Journal Européen des Systèmes Automatisés
Vol. 57, No. 5, October, 2024, pp. 1337-1347
Journal homepage: http://iieta.org/journals/jesa

An Improved Multi-Objective Approach Based on Competitive Optimization Algorithm and Its Engineering Design Application

Bilel Najlaoui[1*], M. Shameer Basha[2], Eihab A. Raouf[2], Hassan Abd El-Hafez[2,3]

[1] Mechanical Engineering Laboratory, National Engineering School of Monastir, University of Monastir, Monastir 5000, Tunisia
[2] Department of Mechanical Engineering, College of Engineering, Qassim University, Buraidah 51452, Saudi Arabia
[3] Faculty of Engineering, Port Said University, Port Said 42523, Egypt

Corresponding Author Email: najlawibilelali@gmail.com

https://doi.org/10.18280/jesa.570509

Received: 23 July 2024
Revised: 29 September 2024
Accepted: 10 October 2024
Available online: 28 October 2024

Keywords:
CCA, computational time, convergence, diversity, multi-objective optimization, engineering design, sorting non-dominated strategy

ABSTRACT

In decision science and engineering, multi-objective optimization is crucial for the understanding of real-world systems. The recently proposed colonial competitive algorithm (CCA) is a successful method in mono-objective optimization. Nevertheless, CCA cannot handle simultaneously the conflicting objectives in multi-objective problem. In addition, its performance diminishes as the dimension of problem increases. In order to deal with these situations, in this work, an improved CCA, named multi-objective improved colonial competitive algorithm (MICCA) is proposed. To enhance the MICCA effectiveness in achieving the global optima, a new approach for the colonies movement toward the imperialists, named enhanced assimilation, is incorporated. Moreover, in contrast to CCA, the Pareto dominance approach is implemented into this algorithm to allow it to tackle problems that involve conflicting objectives. To assess the MICCA efficiency, computational time, convergence and diversity metrics are used. Extensive simulation studies are carried out on standard unconstrained, constrained and multi-criteria engineering design problems. The obtained results are compared to the outcomes of existing CCA variants and other algorithms using the same metrics. The comparative results illustrate that for a wide variety of challenges and engineering applications, the proposed MICCA outperforms other known algorithms in terms of computational time, convergence and global search capabilities.

1. INTRODUCTION

Multi-objective optimization is a decision-making method that takes into account the simultaneous optimization of real-world engineering difficulties that have multiple objective functions. Three key challenges must be overcome in order to tackle real-world multi-objective difficulties. These key challenges include exploitation, exploration, and multi-objectivity.

Firstly, exploitation is the capacity to conduct local searches in and around attractive research areas, which promotes strong convergence toward the global optimum. Secondly, exploration is the process of thoroughly investigating the promising regions of the research area, which produces a wide variety of optimum solutions. Thirdly, there are several conflicting targets for optimizing simultaneously in many multi-objective engineering design issues. To address these challenges, a multi-objective optimization strategy should be used.

Developing a multiobjective algorithm that balances exploitation and exploration is a difficult undertaking. Optimization methods are often categorized as stochastic or deterministic ones. The majority of deterministic methods are grounded on the gradient approach. They could be bothersome with multimodal functions as well as functions having flat sections with low gradient. Unlike deterministic algorithms, stochastic ones can eschew local optimum, particularly with many conflicting objective challenges [1]. Differential evolution method is considered by the study [2]. Enhanced particle swarm optimization (PSO) is investigated by the studies [3, 4]. Many genetic optimizer variants are applied to optimize multiobjective issues [5-8].

The CCA, inspired by imperialist countries' social-political conduct, is established [9] and applied in various challenges. The CCA has been utilized for well predicting oil rates using fuzzy logic [10]. mining data [11], routing of vehicles [12], optimizing the design of a mechanism able to of reproducing the motion of the knee joint [13], improving solar collector thermal efficiency [14], Stirling engine simulation [15], traveling salesmen [16, 17], optimizing the design of sewing machine [18, 19], composite structures [20, 21] as well as various engineering design challenges. With a focus on engineering fields, Hosseini and Al Khaled [22] highlighted the fundamental theories that led to the development of CCA and also its implementation in several engineering disciplines.

