively evades and outmaneuvers the threat to find refuge. These phases are iteratively executed, governed by termination criteria, to pursue optimal solutions for various optimization problems.
- The distinct stages of the IGOA are delineated and formalized through mathematical modeling.
- The efficacy and robustness of the IGOA are assessed by solving a comprehensive set of 60 benchmark functions from the CEC2014 and CEC2017 competitions.
- Furthermore, the IGOA's performance in solving real-world engineering design problems is evaluated across five distinct scenarios.
- To gauge its effectiveness, the optimization results obtained from the IGOA are benchmarked against those of six state-of-the-art algorithms and the original Gazelle Optimization Algorithm (GOA).

2. IMPROVED GAZELLE OPTIMISATION ALGORITHM

2.1. Inspiration and behaviour of gazelle

The gazelles belong to the genus Gazella family. About 19 different gazelles exist globally, ranging from small gazelles like Thomson's and Speke's gazelle to the large gazelle-like the Dama gazelle. Gazelles are categorized as herbivores and primarily consume vegetation such as leaves, grasses, and shoots. They exhibit a social behavior, often forming large groups for security and social interaction, with group sizes reaching up to 700 individuals. Within these groups, there can be segregation based on sex, with females and their fawns forming smaller groups, while bachelor herds consist solely of males who provide defense and support. Their reproductive cycle involves a birth rate of one or two offspring, occurring twice annually, with a gestation period of approximately six months. Breeding typically coincides with the rainy season when food and water are plentiful.

As secondary consumers, gazelles occupy a critical position in the food chain, serving as primary prey for various predators including humans, cheetahs, jackals, hyenas, wild dogs, leopards, and lions. To evade predation, gazelles employ warning signals such as tail flicking, foot stomping, and leaping, known as "stotting," when nervous or excited. They also possess remarkable agility and speed, capable of reaching speeds up to 100 km/hr, which often enables them to outrun their predators. Despite their vulnerability, gazelles maintain a survival rate of 0.66 annually, indicating that predators are successful in predation only 0.34 of the time.

These survival strategies of gazelles serve as a basis for developing a model algorithm aimed at enhancing their survival in natural environments:-

- The most spectacular aspects are grazing and running from predators.
- Predator positions are changed with respect to best position of a gazelle.
- The grazing aspect of gazelle while predator is stalking it can be used for exploration phase
- The ability to outrun spotted predators to a haven can be used for exploration.

2.2. Mathematical model of proposed IGOA

The IGOA is a population-based optimization algorithm that uses randomly initialized search agents.The search agents consist of gazelles and predators in a ratio of 4:1.The search agents are defined as a n X d matrix of candidate solutions as defined in Eq. The IGOA uses the problem's constraint of upper bound (UB) and lower bound (LB) to stochastically define the range of values that the population vector can take.

$$X = \begin{bmatrix} x_{1,1} & \dots & x_{1,j} & \dots & x_{1,d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i,1} & \dots & x_{i,j} & \dots & x_{i,d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N,1} & \dots & x_{N,j} & \dots & x_{N,d} \end{bmatrix}$$

where X is the matrix of the position vector of the candidate population, each position vector is stochastically generated.,$x_{i,j}$ is the randomly generated vector position of the ith population in the jth dimension, N represents the number of search agents, and d is the defined search space (dimension) of the optimization problem. Predators are randomly allocated twenty percent of the indices in the matrix representing predator populations.

$$x_{i,j} = l_j + \text{rand} \cdot (u_j - l_j), \quad i = 1, 2, \dots, N, \quad j = 1, 2, \dots, d,$$

In every iteration, each $x_{i,j}$ produces a candidate solution. The minimum solution is taken to be the best-obtained solution so far. It is said that the strongest or fittest gazelles in nature are more talented in spotting, informing others of the treats, and running from predators. The best-obtained solution so far is selected as the top gazelle to construct an Elite N X d matrix. This matrix is utilized for searching and determining the next step for the predators, taking into account the distance from each gazelle and the distance from the elite matrix. It is also employed in updating the positions of the gazelles.

$$X = \begin{bmatrix} x'_{1,1} & \dots & x'_{1,j} & \dots & x'_{1,d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x'_{i,1} & \dots & x'_{i,j} & \dots & x'_{i,d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x'_{N,1} & \dots & x'_{N,j} & \dots & x'_{N,d} \end{bmatrix}$$

The IGOA considers both predators and gazelles as search agents constituting a population of 1:4 respectively.When a predator stalks a gazelle, it employs Brownian motion tactics during the exploitation phase.The predator stealthily approaches the gazelles, moving towards elite positions.Upon detection, both predator and prey flee in unison toward the haven. As the gazelles evade capture, the predator effectively explores the search space.

2.2.1. Brownian motion

In a standard Brownian motion, the displacement follows a Normal (Gaussian) probability distribution function with a mean of $\mu = 0$ and a variance of $\sigma^2 = 1$ at point x. It can be represented by the equation [8] :

$$f(x|\mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where $f(x|\mu, \sigma^2)$ represents the probability density function of the normal distribution.

2.2.2. Levy flight

The Levy flight performs a random walk using the Levy distribution (power-law tail) given as in [9]

$$L(x_j) \approx \left|x_j\right|^{1-\alpha}$$

where x_j denotes the flight distance and α=(1,2] represents the power law exponent.Levy stable process can be denoted as an integral by [10]:-

$$F_L(x; \alpha, \gamma) = \frac{1}{\pi} \int_0^\infty exp(-\gamma q^\alpha) cos(qx)\, dx$$

where α is the distribution index that controls everything about the motion, and γ denotes the scale unit. Our work used an algorithm that generates a stable Levy motion proposed by GOA. The algorithm uses α within the range of 0.3–1.99, and it is defined as in [10]:-

$$Levy(\alpha) = 0.05X \frac{x}{\left| y^{\frac{1}{\alpha}} \right|}$$

where α,x and y are defined as follows:- x = Normal(0,σ_x^2) and y = Normal (0, σ_y^2)

$$\sigma_x = \left[\frac{\Gamma(1+\alpha) sin(\frac{\pi\alpha}{2})}{\Gamma(\frac{1+\alpha}{2}) \alpha 2^{\frac{(\alpha-1)}{2}}} \right]^{\frac{1}{\alpha}}$$

σ_y=1 and α=1.5

2.2.3. Gazelle grazing peacefully (Exploitation phase)

This phase assumes the gazelles are grazing peacefully while the predator is stalking the gazelles. In this phase, the Brownian motion characterized by uniform and controlled steps was used to effectively cover neighborhood areas of the domain. The gazelles are assumed to move in Brownian motion while grazing, as depicted in

$$gazelle_{i+1} = gazelle_i + s.R * R_B * .(Elite_i - R_b * .gazelle_i)$$

where $gazelle_{i+1}$ refers to position of gazelle in next iteration,$gazelle_i$ is solution at the current iteration, s denotes the grazing speed of gazelles, $\vec{R_B}$ is a vector containing random numbers representing the Brownian motion,R is a vector of uniform random numbers [0,1].

When a predator stalks a gazelle, it employs Brownian motion tactics during the exploitation phase.The predator's proximity to each gazelle is measured, and its movements are adjusted based on this proximity and the strategic positions of the elite gazelles.The predator stealthily approaches the gazelles, moving towards elite positions as depicted in

$$gazelle_{i+1} = Elite_i + sPredator.R * sign(R_B * .(Gazelle_l * R_B - gazelle_i))$$

where $gazelle_{i+1}$ refers to position of predator in next iteration,$gazelle_i$ is solution at the current iteration, $gazelle_l$ refers to position of other gazelles. $Elite_i$ is position of best fitness gazelle, sPredator denotes the stalking speed of predators, $\vec{R_B}$ is a vector containing random numbers representing the Brownian motion,R is a vector of uniform random numbers [0,1].

2.2.4. Gazelle running away from predator (Exploration phase)

The exploration phase begins immediately upon the predator being detected. Gazelles respond to threats by exhibiting various behaviors such as tail flicking, foot stomping, or leaping up to 2 meters in the air with all four feet, which is simulated by converting the 2-meter height into a scaled value between 0 and 1. This phase employs a Levy flight algorithm, characterized by small steps interspersed with occasional long jumps. Once the predator is sighted, the gazelle initiates a flight response, and the predator gives chase. To evade the predator, the gazelle employs sudden changes in direction, denoted by the parameter μ. This study assumes that such directional changes occur in every iteration; specifically, when the iteration number is odd, the gazelle moves in one direction, and when it is even, it moves in the opposite direction.The mathematical model describing the gazelle's behavior upon detecting the predator is depicted by the following equation:

$$gazelle_{i+1} = gazelle_i + S.\mu.R * .R_L * .(Elite_i - R_L * .gazelle_i)$$

where $gazelle_{i+1}$ represents position of gazelle in next iteration, $gazelle_i$ is solution at current iteration.$Elite_i$ is position of best fitness gazelle,S is top speed,μ represents direction of gazelle, $\vec{R_L}$ is a vector containing random numbers representing the Levy flight. During the exploration phase, predators utilize Levy flight patterns to pursue the leading gazelles of the herd. The distance between each predator and gazelle is calculated, and the predator's position is then adjusted according to its closeness to and distance from the foremost gazelle. To modify the predator's direction, the sign function is employed, which yields a value of 1 when the generated step size is positive and -1 when it is negative. The mathematical model for the behavior of the predator chasing the gazelle is shown in Eq

$$gazelle_{i+1} = Elite_i + SPredator.R * sign(R_L * .(Gazelle_l * R_L - gazelle_i))$$

where $gazelle_{i+1}$ refers to position of predator in next iteration,$gazelle_l$ refers to position of other gazelles,SPredator refers to top speed of predator,$gazelle_l$ refers to position of other gazelles.

PSRs is the predator success rates, the effect affects the ability of the gazelle to escape, which means the algorithm avoids being trapped in a local minimum. The PSRs effect is modeled as in Eq

$$gazelle_{i+1} = gazelle_i + CF(lb + R * (ub - lb)) * .U \ if \ r \leq PSRs$$

$$gazelle_i + (PSRs(1 - r) + r)(gazelle_{r_1} - gazelle_{r_2}) \ else$$

where $\vec{U}$ denotes a binary vector which is constructed by r in [0,1] such that $\vec{U}$=0 when $r < 0.34$ and 1 otherwise

2.3. Pseudo-Code

The implementation flow of these phases, as defined by their respective mathematical model, is shown in the pseudocode for the IGOA given below:-

Algorithm 1: Pseudo-code of IGOA

```
Input: Optimization problem information
Output: Best candidate solution obtained by IGOA
IGOA () Input the optimization problem information;
s = [0, 1];
μ = [−1, 1];
S = 88 km/hr;
S_Predator = 115 km/hr;
s_Predator = [0, 1];
PSRs = 0.34;
R and r are random numbers [0, 1];
Determine the IGOA population size N and the number of iterations T;
Initialization of the position of gazelles and predators (search agents);
for t = 1 to T do
    Calculate the fitness of search agents , top_gazelle = fittest gazelle;
    Construct the elite matrix;
    for i = 1 to N do
        if gazelle then
            if r < 0.5 then
                In exploitation phase;
                    gazelle_{i+1} = gazelle_i + s.R * R_B * .(Elite_i − R_b * .gazelle_i)
            end
            else
                In exploration phase;
                Update value of μ to -1 for even iteration else to 1;
                    gazelle_{i+1} = gazelle_i + S.μ.R * .R_L * .(Elite_i − R_L * .gazelle_i)
            end
        end
        else
            for l = 1 to N do
                For every gazelle in population;
                if r < 0.5 then
                    Predator in exploitation phase;
                        gazelle_{i+1} = Elite_i + sPredator.R * sign(R_B * .(Gazelle_l * R_B − gazelle_i))
                end
                else
                    Predator in exploration phase;
                        gazelle_{i+1} = Elite_i + SPredator.R * sign(R_L * .(Gazelle_l * R_L − gazelle_i))
                end
            end
        end
        Fitness update;
        top-gazelle update;
        Applying PSRs effect and update based on;
            gazelle_{i+1} = gazelle_i + CF(lb + R * (ub − lb)) * .U if r ≤ PSRs
                            gazelle_i + (PSRs(1 − r) + r)(gazelle_{r_1} − gazelle_{r_2}) else
    end
end
Output best candidate solution obtained by IGOA;
```

2.4. Computational complexity

The primary operations within the GOA optimization process usually revolve around initializing solutions, evaluatingI fitness functions, and iteratively updating solutions.The updating procedure is driven by equations designed to explore optimal positions ensuring the best possible solution. Consequently, these equations also guide the adjustment of other solution positions towards the optimal one. Thus, the computational complexity of the entire optimization process in IGOA is,

$$O(iter \times n \times n \times d + (CFE \times n))$$

where CFE refers to the cost of function evaluation, d refers to dimensions and n refers to search agents.

3. Literature Survey

3.1. Literature Review: Metaheuristic Algorithms in Optimization

Optimization problems are ubiquitous across various scientific and engineering disciplines. They involve finding the best solution (minimum or maximum value) for an objective function while adhering to specific constraints. Traditional deterministic methods often struggle with complex, non-linear problems due to computational limitations or the risk of getting stuck in local optima (non-ideal solutions). Artificial intelligence (AI) offers promising techniques for tackling these challenges. AI-driven optimization algorithms leverage machine learning, evolutionary computing, and other AI subfields to find optimal solutions efficiently. Metaheuristic algorithms have found application in solving optimization problems, among which are nature-inspired metaheuristic algorithms which are popular nowadays among researchers. These successes have been attributed to the manner these algorithms solve the problems [21]. The nature-inspired metaheuristic algorithms mimic natural human and animal behavior and physical phenomena [24]. These algorithms have mimicked hunting, foraging, et al. mating, mutation, survival, swarming, and nesting to solve optimization problems [14, 22].

Nature-inspired metaheuristic algorithms use the evo- lutionary or exploratory approach to find optimal solutions to solve these problems [23]. This has led to a taxonomy of metaheuristic algorithms based on evolutionary algorithms [30], swarm intelligence algorithms [38], physical algorithms [32], and algorithms based on human behavior [29]. So many other taxonomies exist in the literature, a comprehensive consensus classification for metaheuristic algorithms has not yet been introduced. A contributing factor is the rate at which new algorithms are being proposed. The application of these metaheuristic algorithms and other computation intelligence methods in disease classification, clustering, and many others is well documented in the literature [13, 15].

A detailed taxonomy of metaheuristic algorithms can be found in [24]. However, an abridged version of the taxonomy is given in Fig. 1 to enhance our discussion in this section.

Evolutionary algorithms: These methods mimic Charles Darwin's ideas of evolution. They start with random solutions and improve them over time using rules specific to each algorithm. Examples include genetic algorithms (GAs)[25], differential evolution (DE)[35], and artificial algae algorithms (AAAs)[36]. Physical methods: These algorithms are inspired by physical laws like those governing temperature changes (simulated annealing)[28] or gravity (gravitational search algorithm)[19 33].

Swarm-based methods: These algorithms take inspiration from how animals like bees or ants work together to find food or build nests. Examples include particle swarm optimization (PSO)[37], artificial bee colony (ABC)[16,17], and ant colony optimization (ACO).[20]

Other inspirations: Some algorithms mimic human behavior (teaching-learning-based optimization, TLBO)[34], social dynamics (imperialist competitive algorithm, ICA)[18], or plant growth (invasive weed optimization, IWO)[37].

New metaheuristic algorithms are being proposed at a high rate; most have a different source of inspiration, while others improve existing ones. The arithmetic optimization algorithm (AOA) was proposed by [12] and uses the dis- tributive power of simple arithmetic operators to find an optimal solution for optimization problems. Using advanced arithmetic operators like the natural logarithm and exponential operators, [44] proposed the nAOA, which enhanced the exploratory and exploitation ability of the original AOA. Similarly, the krill herd algorithm (KH) was improved using the low discrepancy sequences for the initial population [13, 15]. A novel optimization method called ebola optimization search algorithm (EOSA), which mimics how the Ebola virus spreads, was proposed by [31]. Others include [26, 39].

3.2. Motivation

The original Gazelle Optimization Algorithm (GOA) presented a novel concept, but its initial assumption regarding predator and prey populations might not be the most efficient. The GOA divided the search agents in half, with one group representing gazelles fleeing predators and the other group representing the predators themselves. However, this approach may not reflect real-world scenarios where gazelle populations typically outnumber predators in a given area. Recognizing this, we explored a revised population ratio of 4:1 gazelles to predators, aiming to create a more ecologically realistic model. Furthermore, we hypothesized that incorporating predator awareness into the gazelle's behavior could significantly improve convergence speed. By allowing the gazelles to react strategically and flee directly away from perceived threats, we envisioned a more efficient search process that would converge upon optimal solutions faster than the baseline algorithm. These refinements sought to enhance the GOA's effectiveness by grounding its inspiration in a more realistic understanding of predator-prey dynamics and incorporating a more strategic response from the gazelles.

4. Algorithms, test problems and comparison criteria

4.1. Algorithms and comparison criteria

This section presents the design of experiments conducted to evaluate the performance of IGOA. Thirty(30) CEC 2014 test functions,thirty(30) CEC 2017 test functions, and five (5) selected optimization design problems in the engineering domain were used to evaluate the IGOA. The results obtained were compared with that of the following algorithms:-

- Gazelle Optimization Algorithm (GOA) [1]
- Sand Cat Optimization Algorithm (SCSO) [4]
- Arithmetic Otimization Algorithm (AOA) [6]
- Fire Hawk Optimizer (FHO) [3]
- Whale Optimization Algorithm (WOA) [5]
- Grey Wolf Optimizer (GWO) [2]
- Crayfish Optimization Algorithm (COA) [7]

The population size and the maximum number of iterations are sensitive parameters and need to be tuned. For this study, the population size is tuned to 300, and the maximum number of iterations is tuned to 100.The stop criterium is the maximum number of iterations. The number of independent runs for each algorithm is set at 50.

4.2. CEC 2014 AND CEC 2017 Benchmark functions

Typology	No.	Function name	Opt.
Unimodal Functions	1	Shifted and Rotated Bent Cigar	100
	2	Shifted and Rotated Sum of Different Power	200
	3	Shifted and Rotated Zakharov	300
Simple Multimodal Functions	4	Shifted and Rotated Rosenbrock	400
	5	Shifted and Rotated Rastrigin	500
	6	Shifted and Rotated Expanded Schaffer F6	600
	7	Shifted and Rotated Lunacek Bi-Rastrigin	700
	8	Shifted and Rotated Non-Continuous Rastrigin	800
	9	Shifted and Rotated Levy	900
	10	Shifted and Rotated Schwefel	1000
Hybrid Functions	11	Zakharov; Rosenbrock; Rastrigin	1100
	12	High-conditioned Elliptic; Modified Schwefel; Bent Cigar	1200
	13	Bent Cigar; Rosenbrock; Lunacek bi-Rastrigin	1300
	14	High-conditioned Elliptic; Ackley; Schaffer F7; Rastrigin	1400
	15	Bent Cigar; HGBat; Rastrigin; Rosenbrock	1500
	16	Expanded Schaffer F6; HGBat; Rosenbrock; Modified Schwefel	1600
	17	Katsuura; Ackley; Expanded Griewank plus Rosenbrock; Schwefel; Rastrigin	1700
	18	High-conditioned Elliptic; Ackley; Rastrigin; HGBat; Discus	1800
	19	Bent Cigar; Rastrigin; Griewank plus Rosenbrock; Weierstrass; Expanded Schaffer F6	1900
	20	HappyCat; Katsuura; Ackley; Rastrigin; Modified Schwefel; Schaffer F7	2000
Composition Functions	21	Rosenbrock; High-conditioned Elliptic; Rastrigin	2100
	22	Rastrigin; Griewank; Modified Schwefel	2200
	23	Rosenbrock; Ackley; Modified Schwefel; Rastrigin	2300
	24	Ackley; High-conditioned Elliptic; Griewank; Rastrigin	2400
	25	Rastrigin; HappyCat; Ackley; Discus; Rosenbrock	2500
	26	Expanded Schaffer F6; Modified Schwefel; Griewank; Rosenbrock; Rastrigin	2600
	27	HGBat; Rastrigin; Modified Schwefel; Bent Cigar; High-conditioned Elliptic; Expanded Schaffer F6	2700
	28	Ackley; Griewank; Discus; Rosenbrock; HappyCat; Expanded Schaffer F6	2800
	29	f_{15}; f_{16}; f_{17}	2900
	30	f_{15}; f_{18}; f_{19}	3000