The CCA method has several advantages, including the simplicity of executing neighborhood movement, less reliance on starting solutions, and a shorter computation time [9, 22].

Nevertheless, the CCA method has drawbacks. One of the drawbacks of the CCA is competition among empires. If the quality improvement approach for empires is not strong enough, competition occurs too often, and weak empires get eliminated quickly. In this condition, population diversity quickly degrades, and consequently, the algorithm is trapped in local optima due to the loss of diversity. These negative points may cause premature convergence toa local optimum position. Adding a technique for enhancing empires quality can improve the algorithm's effectiveness to reach the location of the globally optimal target in the search space. Moreover, in challenges involving multiple conflicting objectives, CCA is unable to tackle them all simultaneously [22-25]. Until recently, despite its efficiency in mono-objective optimization, and to our best of knowledge, the literature does not contain any work dealing with the application of the original CCA to solve multi-objective optimization problems.

Several studies incorporate CCA with other methods to address multiobjective engineering challenges. To improve the power system's electric power planning, Ghasemi et al. [24] suggested an enhanced algorithm by incorporating the Gaussian bare-bones method with the CCA optimizer. Ghasemi et al. [25] introduced a hybrid algorithm by coupling the CCA and the teaching learning (TLA) method to tackle the optimum power flow problem with non-smooth cost functions. To solve the profit-based unit commitment problem, an upgraded colonial competitive method (CICA3-LCM) is developed and used [26]. Armaghani et al. [27] proposed an intelligent system by combining artificial neural networks and colonial competitive algorithm to predict rate of underground boring machinery penetration.

Bilel et al. [28] suggested a multiobjective version of CCA named MOICA. However, the shortcoming of CCA in terms of convergence to the local optimal result remains persists in this method. Bilel et al. [29] proposed an extended version of CCA by incorporating the variable neighborhood search (VNSA) technique in the CCA algorithm. Nevertheless, the VNSA approach used [29] altered the coefficient of assimilation, which can enhance only the global search capability without taking the algorithm's exploration capability into account.

The main goal of this study is to expand the CCA algorithm to solve multi-objective optimization problems by developing a multi-objective improved colonial competitive algorithm (MICCA) based on the Pareto front concept. We implement also a new approach named enhanced assimilation, in the assimilation step, in order to improve the performances of the algorithm to reach the global optimal position. The performances of MICCA are tested through well-known unconstrained, constrained multi-objective problems, and several multi-criteria engineering design challenges. Compared with the existing CCA variants and other well-known evolutionary algorithms, experimental results demonstrate that the proposed method (MICCA) can effectively handle simultaneously the conflicting objectives in multi-objective design problem, converge towards the global optimum, and operate promising areas of the search space.

2. THE CCA OPTIMIZER

The CCA is a worldwide search method proposed [9] that draws inspiration from either the sociopolitical human evolution. Figure 1 illustrates the CCA flowchart. This method is made up of four basic phases, which are listed below.

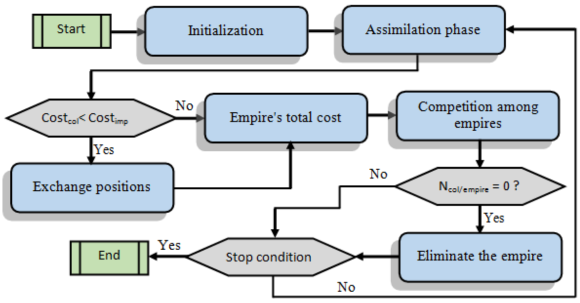


Figure 1. The CCA flowchart

Phase 1 - Initialization

The first phase of CCA method is to form the initial countries at random. The country's cost is determined by evaluating the country's performances. Countries are classified as "imperialists" or "colonies" according to their costs. The nth imperialist's normalized cost (Cn) and normalized power (P_n) are determined by:

$$C_n = \max_i \{Cost_i\} - Cost_n \tag{1}$$

$$P_n = \left| \frac{C_n}{\sum_i C_i} \right| \tag{2}$$

$Cost_i$ denotes the ith imperialist's cost. Each imperialist has a number of colonies under its control according to its normalized power. So, an empire is defined by the associated colonies and its relevant imperialist.

Phase 2 - Assimilation phase

After initializing empires, colonies move toward their imperialists to growth more power. Eq. (3) illustrates this movement's mathematical relationship [9].

$$P' = P + \beta \times \{v\} \otimes (\ell - P) \tag{3}$$

P' denotes the new position of the colony. β represents a coefficient exceeding one. v is a randomly vector utilized to present motion deviations for better use. ℓ and P are imperialist and colonial positions, respectively. Figure 2 depicts the colonial to imperialist movement. Z and θ generated randomly by the uniform distribution within, respectively, the bounds of U (0, β×e) and U (-δ, δ). δ and e are a random angle and a colony distance to imperialist.

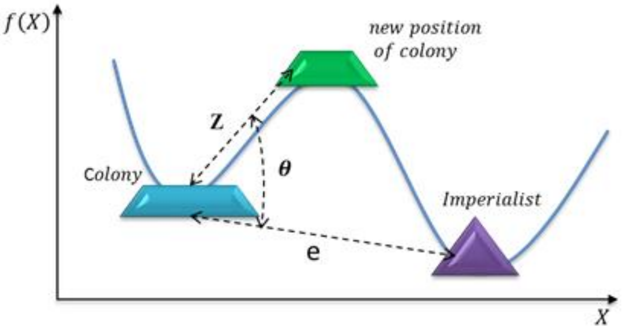


Figure 2. Colonial to imperialist movement

Assimilation can lead to a situation in which a colony outperforms its imperialist. In this situation, the colonies' and imperialists' positions are exchanged.

Phase 3 - Empire's total cost

The empire's total cost is computed by adding the cost of the imperialist to a percent of the cost of the colonies based on Eq. (4):

$$TC_n = Cost_n + \xi \text{ mean}\left(Cost_{col,\, n}\right) \text{ and } 0 < \xi < 1 \qquad (4)$$

The empire's normalized cost is computed by:

$$NTC_n = \max_i \{TC_i\} - TC_n \qquad (5)$$

Phase 4 - Competition among empires

During this phase, the powerful empires acquire weak colonies based on its possession likelihood (Figure 3). The possession likelihood is determined based on Eq. (6):

$$P_{p_n} = \left| \frac{NTC_n}{\sum_i NTC_i} \right| \qquad (6)$$

As weak empires collapse, powerful empires gradually seize control of all their colonies. The most powerful empire remains after multiple competitions. This situation corresponds to the CCA stop condition.

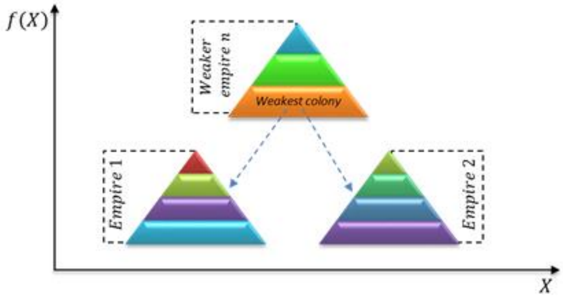


Figure 3. Competition among empires

3. MICCA ALGORITHM

In the following, the MICCA algorithm, as a variant of the original CCA, is suggested (Figure 4).