Figure 1: CEC2017 BENCHMARK FUNCTIONS

No.	Functions	$f(x^*)$	NO.	Functions	$f(x^*)$
F01	Rotated High Conditioned Elliptic Function	100	F16	Shifted and Rotated Expanded Scaffer's F6 Function	1600
F02	Rotated Bent Cigar Function	200	F17	Hybrid Function 2	1700
F03	Rotated Discus Function	300	F18	Hybrid Function 2	1800
F04	Shifted and Rotated Rosenbrock's Function	400	F19	Hybrid Function 3	1900
F05	Shifted and Rotated Ackley's Function	500	F20	Hybrid Function 4	2000
F06	hifted and Rotated Weierstrass Function	600	F21	Hybrid Function 5	2100
F07	Shifted and Rotated Griewank's Function	700	F22	Hybrid Function 6	2200
F08	Shifted Rastrigin's Function	800	F23	Composition Function 1	2300
F09	Shifted and Rotated Rastrigin's Function	900	F24	Composition Function 2	2400
F10	Shifted Schwefel's Function	1000	F25	Composition Function 3	2500
F11	Shifted and Rotated Schwefel's Function	1100	F26	Composition Function 4	2600
F12	Shifted and Rotated Katsuura Function	1200	F27	Composition Function 5	2700
F13	Shifted and Rotated HappyCat Function	1300	F28	Composition Function 6	2800
F14	Shifted and Rotated HGBat Function	1400	F29	Composition Function 7	2900
F15	Shifted and Rotated Expanded Griewank's plus Rosenbrock's Function	1500	F30	Composition Function 8	3000

Search Range: $[-100, 100]^D$

Figure 2: CEC2014 BENCHMARK FUNCTIONS

4.3. Engineering Problems

4.3.1. F3:Pressure vessel design

A pressure vessel design model (PVD) [40] consists of a cylinder closed with end caps. The PVD has four decision variables where x1 is the thickness of the pressure vessel Ts, x2 is the thickness of the head Th,x3 stands for the inner radius of the vessel R, and x4 is the length of the vessel barring head L. The mathematical model for PVD is given in Eq. (Sandgren, 1990). Given that

$$l = [l_1, l_2, l_3, l_4] = [T_s T_h R L],$$

,

$$Minf(\vec{l}) = 0.6224 l_1 l_3 l_4 + 1.781 l_2 l_3^2 + 3.1661 l_1^2 l_4 + 19.84 l_1^2 l_3$$

$$s_1(\vec{l}) = -l_1 + 0.0193 l_3 \leq 0,$$

$$s_2(\vec{l}) = -l_2 + 0.0095 l_3 \leq 0,$$

$$s_3(\vec{l}) = -\pi l_3^2 l_4 - \frac{4}{3}\pi l_3^3 + 1,296,000 \leq 0$$

$$s_4(\vec{l}) = l_4 - 240 \leq 0,$$

The interval of the design variables are as follows:-

$$0 \leq l_1 \leq 99$$

$$0 \leq l_2 \leq 99$$

$$10 \leq l_3 \leq 200$$

$$10 \leq l_4 \leq 200$$

4.3.2. F4:Three-bar truss design problem

Three bar truss design optimization problem [41] was firstly revealed by Ray and Saini.According to this, it is desired that three bars are placed. It is aimed to minimize the weight of bars in this position.This is a constraining optimization problem. There are two design parameters (x1, x2) and three restrictive functions in this problem. The problem is expressed mathematically as follows

$$min f(x) = (2\sqrt{2}x_1 + x_2)L$$

$$g_1 = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2}P - \sigma \leq 0$$

$$g_2 = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2}P - \sigma \leq 0$$

$$g_3 = \frac{1}{x_1 + \sqrt{2}x_2}P - \sigma \leq 0$$

where $0 \leq x_1, x_2 \leq 1$. The constraints are $L = 100\,\text{cm}$, $P = 2\,\text{KN/cm}^2$, and $\sigma = 2\,\text{KN/cm}^2$.

4.3.3. F6:Cantilever beam

In the current optimization application, the weight of a five-piece and fixed-connected cantilever beam that vertical P load applied from its end point, and consisted from divisions, which have hollowed square sections, was minimized [43]

$$Min f(x) = 0.0624(x_1 + x_2 + x_3 + x_4 + x_5)$$

$$0.01 \leq x \leq 100$$

$$g_1 = \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \leq 0$$

4.3.4. F8:Tubular column design

The objective of the tabular column design problem [42] is to minimize the cost of designing a uniform column of the tabular section, including material and construction costs. The column is made of material of length (L) with a yield stress (S), a modulus of elasticity (E), and a density (D) carrying a compressive load (P). This optimization problem has two optimized variables: the mean diameter of the column (x1) and tube thickness (x2). There are six constraints: the stress included in the column should be less than the buckling stress (g1(X)), and the yield stress (g2(X)); the mean diameter of the column is restricted between 2 and 14 cm (g3(X) and g4(X)); and columns with thickness outside the range of 0.2–0.8 cm are not commercially available (g5(X) and g6(X)).

$$Min f(x) = 9.8x_1x_2 + 2x_1$$

$$g_1 = 1.59 - x_1x_2$$

$$g_2 = 47.4 - (x_1^2 + x_2^2)x_1x_2$$

$$g_3 = \frac{2}{x_1} - 1$$

$$g_4 = \frac{x_1}{14} - 1$$

$$g_5 = \frac{2}{x_1} - 1$$

$$g_6 = \frac{x_1}{8} - 1$$

4.3.5. F13:Reinforced concrete beam design

Optimizing the design of reinforced concrete beams involves systematically improving their structural performance, efficiency, and reliability through mathematical modeling, analysis, and iterative design refinement. Reinforced concrete beams are essential structural elements widely used in building construction, bridges, and other infrastructure projects due to their strength, durability, and versatility. The objective function $f_{13}(x)$ for Beam Design is given by:

$$z = 29.4 \cdot \text{As} + 0.6 \cdot b \cdot h$$

where As is the area of steel reinforcement, b is the width of the beam, and h is the height of the beam. The constraints for the optimization problem are given as follows:

$$g_1(x) = \frac{b}{h} - 4 \leq 0$$
$$g_2(x) = 180 + 7.375 \cdot \text{As}^2/h - \text{As} \cdot b \leq 0$$

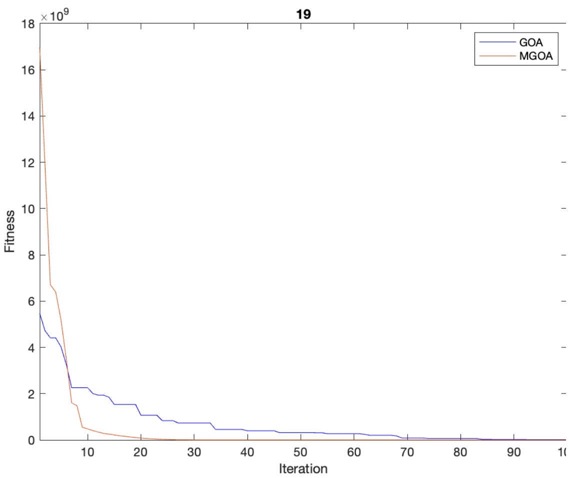
5. Experimental results and discussion

5.1. Performance Comparison of GOA and IGOA on CEC2014 and CEC2017

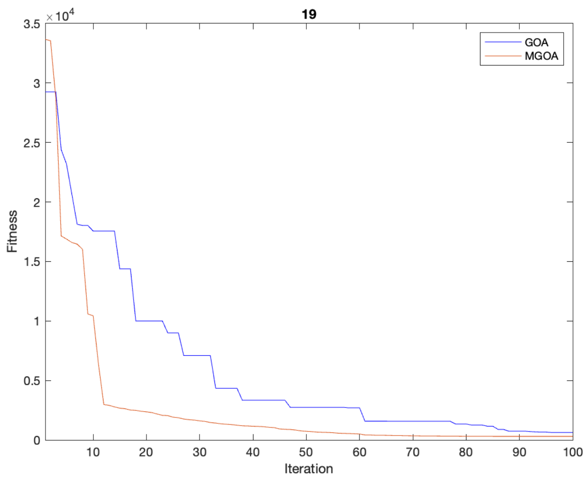
	CEC 2014				
	GOA		IGOA		
	MEAN	STD DEV	MEAN	STD DEV	P VALUE
F1	300880.276	114633.877	66023.8093	50989.2605	1.60E-16
F2	3994809.73	2975509.36	6061.32011	4794.45698	7.07E-18
F3	749.3972	263.851901	448.9648	80.0890259	5.84E-12
F4	408.6052	5.33730355	403.5622	3.58365581	1.80E-09
F5	519.6638	2.98670926	519.3356	3.40756132	2.71E-15
F6	601.638959	0.83059421	603.018705	1.03950425	2.13E-09
F7	701.0146	0.13592029	700.5066	0.28834935	1.58E-13
F8	805.9502	2.16471942	806.7314	2.87476637	2.81E-01
F9	914.082551	4.06381384	916.590102	5.6855826	3.02E-02
F10	1098.04293	61.3588524	1170.50471	120.774695	4.76E-03
F11	1602.82	211.540669	1589.818	215.482686	7.49E-01
F12	1200.614	0.16036947	1200.27	0.15811388	8.94E-14
F13	1300.24	0.05714286	1300.212	0.05584234	4.18E-03
F14	1400.174	0.05996598	1400.164	0.04848732	1.34E-01
F15	1502.298	0.56333192	1502.022	0.66156586	1.20E-02
F16	1602.664	0.33791679	1602.786	0.35629041	4.38E-02
F17	2571.7712	217.83451	2262.94088	162.199847	1.84E-11
F18	1885.526	23.4985871	1873.414	20.4020408	1.06E-02
F19	1902.084	0.34544263	1902.03	0.41661904	3.95E-01
F20	2054.04401	16.652488	2031.38912	11.2249492	3.02E-10
F21	2460.964	121.960723	2294.818	72.6055676	7.12E-11
F22	2223.858	2.65015132	2220.492	5.67428284	1.58E-07
F23	2577.0601	100.417283	2549.16488	123.99137	2.99E-05
F24	2519.574	5.25344629	2522.812	5.81474691	8.71E-03
F25	2641.76766	8.33748165	2636.94979	8.27588456	6.96E-03
F26	2700.22448	0.04760925	2700.18401	0.05327346	1.44E-04
F27	2704.41858	0.94737945	2706.41116	1.66376878	1.34E-09
F28	3177.76828	5.29103006	3175.83957	5.06837583	1.61E-01
F29	3310.46431	69.6638383	3215.38737	48.0666944	1.48E-10
F30	3618.9	68.2490069	3592.532	81.9177722	1.60E-02

	CEC 2017				
	GOA		IGOA		
	MEAN	STD DEV	MEAN	STD DEV	P VALUE
F1	6105200	4953295.799	3961.3	1925.703435	7.07E-18
F2	0	0	0	0	NA
F3	518.46	133.1589038	304.27	10.63107378	8.99E-18
F4	406.55	4.327215389	405.52	8.355688736	7.27E-05
F5	513.67	4.655558052	516.86	4.965044028	3.75E-03
F6	601.22	0.5542488942	603.79	2.146175802	2.68E-12
F7	733.68	8.241275421	735.5	9.79122853	2.66E-01
F8	813.98	5.060699238	817.43	5.958340507	6.40E-03
F9	903.34	3.209441588	945.8	38.03511685	1.73E-17
F10	1479.4	201.8103391	1434.7	190.5874005	1.96E-01
F11	1109.1	3.26273017	1107.9	3.962951384	3.70E-02
F12	104310	74553.7797	29835	20080.48125	1.76E-11
F13	1696.2	138.1582057	1494.5	67.68906173	7.25E-14
F14	1448.8	7.858623237	1436.8	5.97071937	7.12E-11
F15	1555.5	16.15404277	1524.8	8.703983606	2.71E-15
F16	1608.3	6.132869289	1610.5	9.985256274	6.97E-01
F17	1739.7	7.335788983	1736.2	6.985898625	2.81E-02
F18	2396.1	371.6164454	2000.3	80.62808484	4.70E-11
F19	1915.3	4.182533653	1909.8	2.451995439	6.49E-11
F20	2038.9	9.913533113	2042.5	10.43487109	2.00E-02
F21	0	0	0	0	NA
F22	0	0	0	0	NA
F23	2666.8	24.69255031	2672.1	7.686649173	4.84E-01
F24	2560.1	32.84298801	2513.6	37.61445792	5.86E-10
F25	2900.1	2.761248608	2898.8	2.57184203	2.86E-02
F26	2782.8	143.5231679	2780.7	148.4723896	1.15E-04
F27	3134.8	9.077507749	3133.1	11.01788694	2.87E-01
F28	3121.6	19.00034543	3110.4	17.89208421	3.08E-07
F29	3165	12.73453941	3160	18.7891193	2.45E-01
F30	9870.1	3471.086882	4699.2	1113.771427	5.24E-15

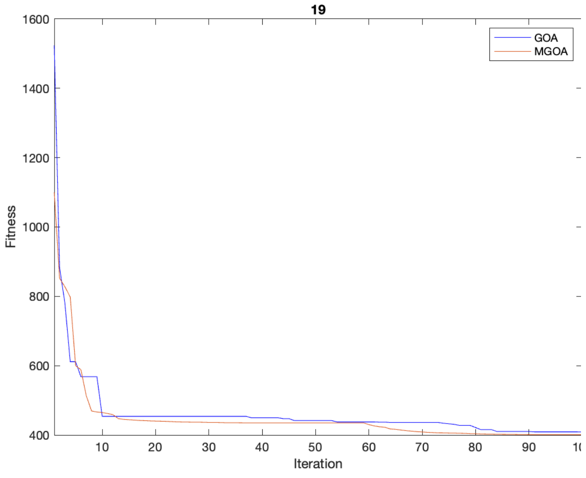
5.1.1. Convergence Curves for CEC2017



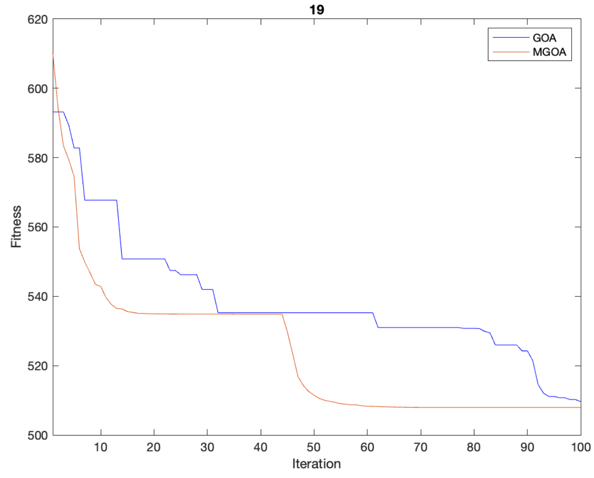
CEC2017-F1



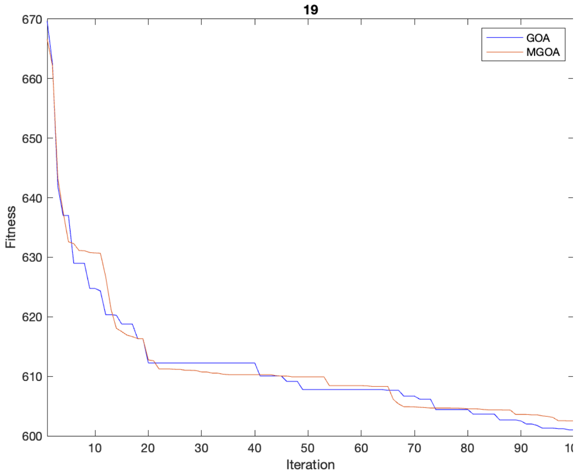
CEC2017-F3



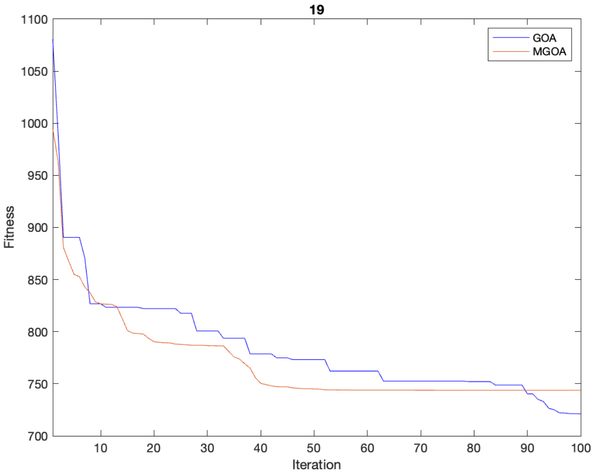
CEC2017-F4



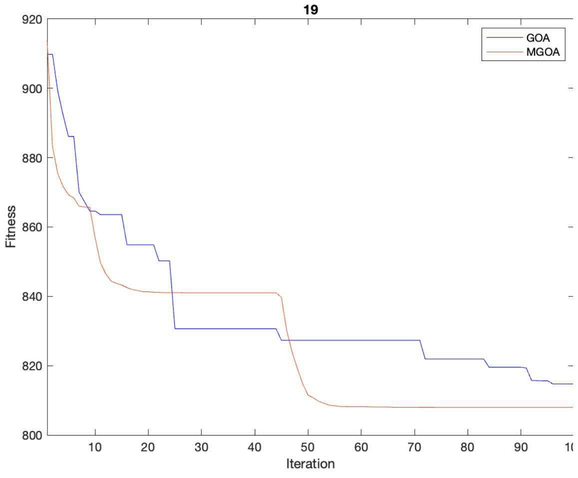
CEC2017-F5



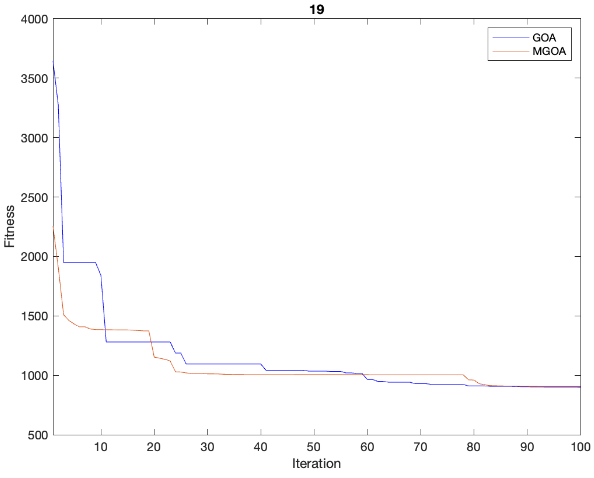
CEC2017-F6



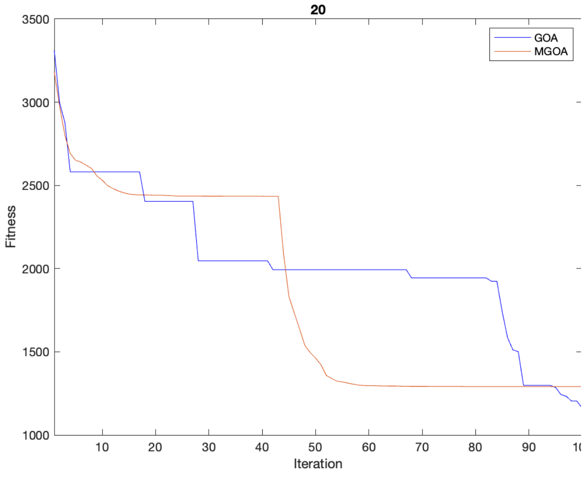
CEC2017-F7



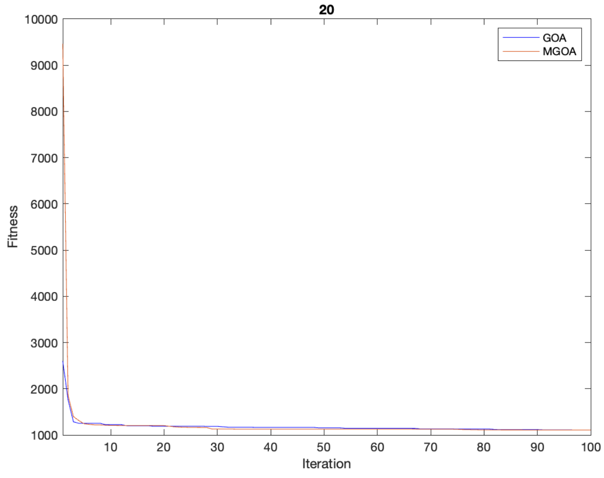
CEC2017-F8



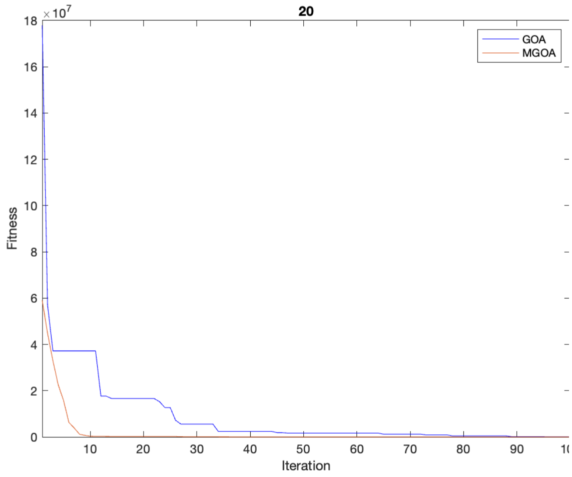
CEC2017-F9



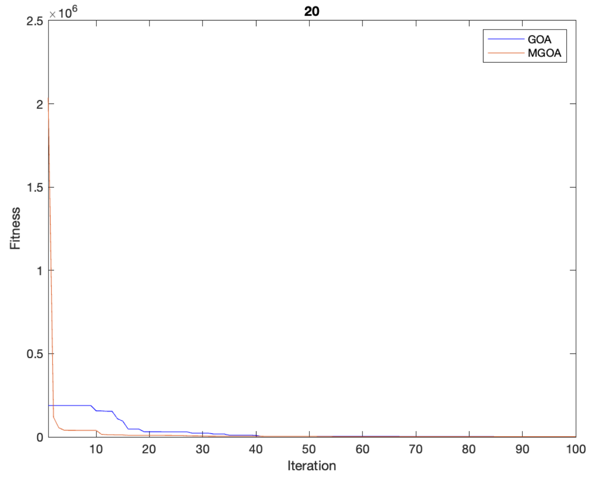
CEC2017-F10



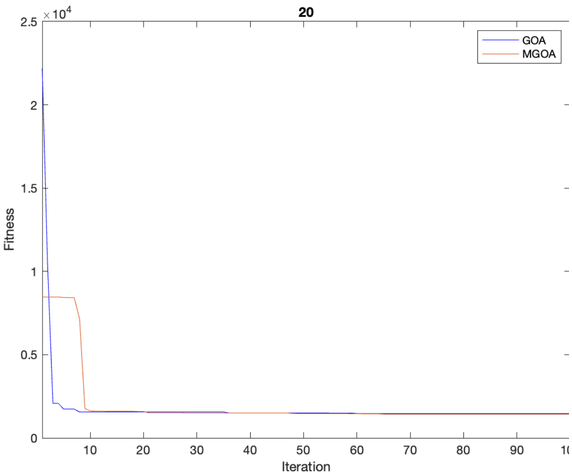
CEC2017-F11



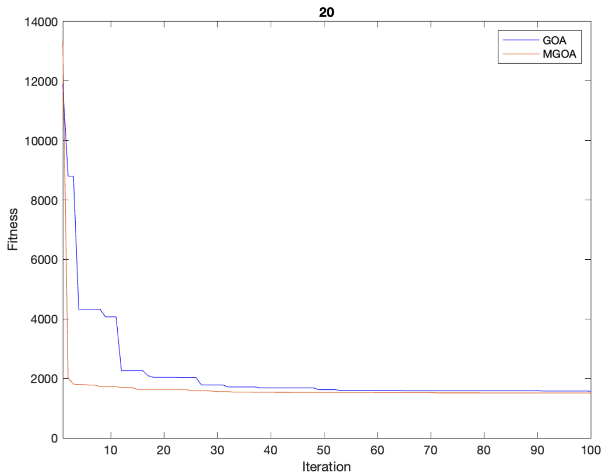
CEC2017-F12



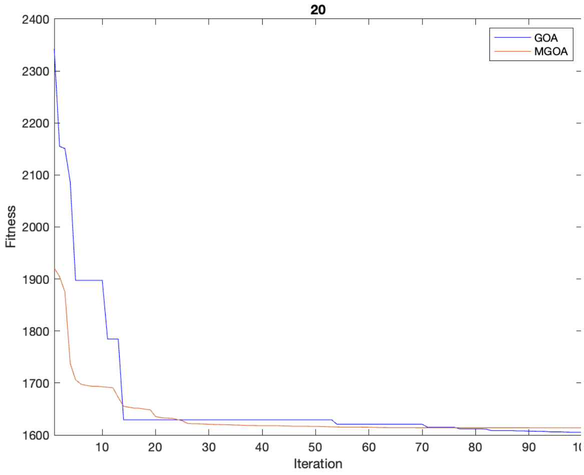
CEC2017-F13



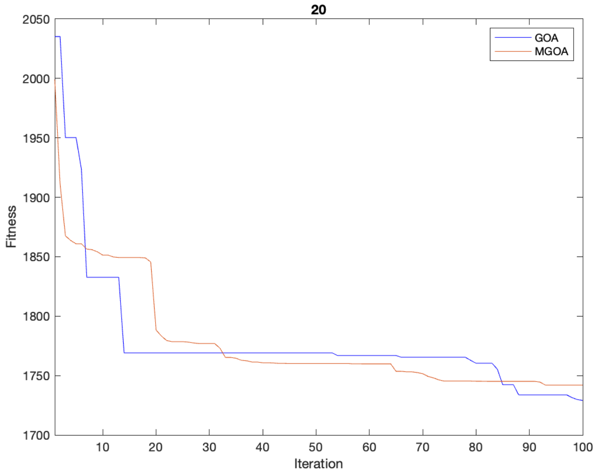
CEC2017-F14



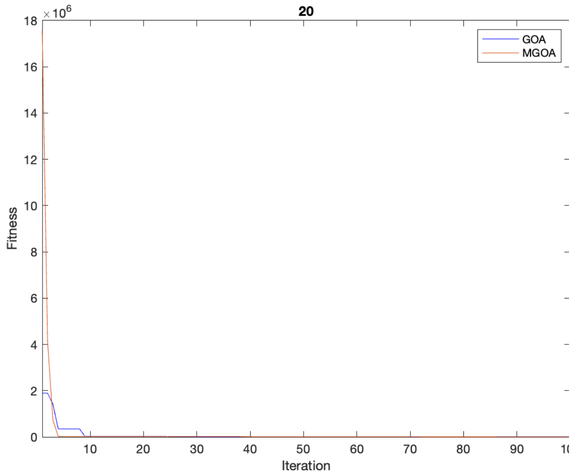
CEC2017-F15



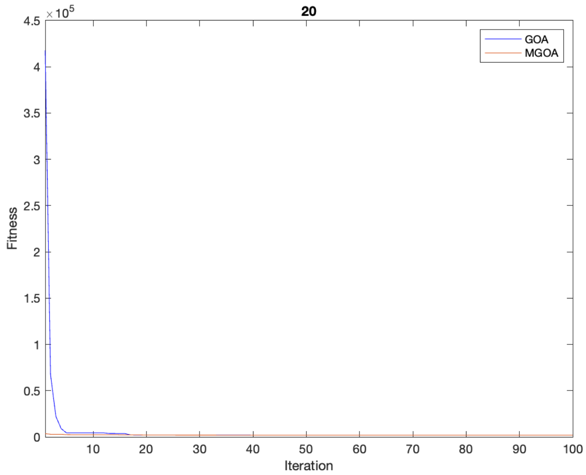
CEC2017-F16



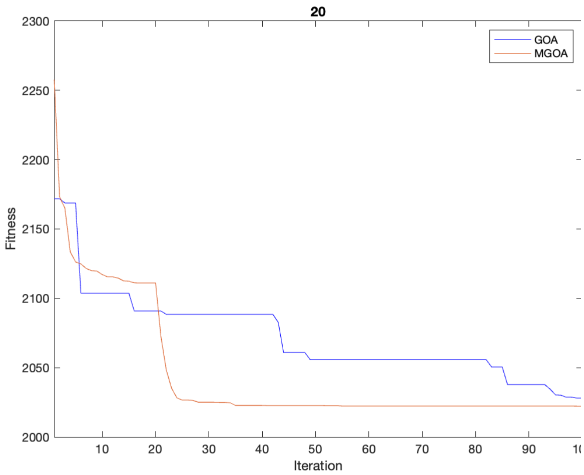
CEC2017-F17



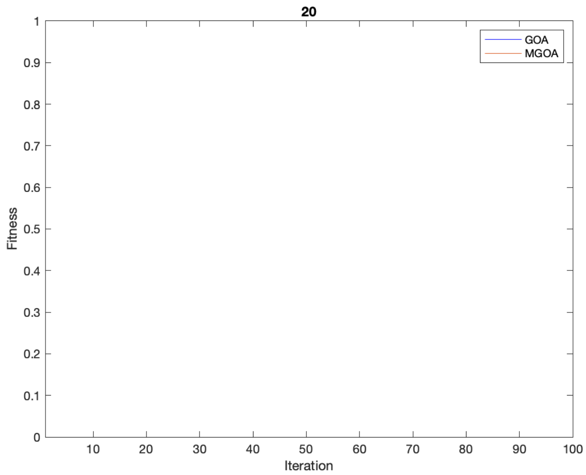
CEC2017-F18



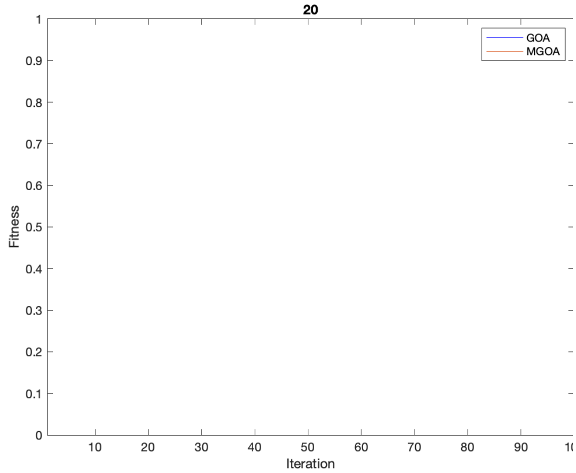
CEC2017-F19



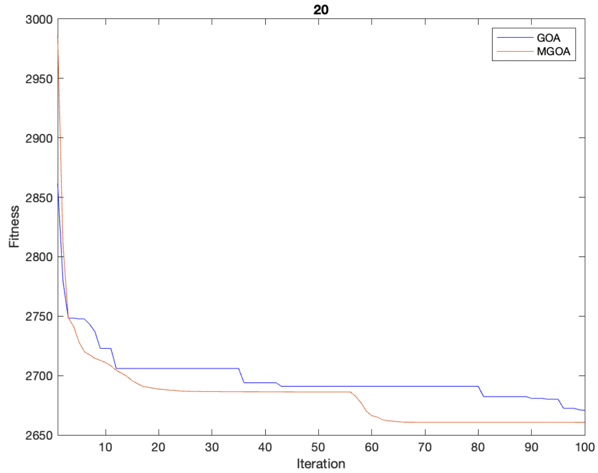
CEC2017-F20



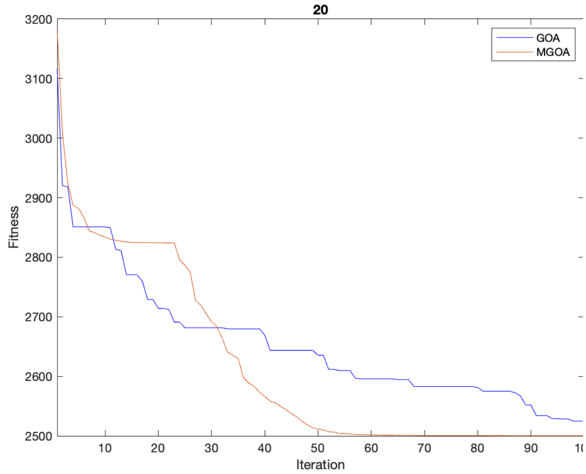
CEC2017-F21



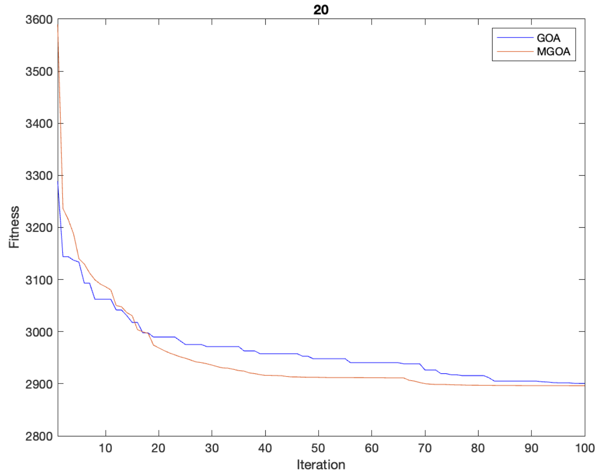
CEC2017-F22



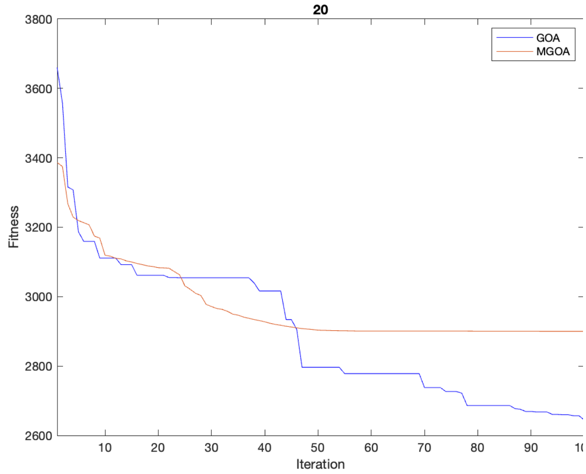
CEC2017-F23



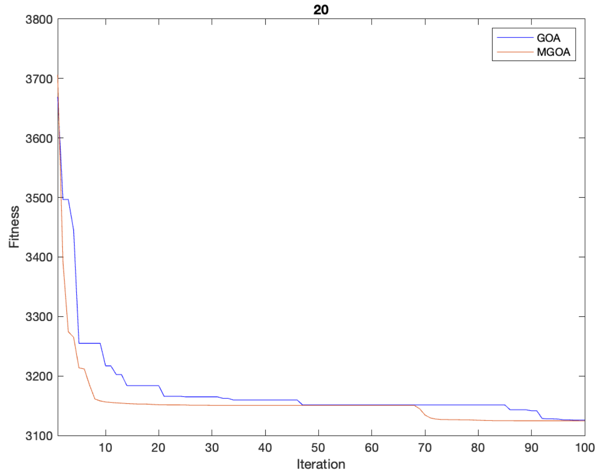
CEC2017-F24



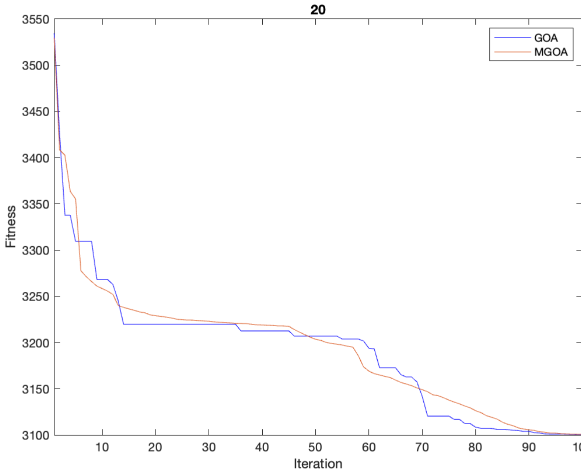
CEC2017-F25



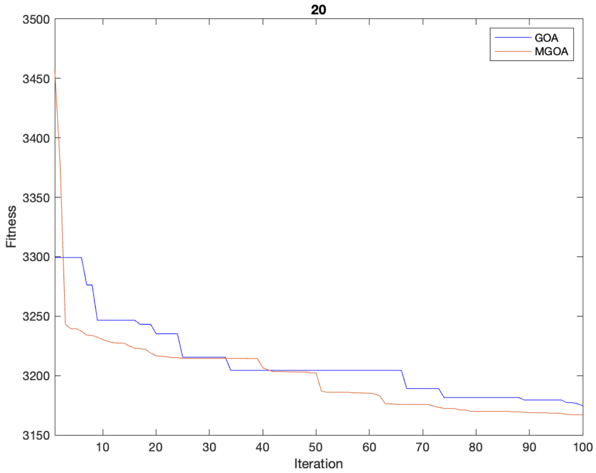
CEC2017-F26



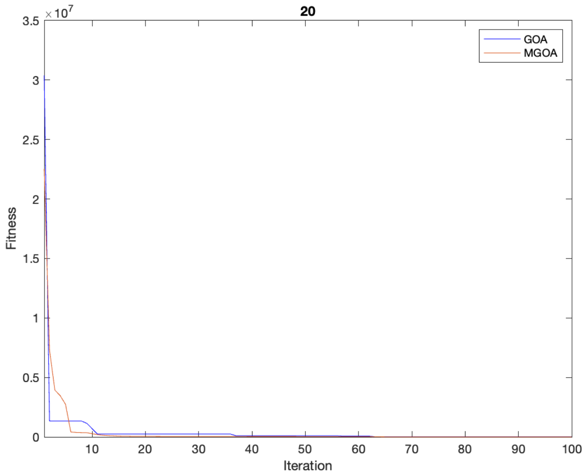
CEC2017-F27



CEC2017-F28

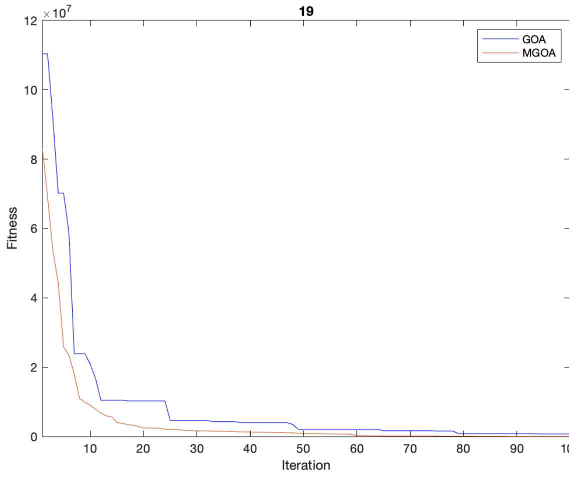


CEC2017-F29

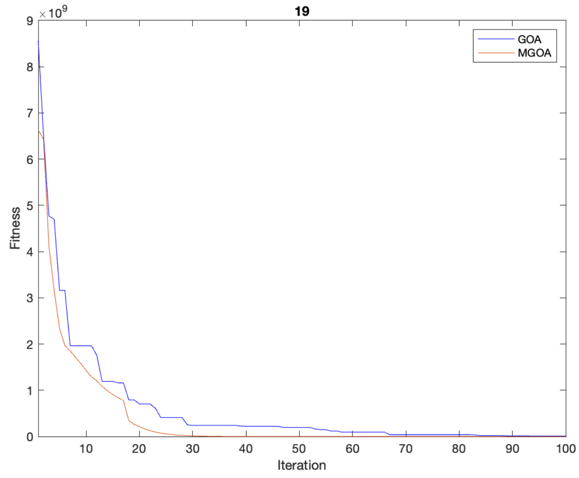


CEC2017-F30

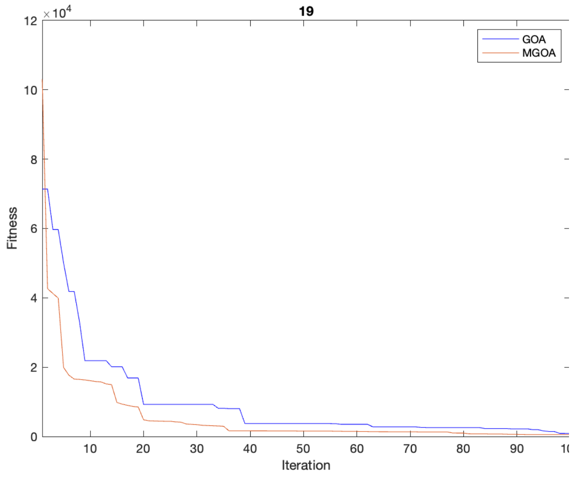
5.1.2. Convergence Curves for CEC2014



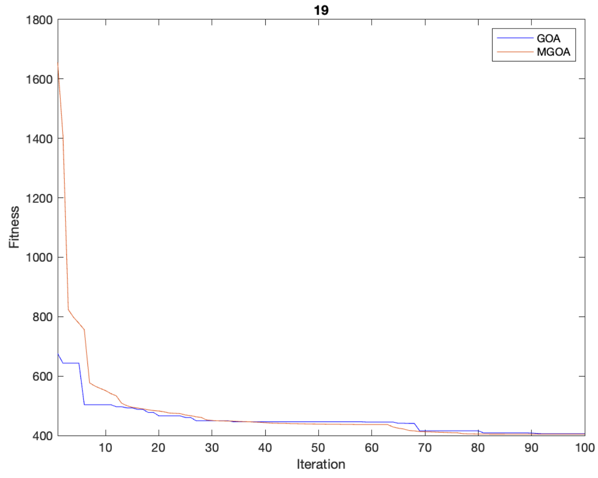
CEC2014-F1



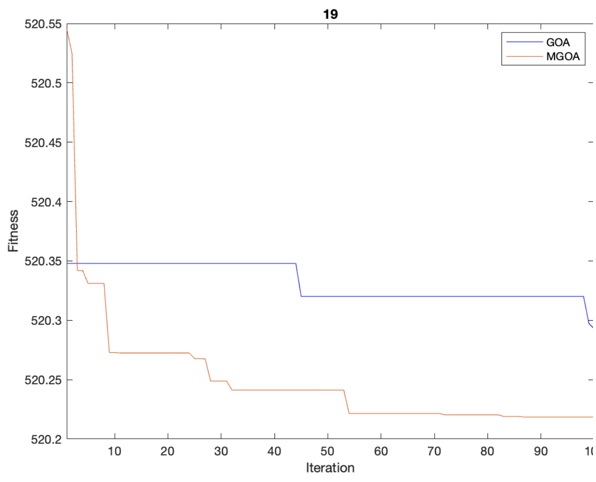
CEC2014-F2



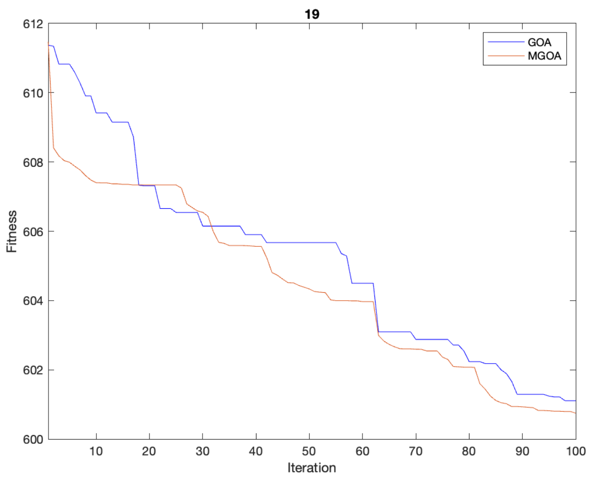
CEC2014-F3



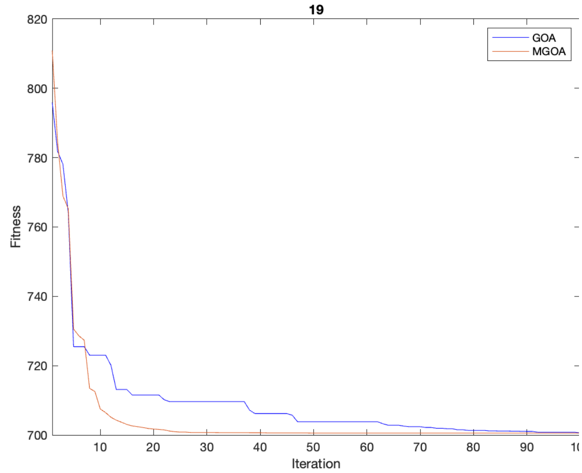
CEC2014-F4



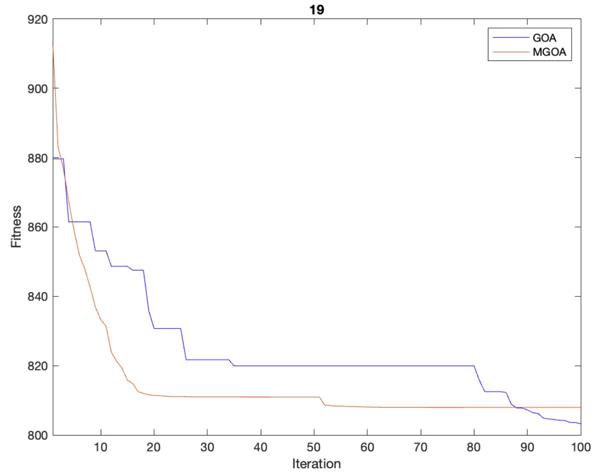
CEC2014-F5



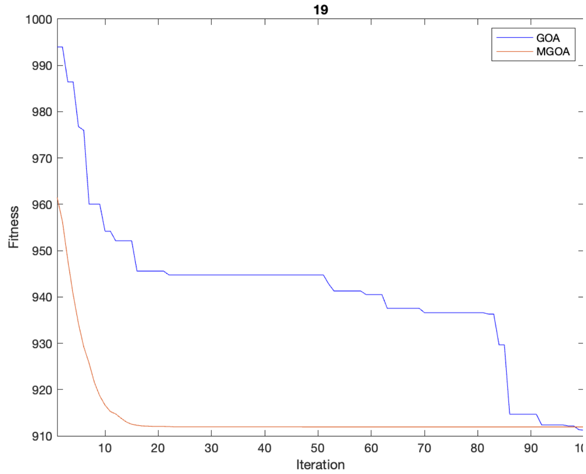
CEC2014-F6



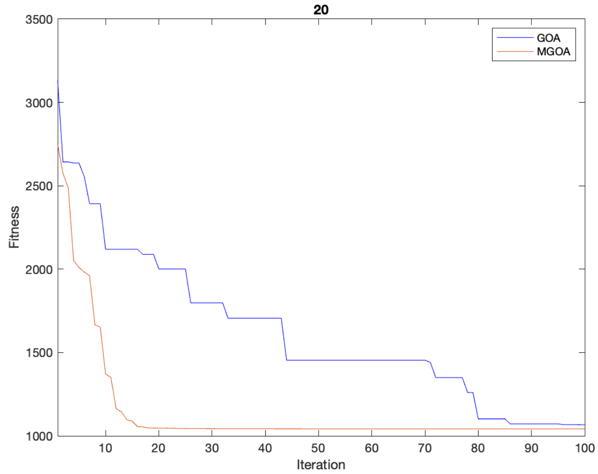
CEC2014-F7



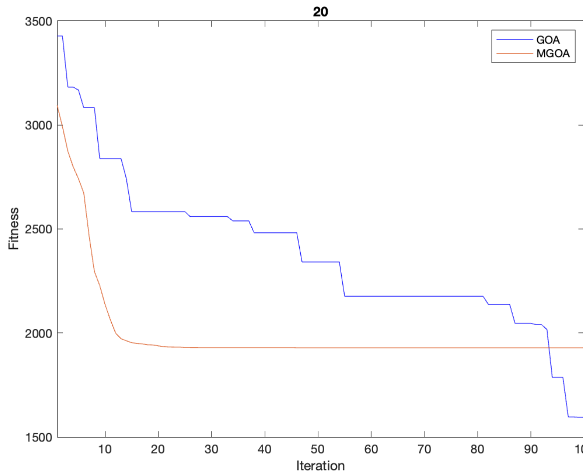
CEC2014-F8



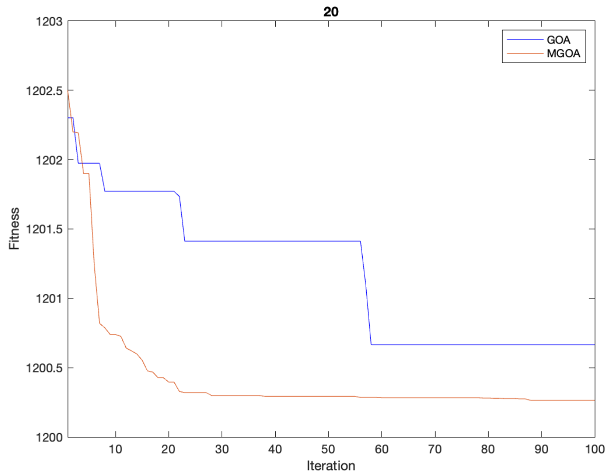
CEC2014-F9



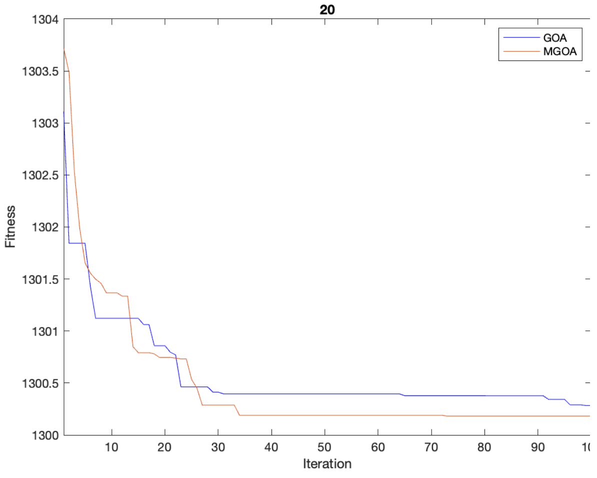
CEC2014-F10



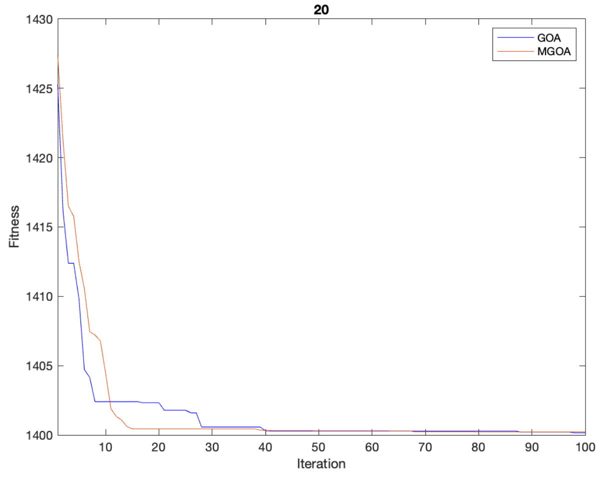
CEC2014-F11



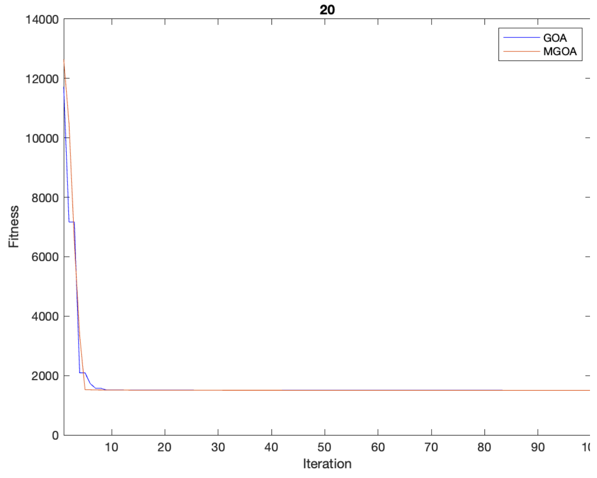
CEC2014-F12



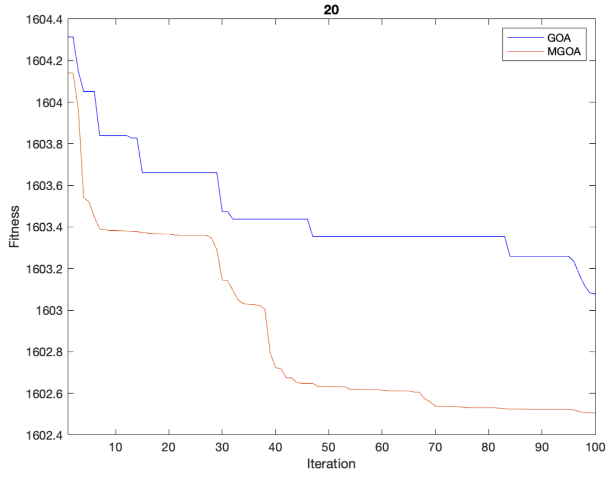
CEC2014-F13



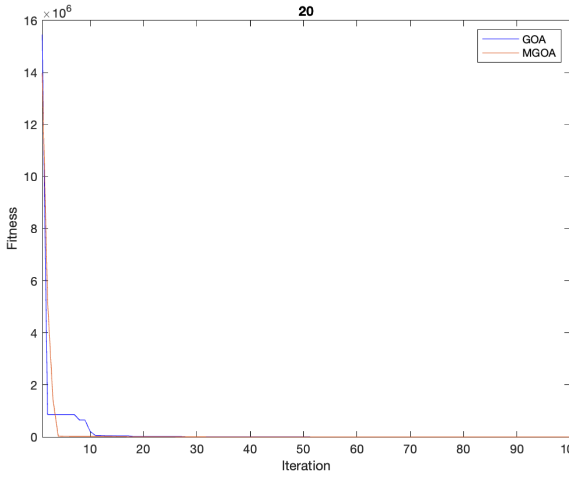
CEC2014-F14



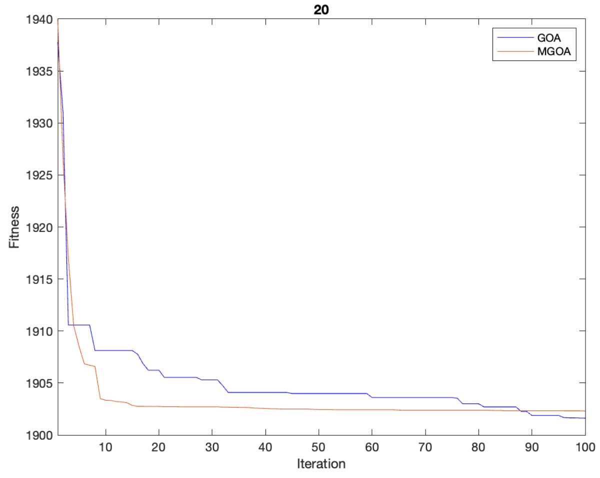
CEC2014-F15



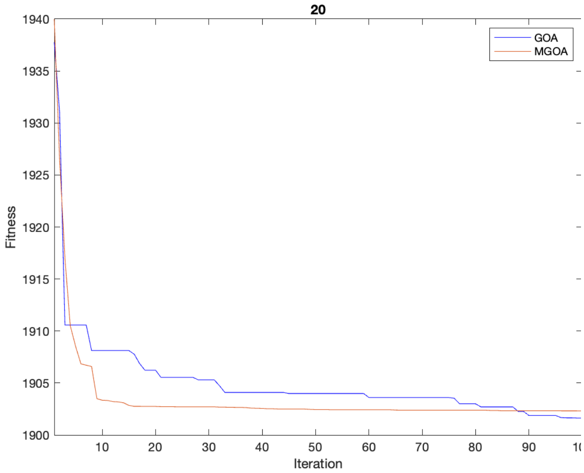
CEC2014-F16



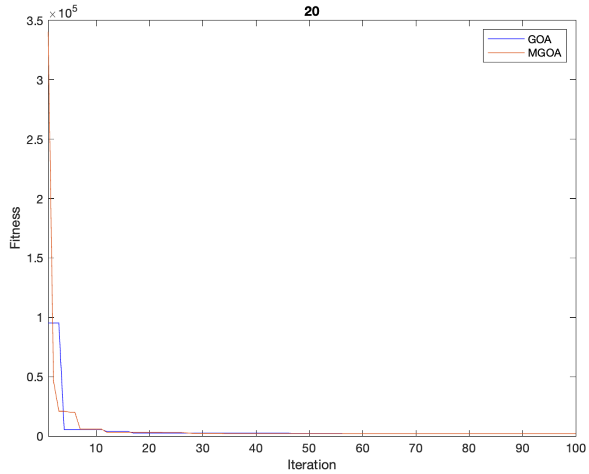
CEC2014-F17



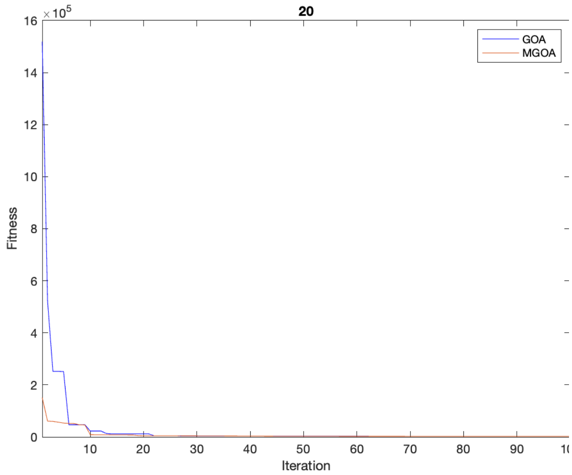
CEC2014-F18



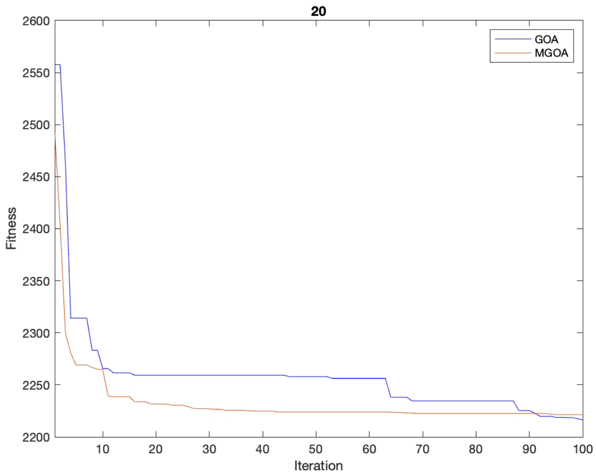
CEC2014-F19



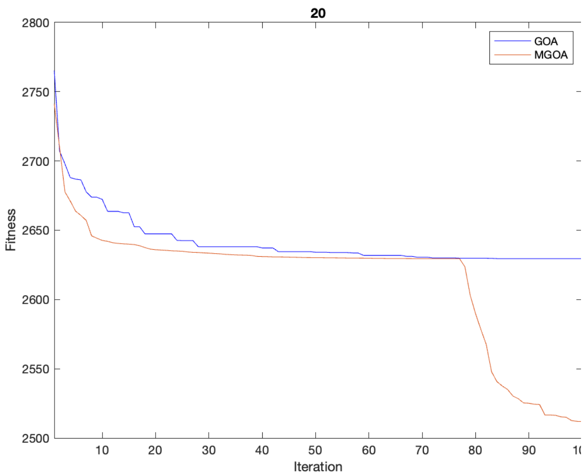
CEC2014-F20



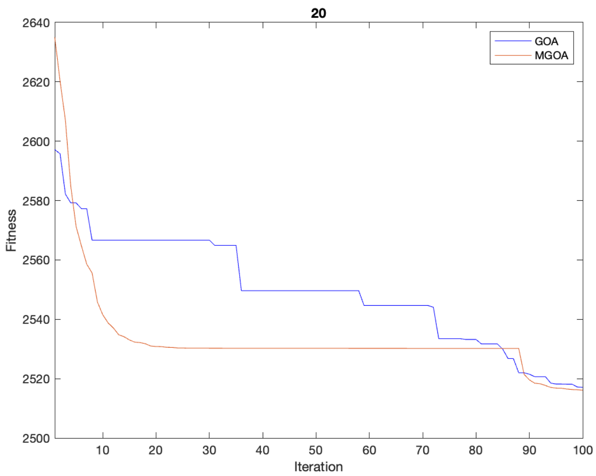
CEC2014-F21



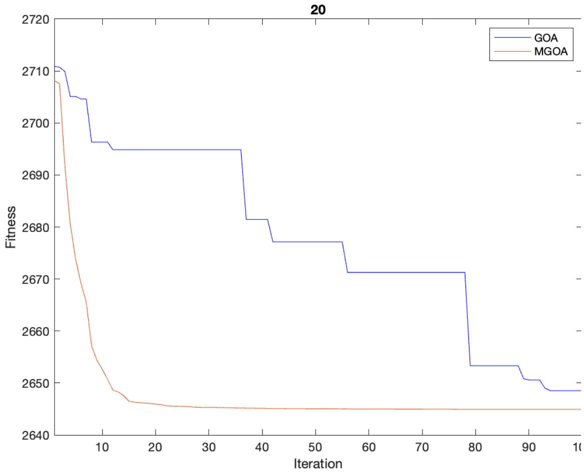
CEC2014-F22



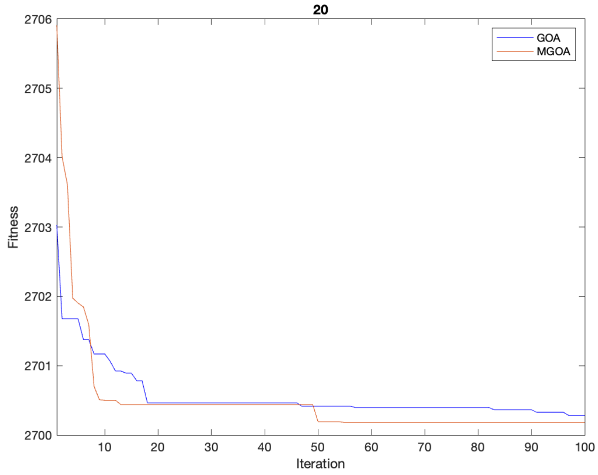
CEC2014-F23



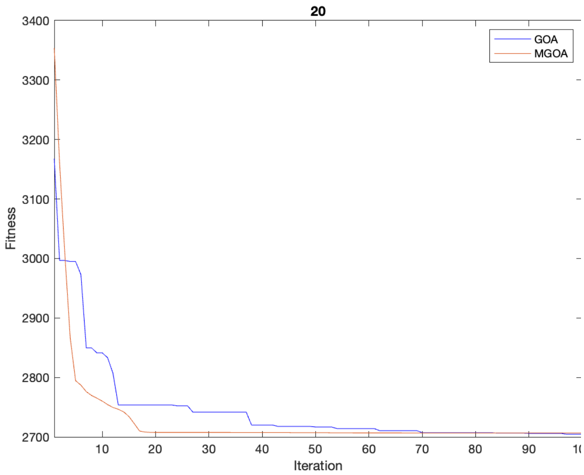
CEC2014-F24



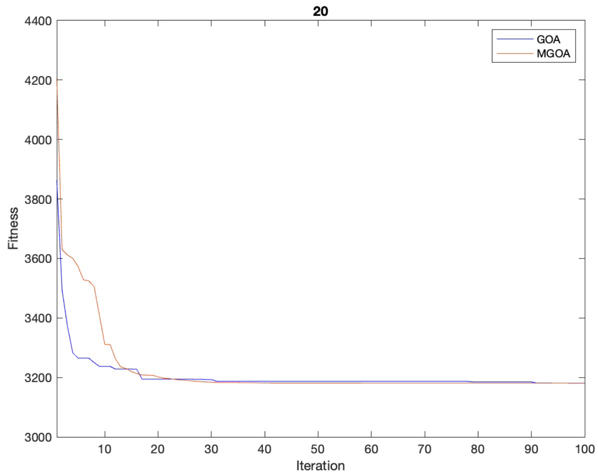
CEC2014-F25



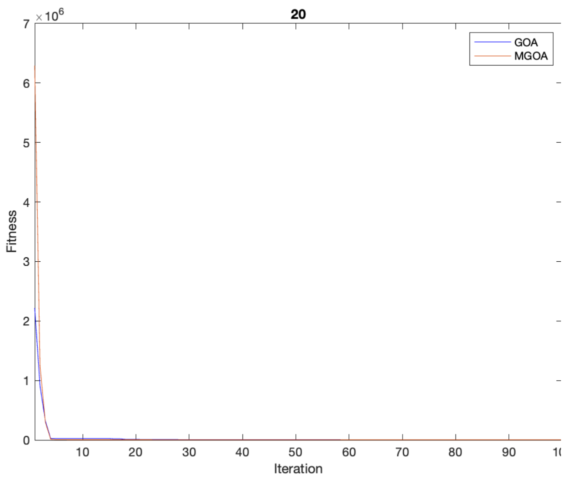
CEC2014-F26



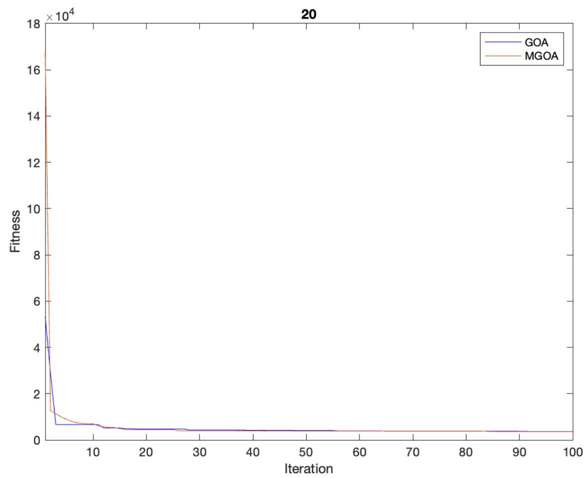
CEC2014-F27



CEC2014-F28

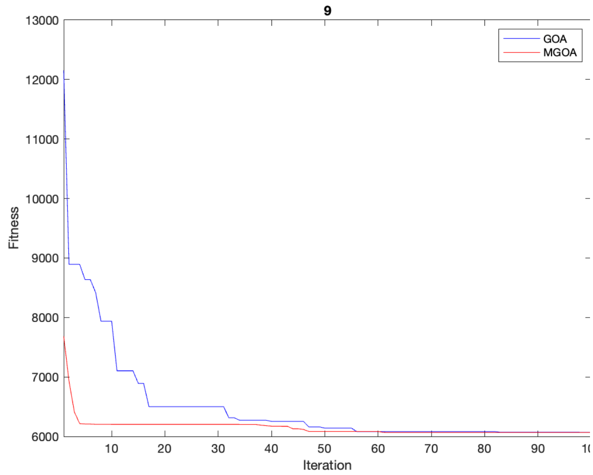


CEC2014-F29

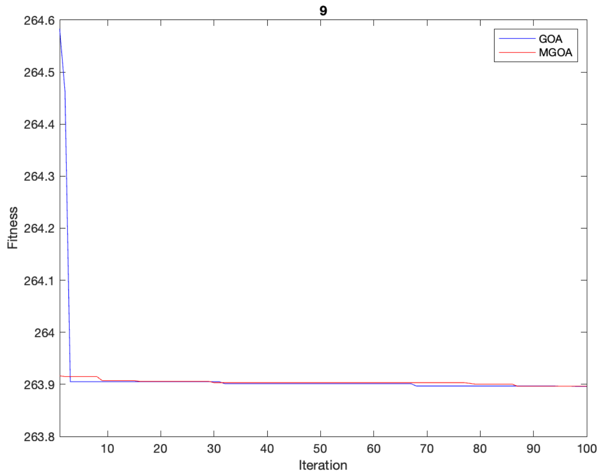


CEC2014-F30

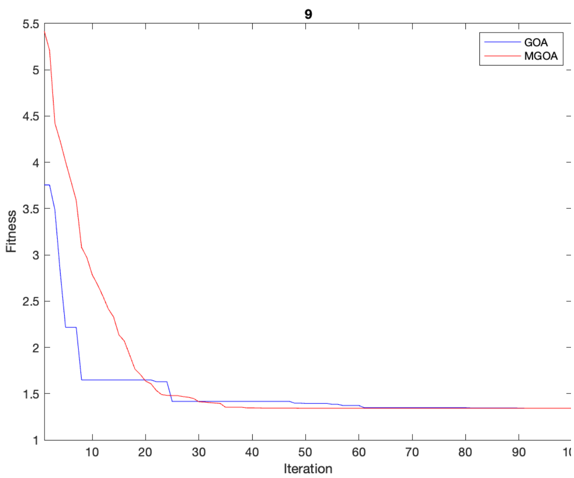
	GOA(ON 50 RUNS)			IGOA(ON 50 RUNS)		
	MEAN	VAR	BEST	MEAN	VAR	BEST
F3	6065.2	18.371	6059.7	6064.8	14.025	6059.7
F4	263.9	2.4997E-07	263.9	263.9	1.2158E-05	263.9
F6	1.3409	3.9831E-07	1.3401	1.341	8.7799E-07	1.34
F8	26.487	2.8514E-07	26.487	26.488	1.0512E-06	26.486
F13	359.21	9.1389E-10	359.21	359.21	4.4943E-10	359.21



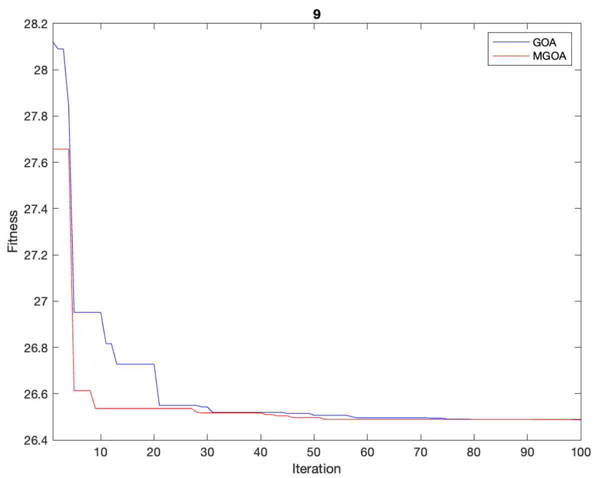
F3



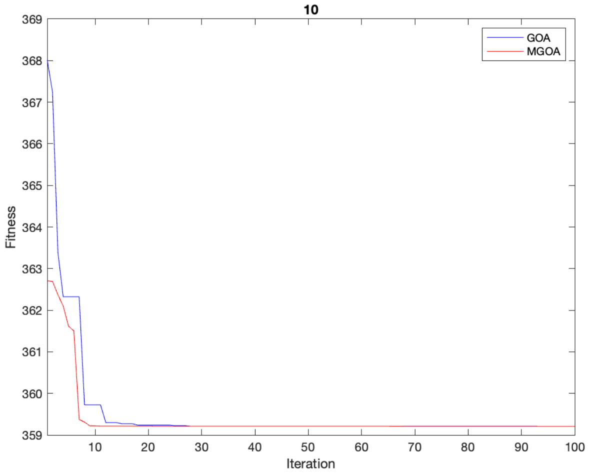
F4



F6



F8



F13

5.3. Performance Comparison with 6 state-of-the-art algorithms

5.3.1. Comparison on CEC2017 AND CEC2014

CEC 2017 (MEAN ON 50 RUNS)

	GOA	IGOA	P VALUE	SCSO	AOA	FHO	WOA	GWO	COA
F1	6105200	3961.3	7.0661E-18	11554000	3667.7	434730000	3249900	14546000	659270
F2	0	0		0	0	0	0	0	0
F3	518.46	304.27	8.9852E-18	897.22	300.8	4794.2	646.69	785.05	3870.9
F4	406.55	405.52	7.2663E-05	432.65	407.64	478.79	444.33	433.9	427.16
F5	513.67	516.86	0.0037453	532.42	520.34	541.54	540.25	510.87	520.75
F6	601.22	603.79	2.6758E-12	610.11	600.59	619.85	627.56	600.87	603.55
F7	733.68	735.5	0.26556	753.35	731.46	781.06	777.8	730.37	749.15
F8	813.98	817.43	0.0064009	834.5	822.35	836.4	842.3	812.11	824.22
F9	903.34	945.8	1.733E-17	1140.2	923.6	1087	1640.5	906.08	1141.5
F10	1479.4	1434.7	0.19615	1822.9	1515.6	2180	1944.6	1512.6	2313.5
F11	1109.1	1107.9	0.037035	1130	1115.7	1216.7	1147.2	1112.7	1127.3
F12	104310	29835	1.7569E-11	2915900	12859	6513100	3019500	2208900	608410
F13	1696.2	1494.5	7.2511E-14	14533	2889.9	31082	14157	13736	8060.9
F14	1448.8	1436.8	7.12E-11	1805.9	1703.4	1609.2	1606.1	1961.1	2323.9
F15	1555.5	1524.8	2.706E-15	1777.3	1602	1970.8	5249.1	2166.8	2506.5
F16	1608.3	1610.5	0.69691	1713.1	1691.8	1697.6	1753.3	1655	1705.5
F17	1739.7	1736.2	0.028115	1761.1	1746.6	1778.2	1803.6	1746.2	1761.2
F18	2396.1	2000.3	4.6997E-11	12067	6735.4	29862	15132	11711	12876
F19	1915.3	1909.8	6.4943E-11	2351.8	1969.2	2843.1	3144	2274.7	2887.3
F20	2038.9	2042.5	0.019983	2089.7	2054.5	2092.7	2153	2044	2074.7
F21	0	0		0	0	0	0	0	0
F22	0	0		0	0	0	0	0	0
F23	2666.8	2672.1	0.4841	2500	2653.1	2538.3	2708.6	2662.2	2500
F24	2560.1	2513.6	5.862E-10	2600	2602.8	2612.1	2758.3	2781.5	2600
F25	2900.1	2898.8	0.028612	2700	2929.3	2974.8	2955.1	2938.4	2700
F26	2782.8	2780.7	0.00011474	2800	3052.9	3039.4	3126.6	3063	2800
F27	3134.8	3133.1	0.28683	2900	3185.5	3134.1	3231.1	3153.5	2900
F28	3121.6	3110.4	3.0771E-07	3000	3155.5	3207.8	3222.8	3180.9	3000
F29	3165	3160	0.24539	3100	3189.3	3202.2	3246.3	3164.5	3100
F30	9870.1	4699.2	5.2389E-15	4759.1	6290.2	15657	684650	12036	3200

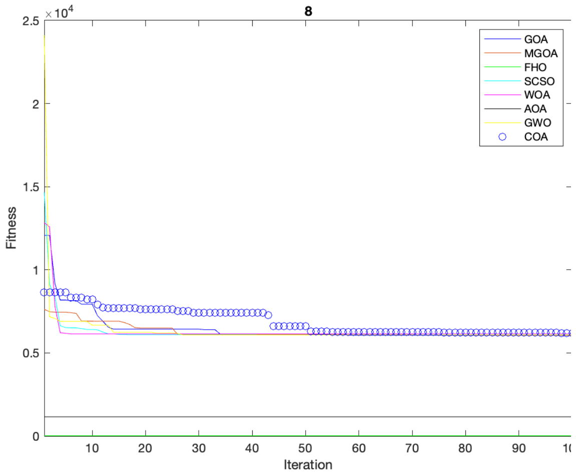
CEC 2014 (MEAN ON 50 RUNS)

	GOA	IGOA	P VALUE	SCSO	AOA	FHO	WOA	GWO	COA
F1	300880.2763	66023.80933	1.60E-16	4010100	104180	4650000	8339600	2771200	1758100
F2	3994809.729	6061.320114	7.07E-18	141960	3829.2	850010000	4580400	175790	150490
F3	749.4	448.96	5.84E-12	2974	595.62	6568.2	53319	6498	10242
F4	408.6	403.56	1.80E-09	429.05	407.79	480.14	445.32	433.3	427.71
F5	519.66	519.34	2.71E-15	520.12	520.04	520.39	520.16	520.45	520.31
F6	601.6389591	603.0187046	2.13E-09	604.85	601.87	606.84	607.52	601.65	604.18
F7	701.01	700.51	1.58E-13	700.65	700.2	706.65	701.12	701.13	700.36
F8	805.95	806.73	2.81E-01	829.05	812.8	835.47	836.86	809.8	811.85
F9	914.0825507	916.5901018	3.02E-02	931.06	922.78	939.5	940.4	911.66	934.94
F10	1098.042925	1170.50471	4.76E-03	1662.1	1217.9	2008.7	1635.6	1367.3	1517.4
F11	1602.8	1589.8	7.49E-01	2057	1891.1	2329.2	2117.7	1553.6	1929.8
F12	1200.6	1200.3	8.94E-14	1200.4	1200.5	1201.2	1200.8	1201.2	1200.6
F13	1300.228171	1300.20402	1.03E-01	1300.3	1300.2	1300.8	1300.4	1300.2	1300.3
F14	1400.17366	1400.154276	1.89E-02	1400.4	1400.3	1400.6	1400.3	1400.2	1400.4
F15	1502.375395	1501.937685	5.05E-04	1503.1	1501.4	1511.4	1506.8	1501.9	1502.5
F16	1602.7	1602.8	4.38E-02	1603	1603	1603.5	1603.4	1602.6	1603
F17	2571.771204	2262.940877	1.84E-11	6738.8	4100.9	11150	217630	24648	32564
F18	1885.5	1873.4	1.06E-02	12785	10181	10698	15087	10322	9900.6
F19	1902.1	1902	3.95E-01	1903.5	1901.9	1907.3	1905.8	1902.5	1902.9
F20	2054.044006	2031.389122	3.02E-10	5252.7	2424.6	3034.2	6718.4	5527.1	5995
F21	2461	2294.8	7.12E-11	8503.1	2488.5	7249.2	106580	8552.8	6985.5
F22	2223.9	2220.5	1.58E-07	2275.2	2271.1	2251	2295.7	2252.5	2251
F23	2577.0601	2549.164879	2.99E-05	2500	2629.5	2648.9	2635.8	2631.1	2500
F24	2519.6	2522.8	8.71E-03	2564.5	2526	2551.6	2557.3	2518.9	2578.3
F25	2641.767658	2636.949789	6.96E-03	2697.4	2651.5	2667.4	2696.3	2697.2	2698.5
F26	2700.224484	2700.18401	1.44E-04	2700.3	2700.2	2700.5	2700.4	2700.2	2700.3
F27	2704.418578	2706.411162	1.34E-09	2845.8	2815.8	2756.7	3024	2969.4	2869.1
F28	3179.4	3176.8	9.83E-03	3000	3175.8	3173.8	3313.7	3226.4	3000
F29	3310.464308	3215.387373	1.48E-10	4349.8	3349.2	45250	4573.3	38388	44642
F30	3630.203666	3595.495106	4.83E-02	4638.3	3790.3	4205	5203.1	3961.5	4263.4

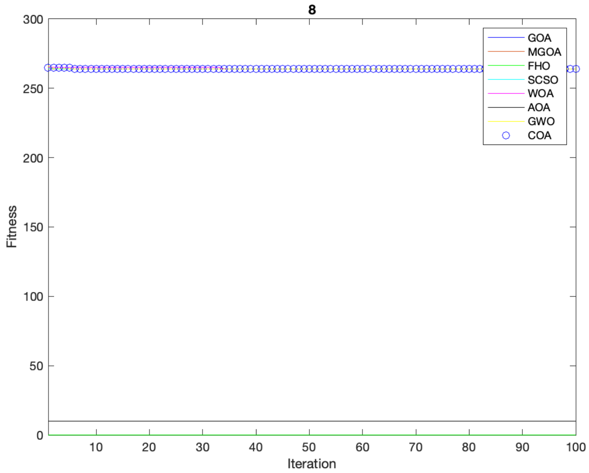
5.3.2. Comparison on Engineering Problems

	GLOBAL MIN	GOA	IGOA	SCSO	FHO	AOA	COA	GWO	WOA
F3	6059.71434	6065.2	6065.9	6576.6	15.902	1032.6	6193	6100.8	6895.8
F4	263.895843	263.9	263.9	263.9	0	12.356	263.9	263.9	263.94
F6	1.3399576	1.3409	1.341	1.3401	0.00312	1.3401	1.3428	1.3401	1.4712
F8	26.4863615	26.487	26.488	26.487	7.92	9.8142	26.486	26.494	26.79
F13	359.208	359.21	359.21	359.82	260.4	271.48	359.27	359.27	361.46

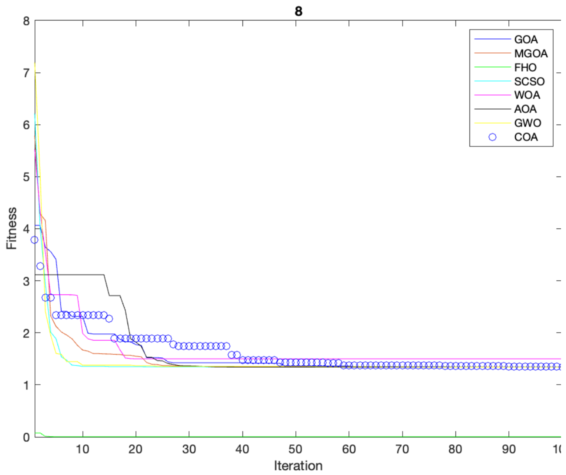
Mean Results computed on 50 runs



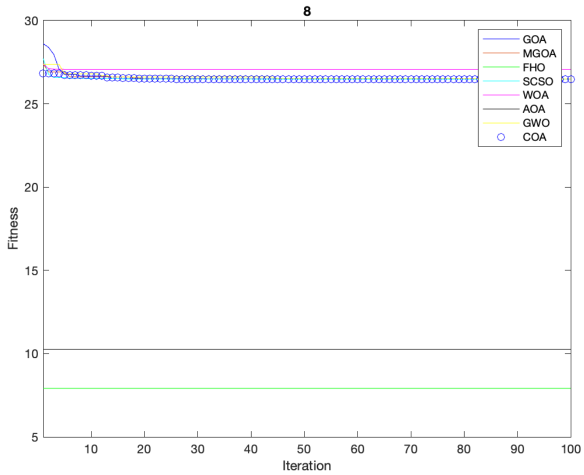
F3



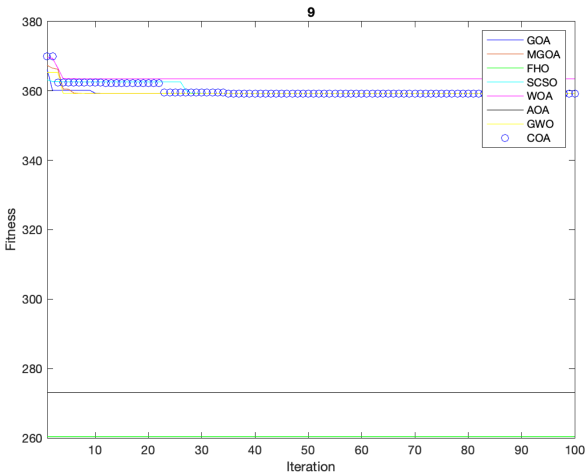
F4



F6

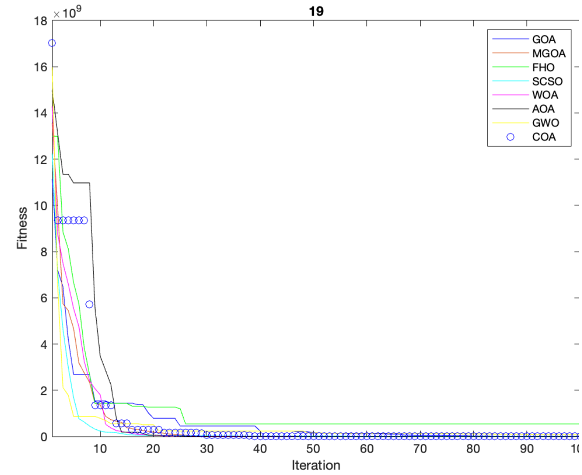


F8

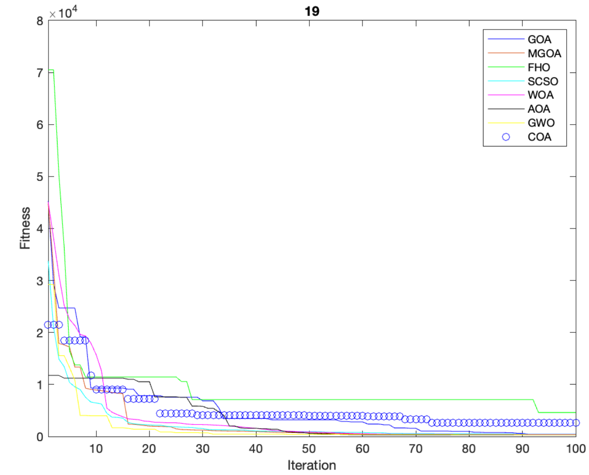


F13

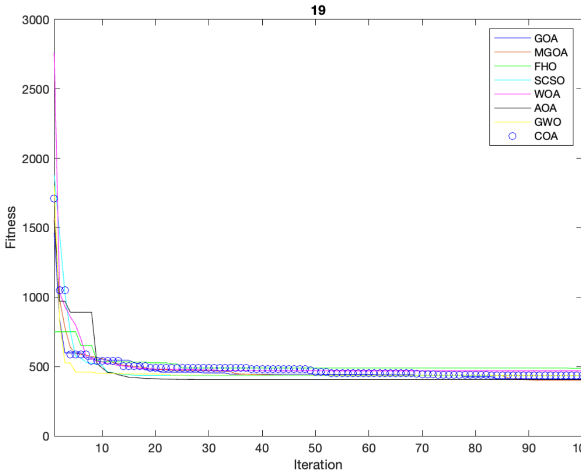
5.3.3. Convergence Curves for CEC2017



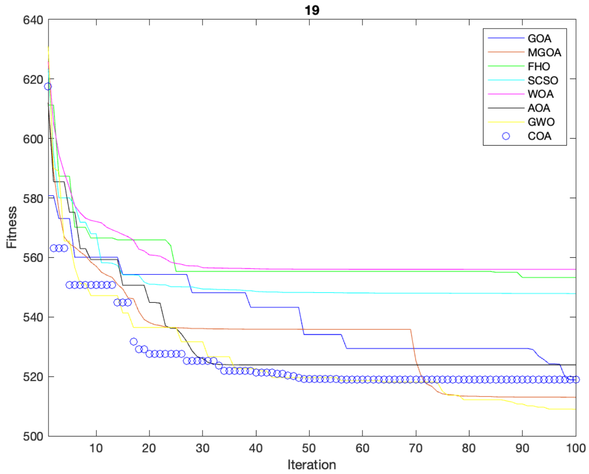
CEC2017-F1



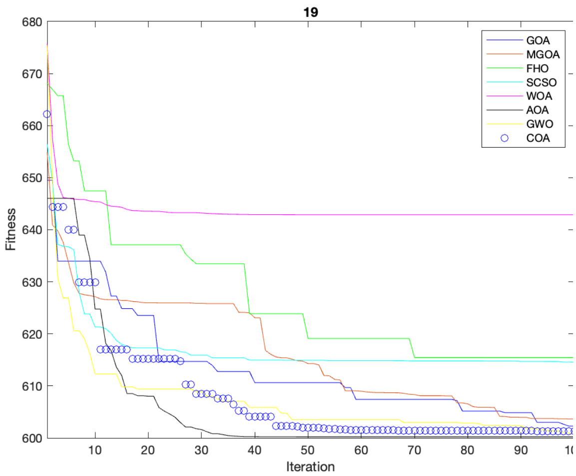
CEC2017-F3



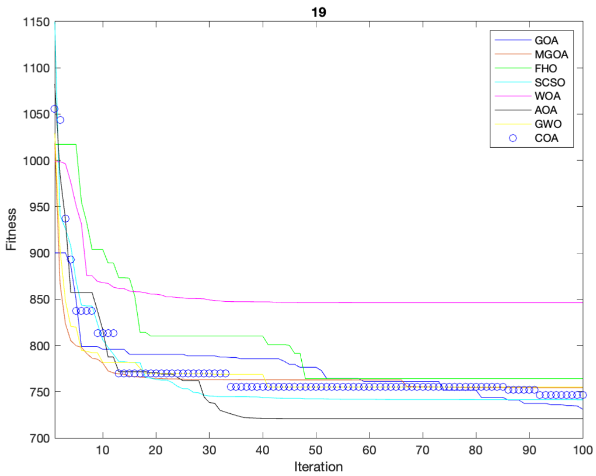
CEC2017-F4



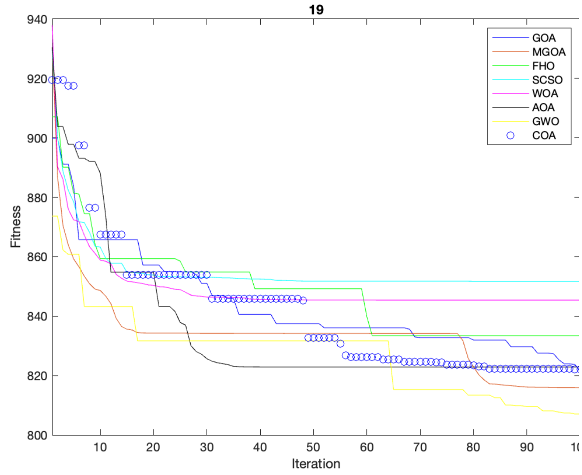
CEC2017-F5



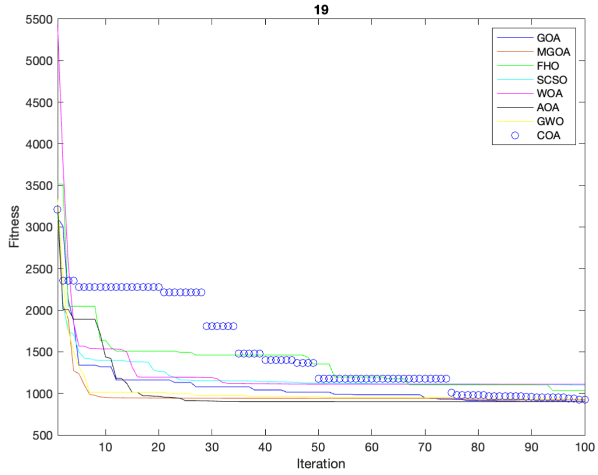
CEC2017-F6



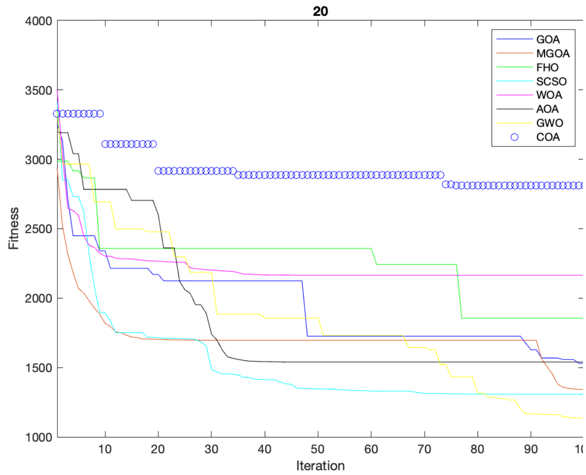
CEC2017-F7



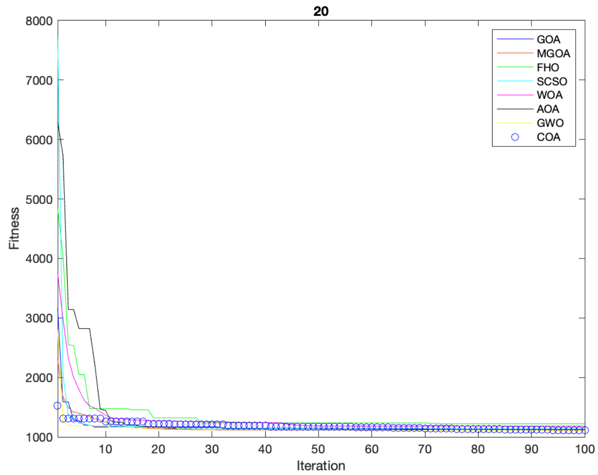
CEC2017-F8



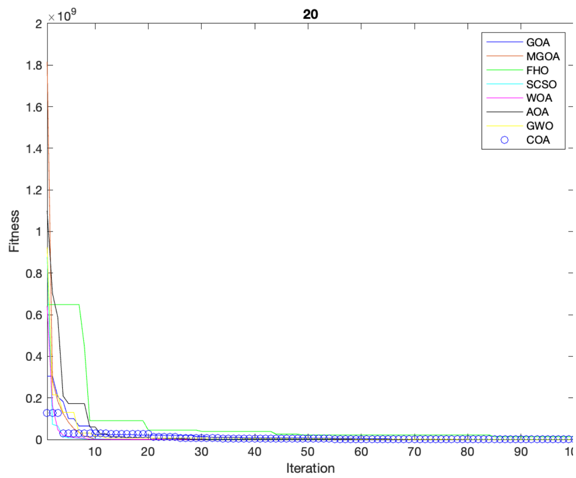
CEC2017-F9



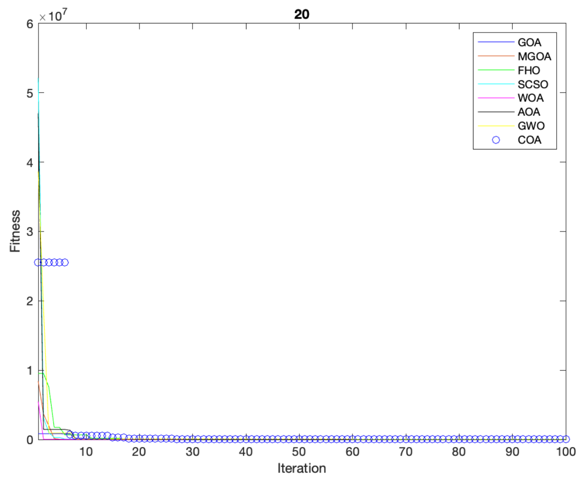
CEC2017-F10



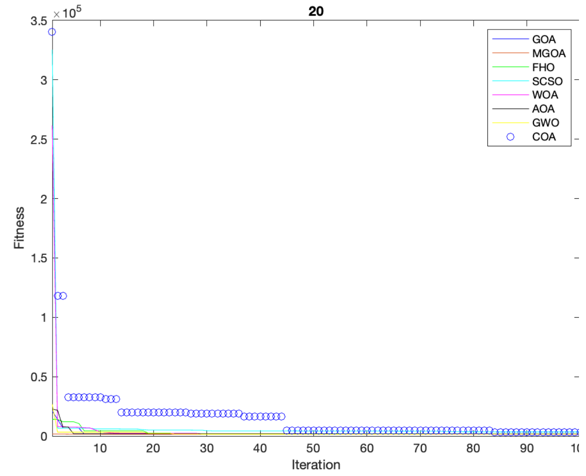
CEC2017-F11



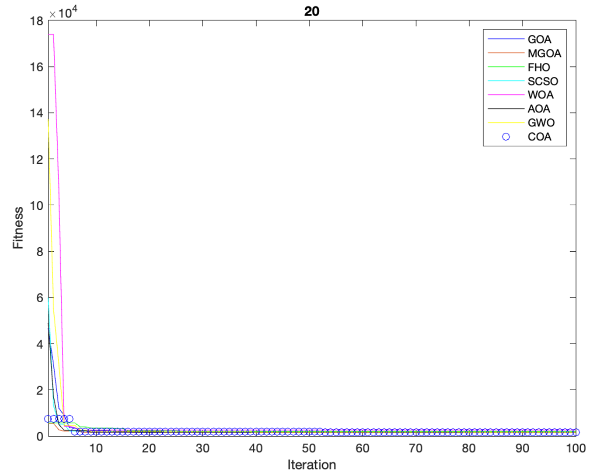
CEC2017-F12



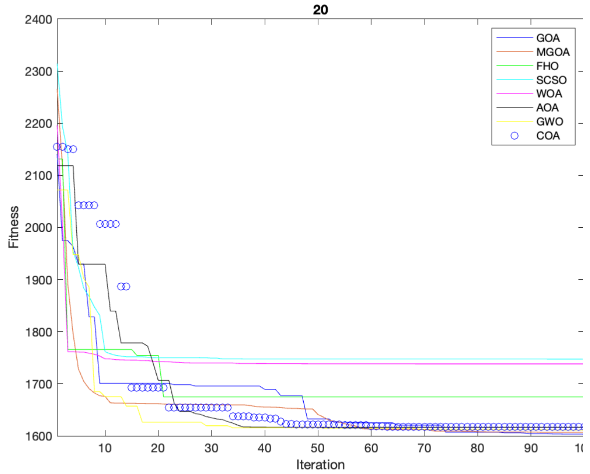
CEC2017-F13



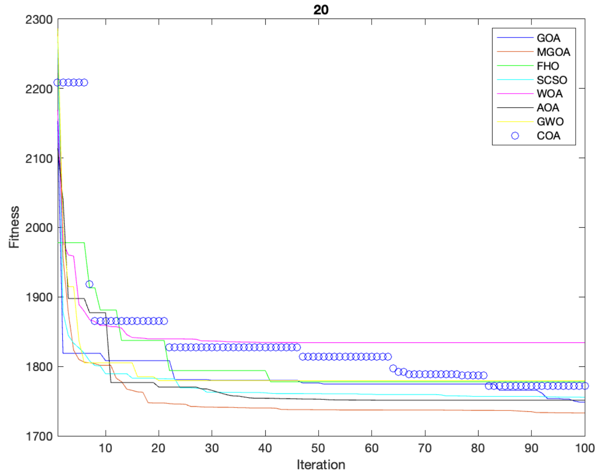
CEC2017-F14



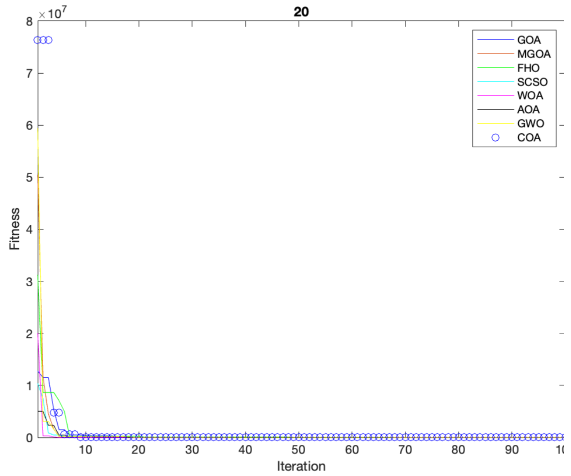
CEC2017-F15



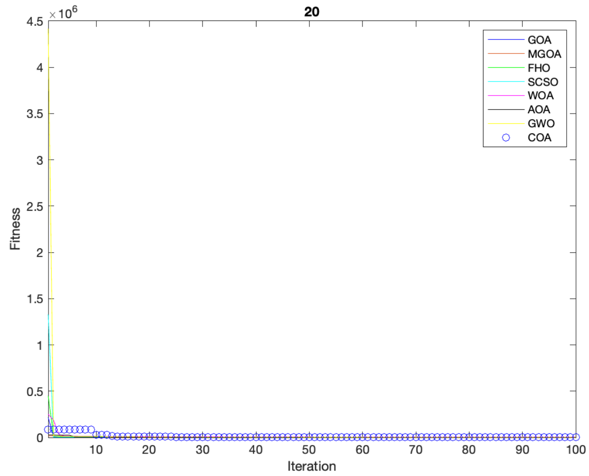
CEC2017-F16



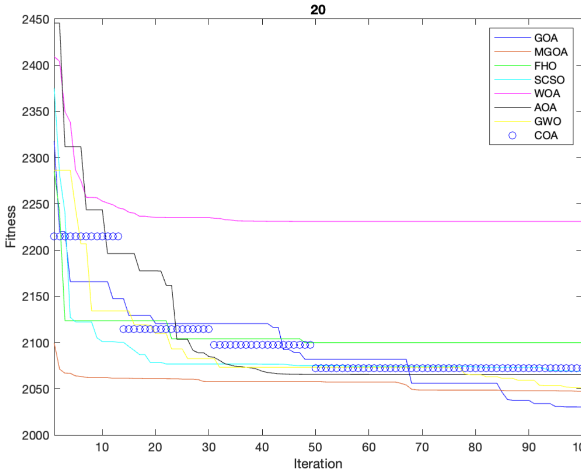
CEC2017-F17



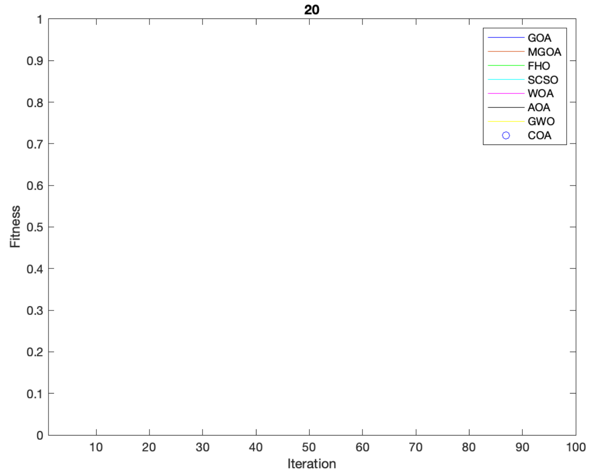
CEC2017-F18



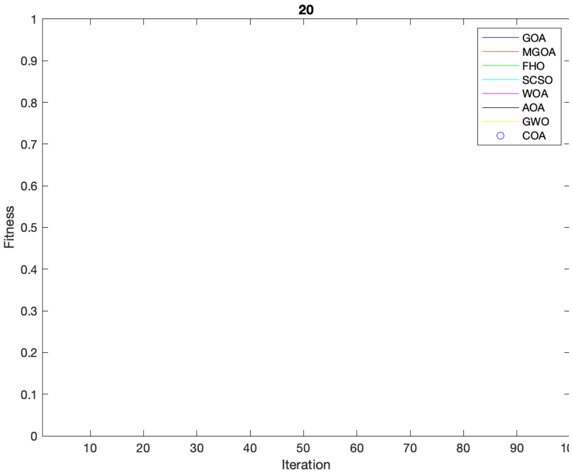
CEC2017-F19



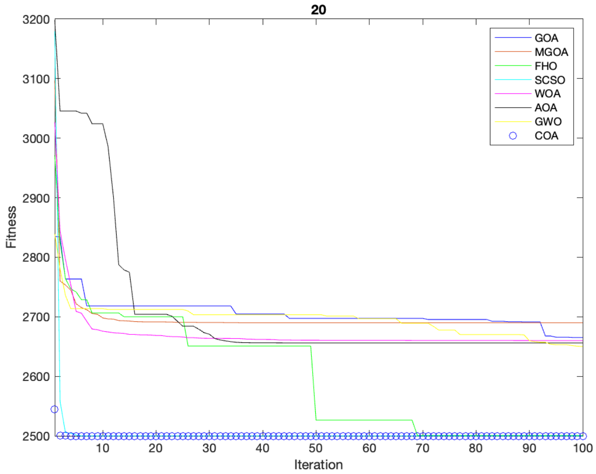
CEC2017-F20



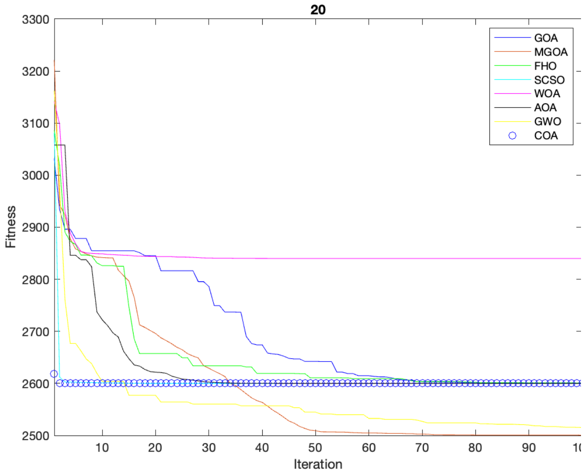
CEC2017-F21



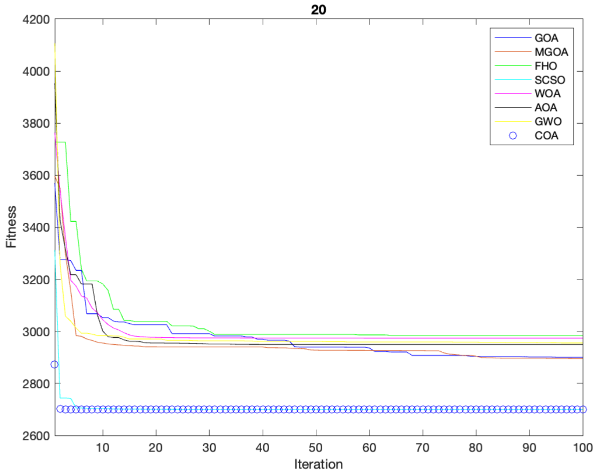
CEC2017-F22



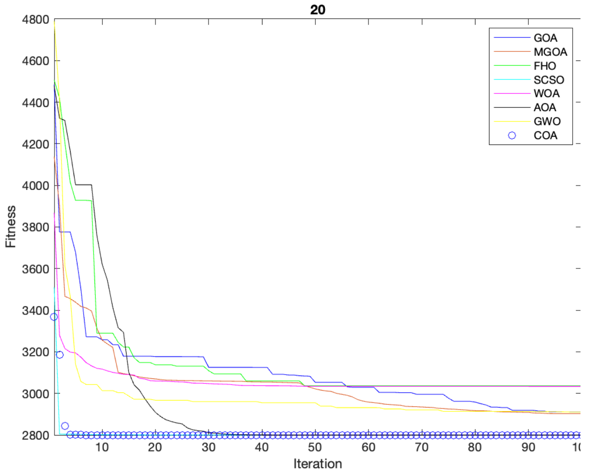
CEC2017-F23



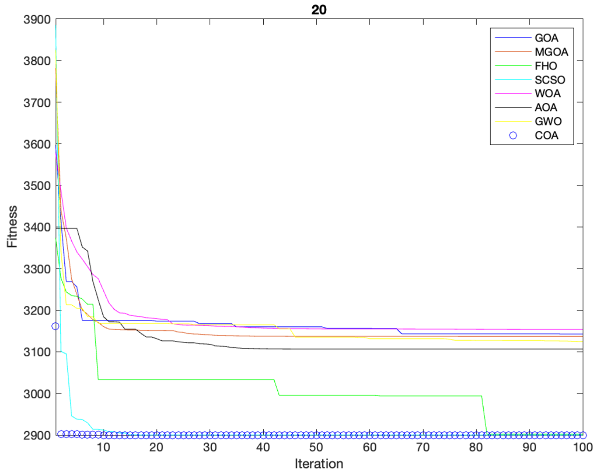
CEC2017-F24



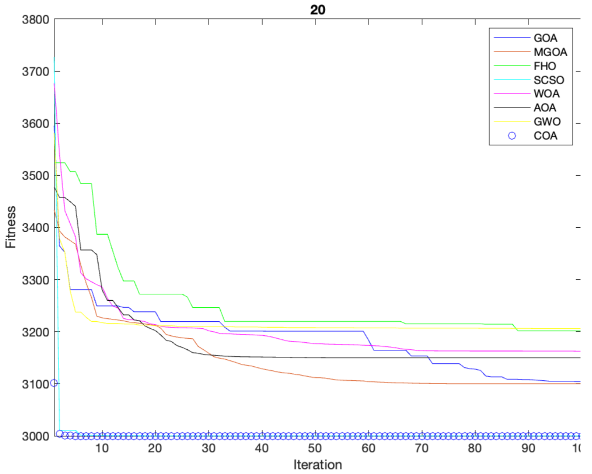
CEC2017-F25



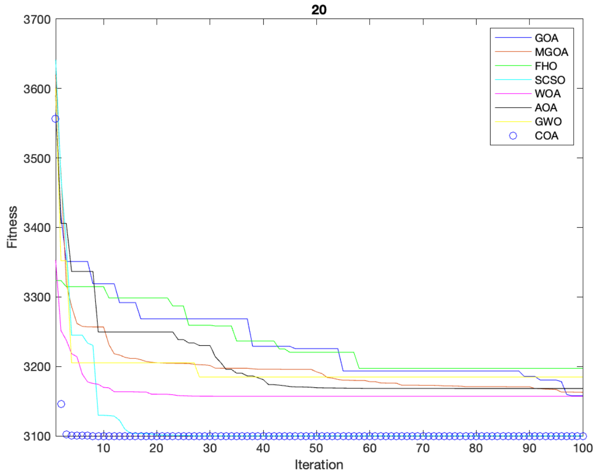
CEC2017-F26



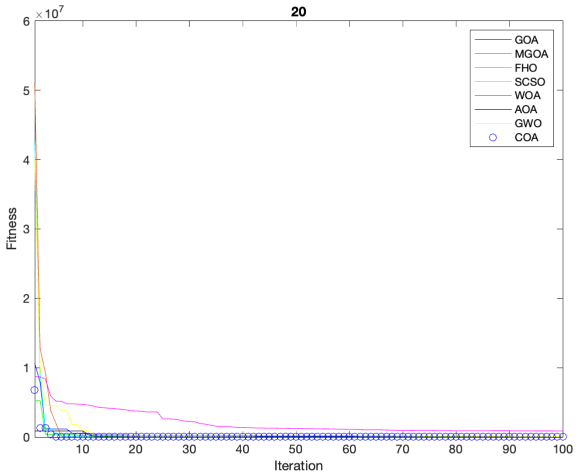
CEC2017-F27



CEC2017-F28

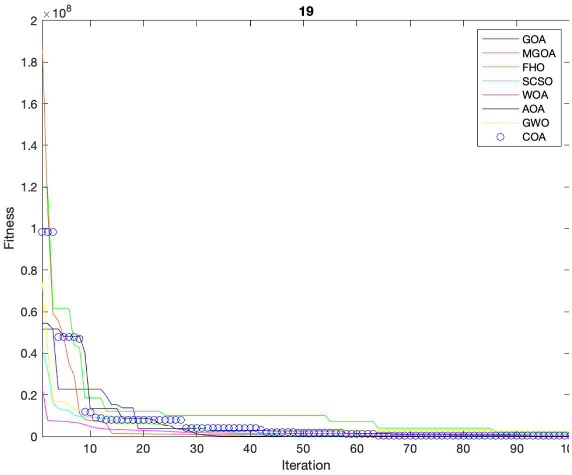


CEC2017-F29

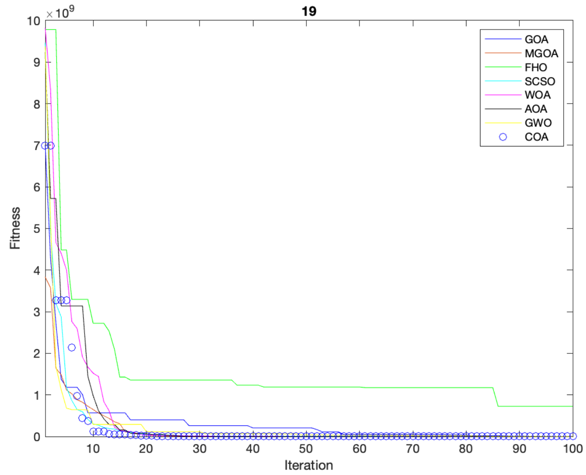


CEC2017-F30

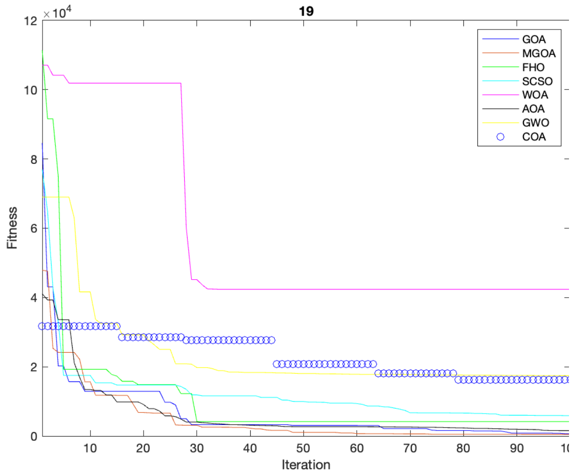
5.3.4. Convergence Curves for CEC2014



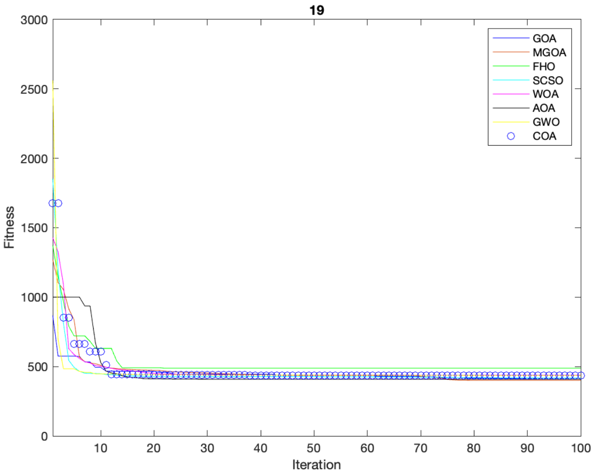
CEC2014-F1



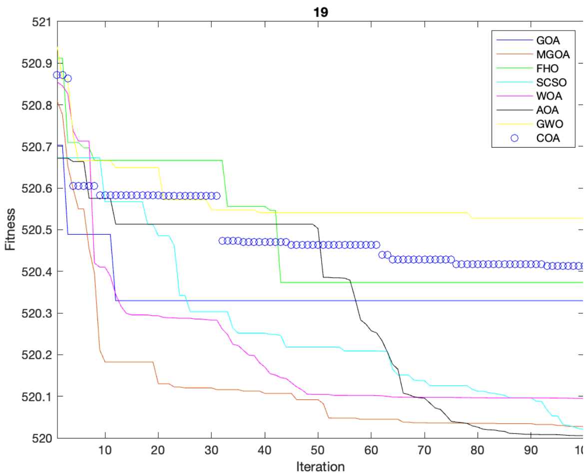
CEC2014-F2



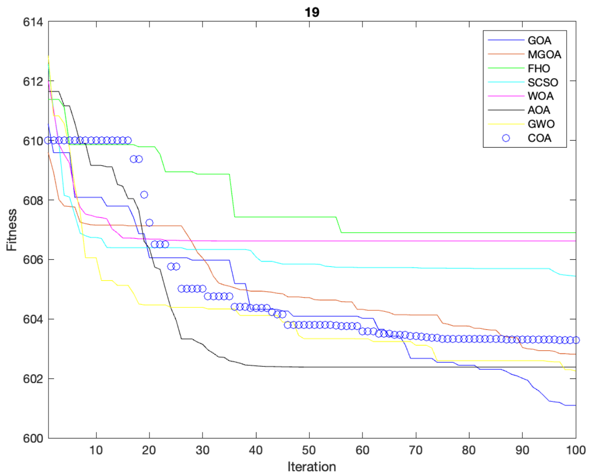
CEC2014-F3



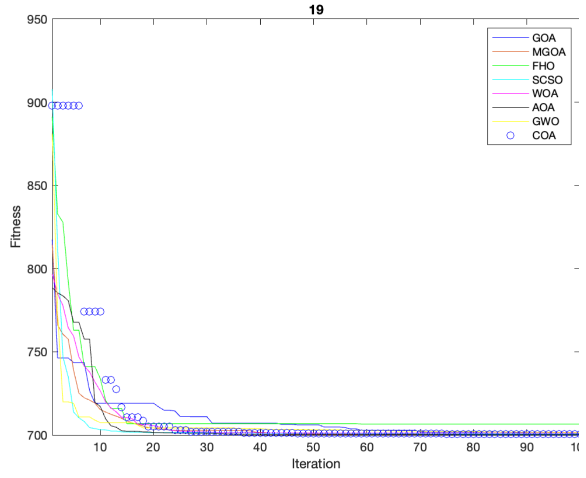
CEC2014-F4



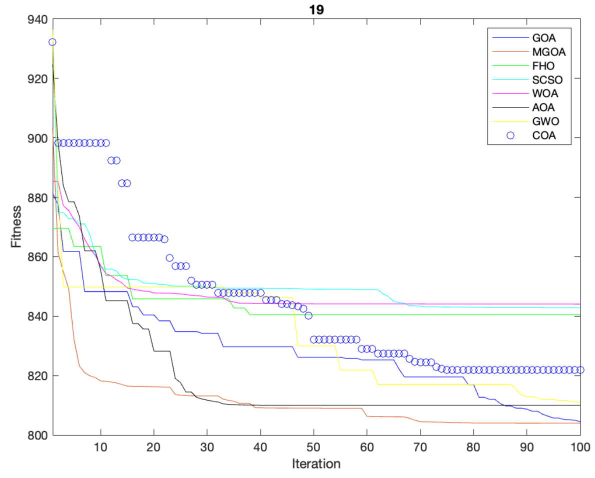
CEC2014-F5



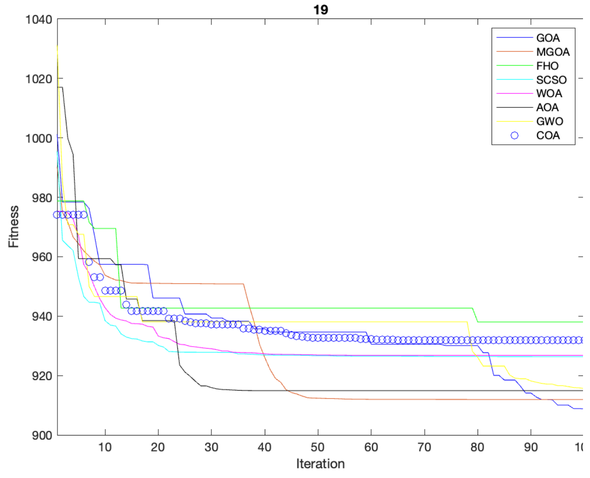
CEC2014-F6



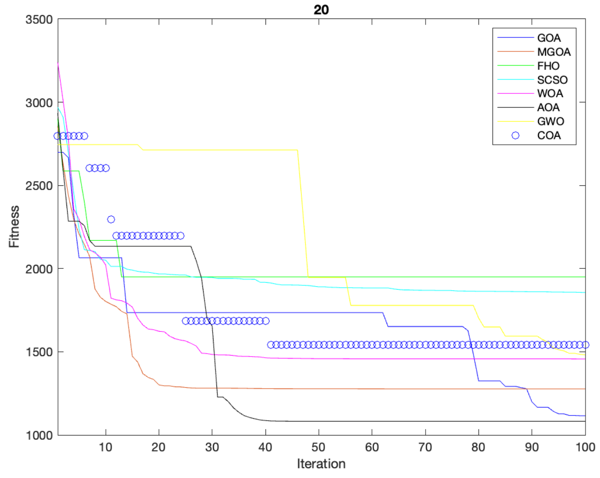
CEC2014-F7



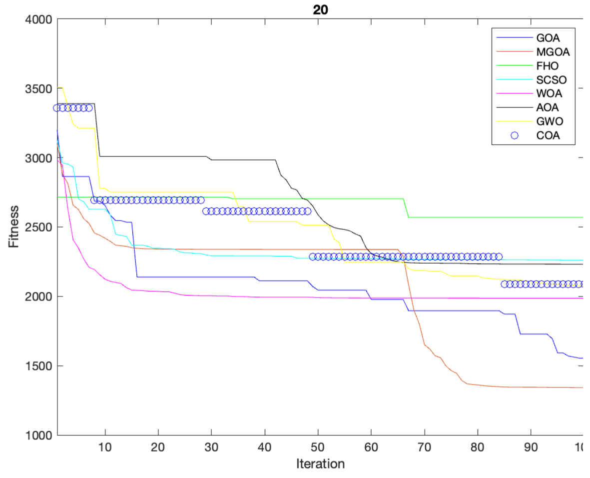
CEC2014-F8



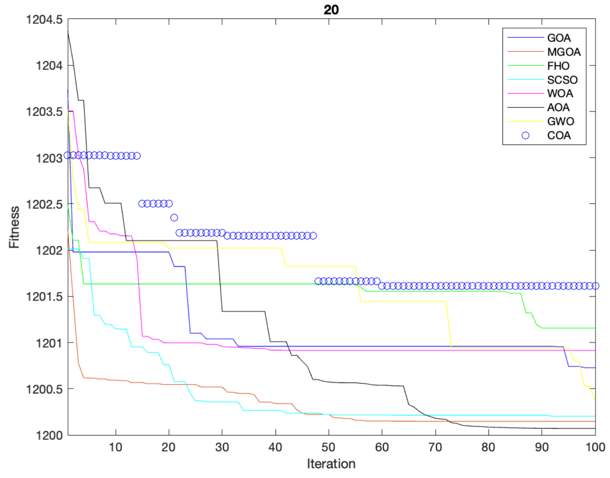
CEC2014-F9



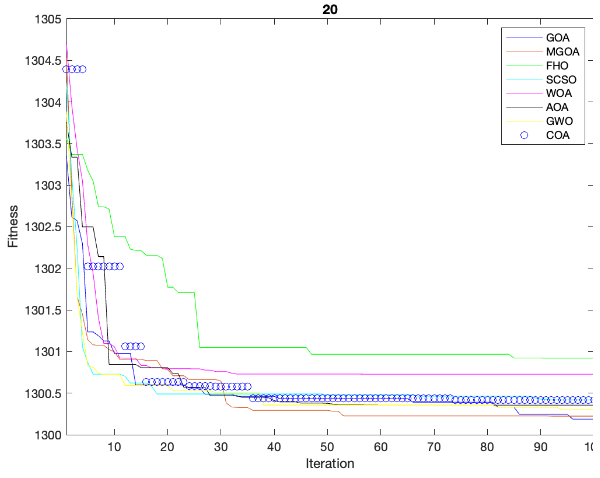
CEC2014-F10



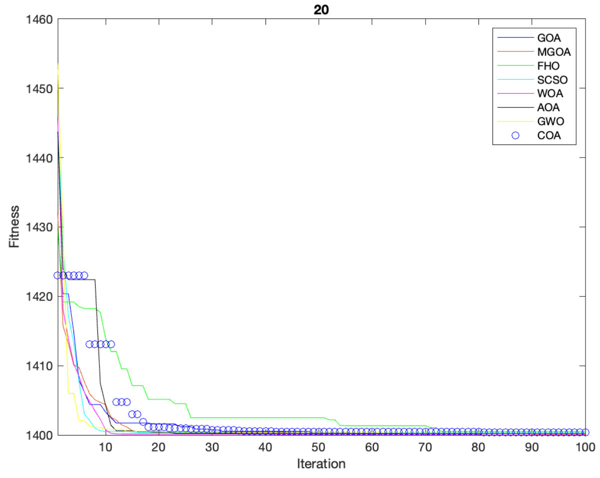
CEC2014-F11



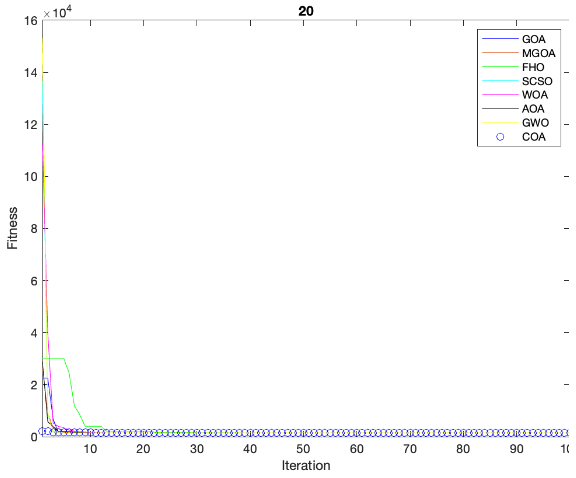
CEC2014-F12



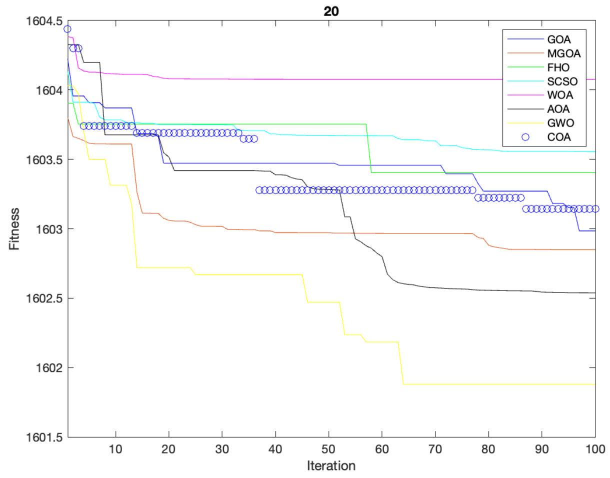
CEC2014-F13



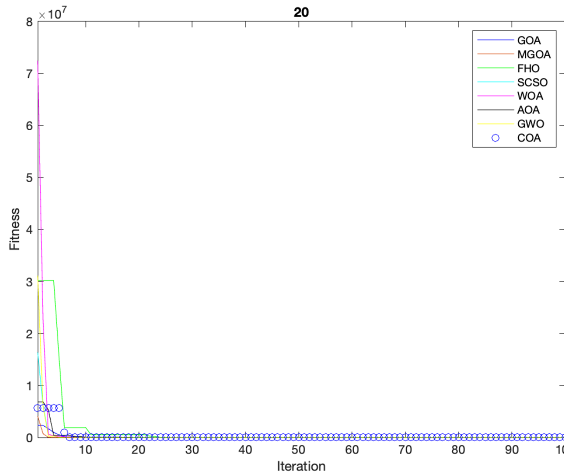
CEC2014-F14



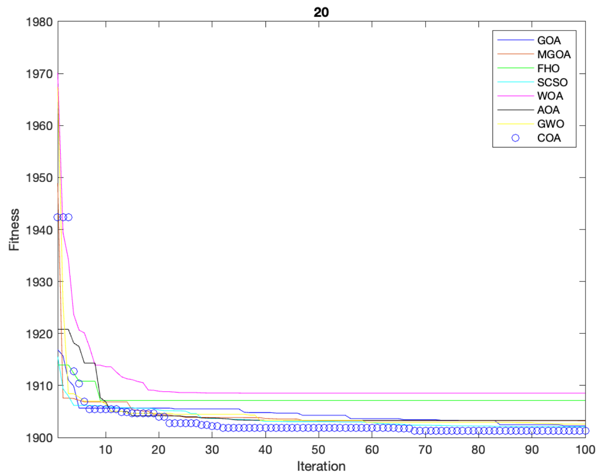
CEC2014-F15



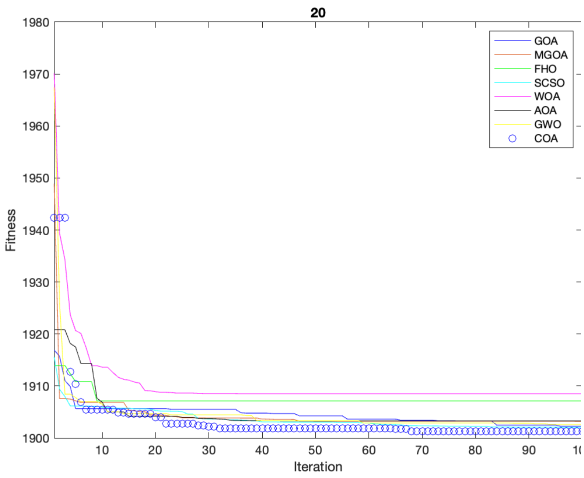
CEC2014-F16



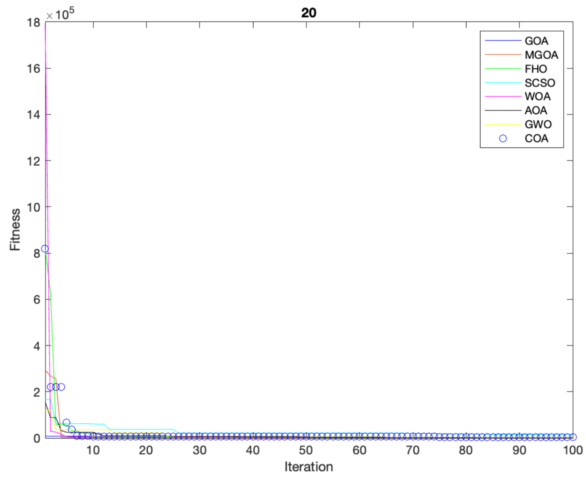
CEC2014-F17



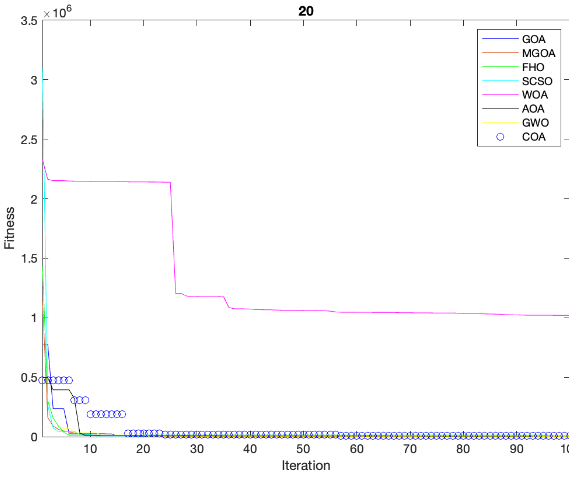
CEC2014-F18



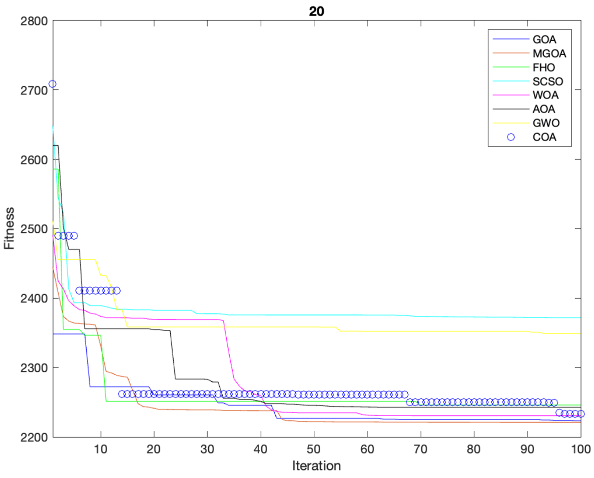
CEC2014-F19



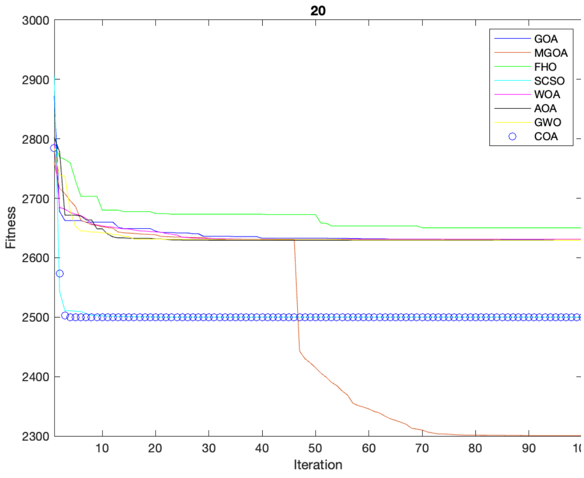
CEC2014-F20



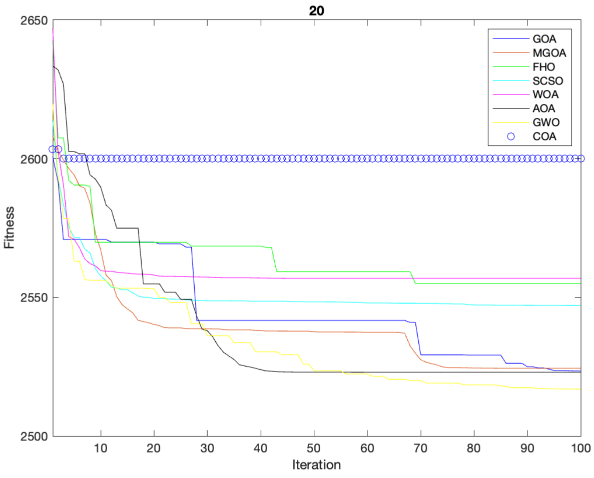
CEC2014-F21



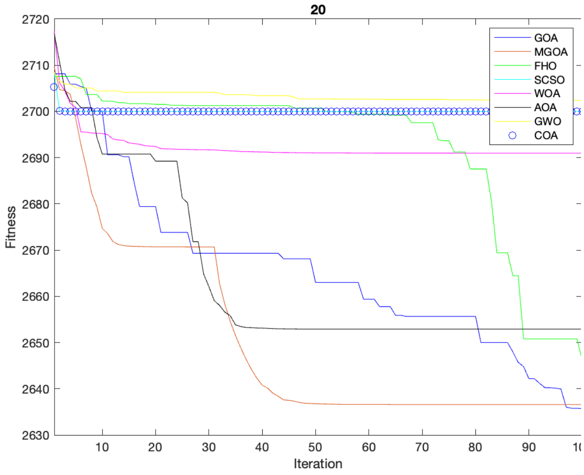
CEC2014-F22



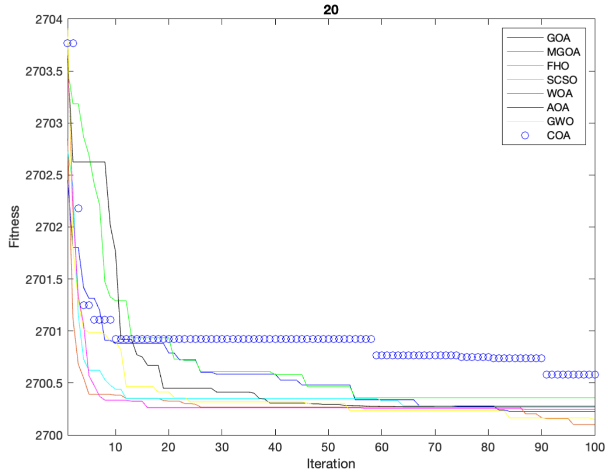
CEC2014-F23



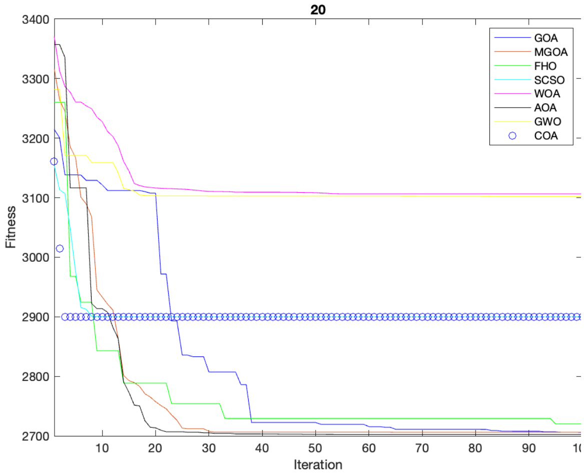
CEC2014-F24



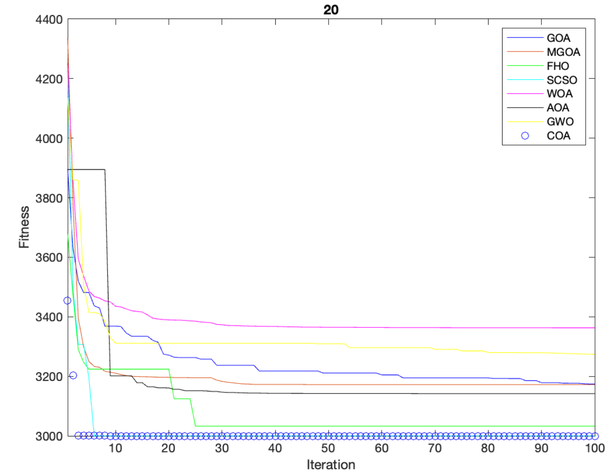
CEC2014-F25



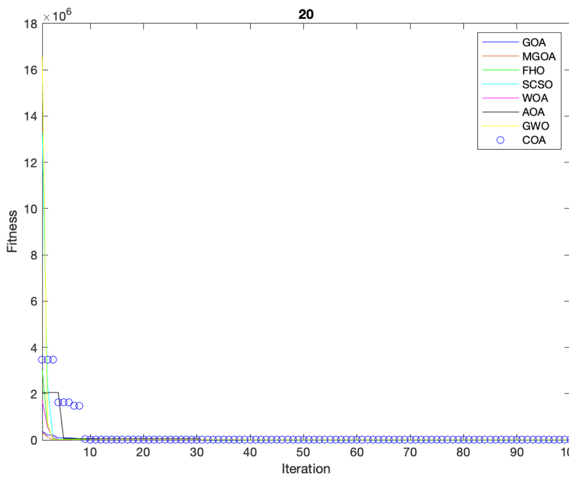
CEC2014-F26



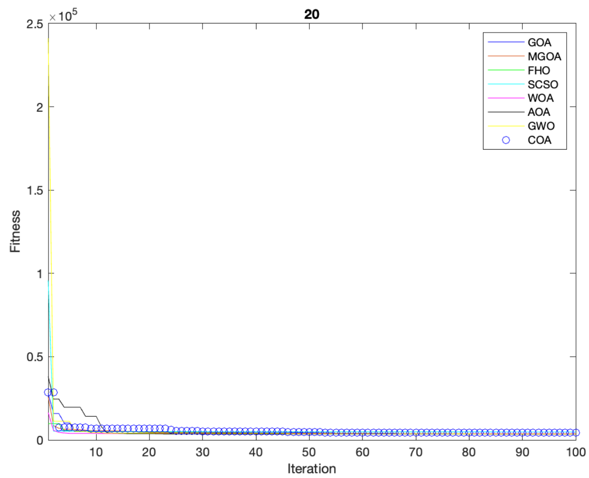
CEC2014-F27



CEC2014-F28



CEC2014-F29



CEC2014-F30

6. Conclusion and future work

The simulation results, encompassing the Comprehensive Economic and Engineering Problems, underscore the robustness of the proposed algorithm. Across benchmarks like the CEC 2014 and CEC 2017 suites, as well as various engineering problems, the algorithm consistently demonstrates superior performance.

Specifically, the algorithm outperforms six state-of-the-art algorithms documented in the literature, alongside the original Gazelle Optimisation Algorithm (GOA). These algorithms represent a diverse array of metaheuristic techniques available in the literature.

Furthermore, rigorous statistical analysis, including the Wilcoxon test, validates the significance of the algorithm's performance. Notably, the Improved Gazelle Optimisation Algorithm (IGOA) exhibits a remarkable ability to yield optimal solutions across most considered problems while maintaining competitiveness in others.

Standard statistical methods further affirm the superiority of IGOA over the selected state-of-the-art algorithms. The findings highlight the algorithm's versatility and efficacy, positioning it as a promising solution for a wide range of real-world problems. These include applications such as parallel machine scheduling, economic load dispatch in electronic sciences, and medical image processing, where its performance surpasses that of previous algorithms and competes admirably with more advanced methodologies..

References

[1] Agushaka, J.O., Ezugwu, A.E. & Abualigah, L. Gazelle optimization algorithm: a novel nature-inspired metaheuristic optimizer. *Neural Comput & Applic* **35**, 4099–4131 (2023). `https://doi.org/10.1007/s00521-022-07854-6`

[2] Mirjalili, Seyedali & Mirjalili, Seyed & Lewis, Andrew. (2014). Grey Wolf Optimizer. Advances in Engineering Software. 69. 46–61. 10.1016/j.advengsoft.2013.12.007.

[3] Azizi, Mahdi & Talatahari, Siamak & Gandomi, Amir. (2022). Fire Hawk Optimizer: a novel metaheuristic algorithm. Artificial Intelligence Review. 56. 1-77. 10.1007/s10462-022-10173-w.

[4] Seyyedabbasi, Amir & Kiani, Farzad. (2022). Sand Cat swarm optimization: a nature-inspired algorithm to solve global optimization problems. Engineering with Computers. 39. 10.1007/s00366-022-01604-x.

[5] Mirjalili, Seyedali & Lewis, Andrew. (2016). The Whale Optimization Algorithm. Advances in Engineering Software. 95. 51-67. 10.1016/j.advengsoft.2016.01.008.

[6] A.Hashim, Fatma & Hussain, Kashif & Houssein, Essam & Mabrouk, Mai & Al-Atabany, Walid. (2021). Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems. Applied Intelligence. 51. 1-21. 10.1007/s10489-020-01893-z.

[7] Jia, Heming & Rao, Honghua & Wen, Changsheng & Mirjalili, Seyedali. (2023). Crayfish optimization algorithm. Artificial Intelligence Review. 1. 1. 10.1007/s10462-023-10567-4.

[8] Einstein A. (1956) Investigations on the Theory of the Brownian Movement. Courier Corporation, US.

[9] Humphries NE, Queiroz N, Dyer JR, Pade NG, Musyl MK, Schaefer KM, Sims DW (2010) Environmental context explains Levy and Brownian movement patterns of marine predators. Nature 465(7301):1066–1069.

[10] Mantegna RN (1994) Fast, accurate algorithm for numerical simulation of Levy stable stochastic processes. Phys Rev E 49(5):4677.

[11] Agushaka, Ovre & Ezugwu, Absalom & Abualigah, Laith. (2021). Gazelle Optimization Algorithm: A novel nature-inspired metaheuristic optimizer for mechanical engineering applications.

[12] Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomi AH (2021) The arithmetic optimization algorithm. Comput Methods Appl Mech Eng 376:113609

[13] Agushaka JO, Ezugwu AE (2020). Diabetes classification techniques: a brief state-of-the-art literature review. In: International Conference on Applied Informatics (pp. 313–329). Ogun: Springer, Cham

[14] Agushaka JO, Ezugwu AE, Abualigah L (2022) Dwarf Mongoose Optimization Algorithm. Comput Methods Appl Mech Eng 391:114570

[15] Agushaka J, Ezugwu A (2020) Influence of initializing krill herd algorithm with low-discrepancy sequences. IEEE Access 8:210886–210909

[16] Akay B, Karaboga D (2012) A modified artificial bee colony algorithm for real-parameter optimization. Inf Sci 192:120–142

[17] Akay B, Karaboga D (2012) Artificial bee colony algorithm for large-scale problems and engineering design optimization. J Intell Manuf 23(4):1001–1014

[18] Atashpaz-Gargari E, Lucas C (2007). Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In 2007 IEEE congress on evolutionary computation (pp. 4661–4667). Ieee

[19] Biswas A, Mishra K, Tiwari S, Misra A (2013). Physics-inspired optimization algorithms: a survey. Journal of Optimization, 2013

[20] Dorigo, M., & Di Caro, G. (1999). Ant colony optimization: a new meta-heuristic. In: Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406) (Vol. 2) (pp. 1470–1477). IEEE

[21] Ezugwu AE (2020) Nature-inspired metaheuristic techniques for automatic clustering: a survey and performance study. SN Applied Sciences 2(2):273

[22] Ezugwu AE, Adeleke OJ, Akinyelu AA, Viriri S (2020) A conceptual comparison of several metaheuristic algorithms on continuous optimization problems. Neural Comput Appl 32(10):6207–6251

[23] Ezugwu AE, Agushaka JO, Abualigah L, Mirjalili S, Gandomi AH (2022) Prairie dog optimization algorithm. Neural Comput Appl. https://doi.org/10.1007/s00521-022-07530-9

[24] Ezugwu AE, Shukla AK, Nath R, Akinyelu AA, Agushaka JO, Chiroma H, Muhuri PK (2021) Metaheuristics: a comprehensive overview and classification along with bibliometric analysis. Artif Intell Rev 54(6):4237–4316

[25] Holland, J. H. (1975). Adaptation in natural and artificial systems. University of Michigan Press. (Second. Michigan: University of Michigan Press. (Second edition: MIT Press, 1992)

[26] Kaveh A, Hamedani KB, Kamalinejad M (2022) Improved slime mould algorithm with elitist strategy and its application to structural optimization with natural frequency constraints. Com- put Struct 264:106760

[27] Kennedy J, Eberhart R (1995). Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks (Vol. 4) (pp. 1942–1948). IEEE

[28] Kirkpatrick S, Gelatt Jr CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220(4598):671–680

[29] Liu H, Zhang X, Liang H, Tu L (2020) Stability analysis of the human behavior-based particle swarm optimization without stagnation assumption. ExpertSystAppl 159:113638

[30] Mohammadi F, Amini M, Arabnia H (2020). Evolutionary computation, optimization, and learning algorithms for data science. Optimization, Learning, and Control for Interdependent Complex Networks.Cham, Switzerland: Springer, 37–65

[31] Oyelade ON, Ezugwu AE-S, Mohamed TI, Abualigah L (2022) Ebola optimization search algorithm: A new nature-inspired metaheuristic optimization algorithm IEEE. Access 10:16150–16177

[32] Pelusi D, Mascella R, Tallini L, Nayak J, Naik B, Deng Y (2020) Improving exploration and exploitation via a hyperbolic gravitational search algorithm. Knowl-Based Syst 193:105

[33] Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179(13):2232–2248

[34] Sarzaeim P, Bozorg-Haddad O, Chu X (2018). Teaching-learning-based optimization (TLBO) algorithm. Advanced Optimization by Nature-Inspired Algorithms. Singapore, Asia: Springer, 51–58

[35] Storn R, Price K (1997) Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11(4):341–359

[36] Uymaz SA, Tezel G, Yel E (2015) Artificial algae algorithm (AAA) for nonlinear global optimization. Appl Soft Comput 31:153–171

[37] Xing B, Gao W (2014). Invasive weed optimization algorithm. Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms. Cham, Switzerland:Springer, 177–181

[38] Yang X, Karamanoglu M (2020). Nature-inspired computation and swarm intelligence: a state-of-the-art overview. Nature-Inspired Computation and Swarm Intelligence. Cambridge, Massachusetts: Academic Press, 3–18

[39] Zhang P, Wang C, Qin Z, Cao H (2022) A multidomain virtual network embedding algorithm based on multiobjective optimization for Internet of Drones architecture in Industry 40. Softw Pract Exper 52(3):710–728

[40] Belkourchia, Yassin & Azrar, Lahcen & Es-sadek, Mohamed. (2019). A Hybrid Optimization Algorithm for Solving Constrained Engineering Design Problems. 1-7. 10.1109/ICOA.2019.8727654.

[41] Yıldız, Betül & Pholdee, Nantiwat & Bureerat, Sujin & Yildiz, Ali & Sait, Sadiq. (2021). Enhanced grasshopper optimization algorithm using elite opposition-based learning for solving real-world engineering problems. Engineering with Computers. 38. 10.1007/s00366-021-01368-w.

[42] Gao, Cong & Hu, Zhongbo & Xiong, Zenggang & Su, Qinghua. (2020). Grey Prediction Evolution Algorithm Based on Accelerated Even Grey Model. IEEE Access. PP. 1-1. 10.1109/ACCESS.2020.3001194.

[43] Melda Yücel, Gebrail Bekdaş, Sinan Melih Nigdeli.Minimizing the Weight of Cantilever Beam via Metaheuristic Methods by Using Different Population-Iteration Combinations. WSEAS TRANSACTIONS on COMPUTERS DOI: 10.37394/23205.2020.19.10: https://wseas.com/journals/computers/2020/a205105-059.pdf

[44] Agushaka JO, Ezugwu AE (2021) Advanced arithmetic opti- mization algorithm for solving mechanical engineering design problems. PLoS ONE. https://doi.org/10.1371/journal.pone. 0255703