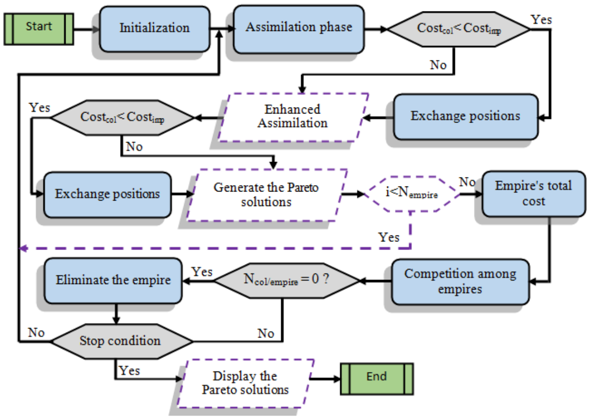


Figure 4. MICCA flowchart

On the original CCA, two major changes are implemented:

• Incorporation of an enhanced approach for assimilation to enhance the algorithm's global search and to boost its exploration capabilities.

• Integration of the sorting non dominated approach (SNDA) to determine Pareto optimum solutions for numerous conflicting functions.

In what fellow, the additional phases in the original CCA are going to be explained.

3.1 Enhanced assimilation phase

Following the usual assimilation stage, countries are undergoing an enhancement assimilation phase. All countries turn toward the finest imperialist when it is identified. As a result, the population seems to converge in a certain part of a search space field that is the finest imperialist neighborhood. The country's newfound position can be expressed as:

$$P'' = P' + \beta' \times \{v\} \otimes (P_{best} - P') \qquad (7)$$

The best positions for imperialist and country are indicated by P_{best} and P', respectively.

$$\beta' = 1 + \left(\frac{1}{P'}\right) \qquad (8)$$

When we compare the movements of countries in enhanced assimilation phase (provided by Eq. (7)) to those in usual assimilation phase (defined by Eq. (3)), we can see two distinctions. First, during the phase of enhancing assimilation, all countries advance toward the best imperialism in each iteration, improving their positions, particularly those of the imperialists, acquired through the assimilation stage and causing them to converge on the best solution established in that iteration. Second, in enhanced assimilation, β' is regarded as a variable coefficient (Eq. (8)), allowing the algorithm to cover the convergence region accurately and look for more solutions in the search space.

3.2 Generation of pareto front

The SNDA technique (Figure 5) is used for representing the most appropriate solutions on a Pareto front [7]. This technique begins on the first iteration with the choosing of the N imperialist. A counter variable (n_I) is assigned to each imperialist. Each time, an imperialist (Ii) is compared against the other imperialists (Rj). When this imperialist is dominated, its counter variable (n_{Ii}) rises. n_{Ii}=0 for the non dominated imperialist.

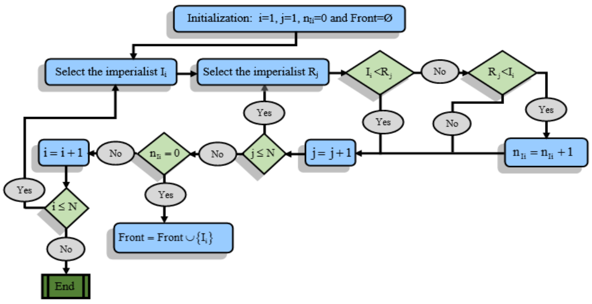


Figure 5. Sorting non dominated (SNDA) approach

At every comparison, the non dominated imperialist will be archived to form the Pareto front. Following N comparisons, this Pareto front is able to be formed.

All subsequent iterations will compare all other imperialists to the prior Pareto front solutions.

4. PERFORMANCES EVALUATION FOR BENCHMARK FUNCTIONS

4.1 Performance metrics

To assess the MICCA performances, the following indicators will be utilized:

- Convergence indicator ϒ: it measures the distance between the obtained non-dominated solutions and the true Pareto front, corresponding to the exact solutions [7].

$$\Upsilon = \frac{\sum_{i=1}^{N} d_{min}}{N} \tag{9}$$

where, N represents the number of non-dominated solutions provided by the respective test functions; d_{min} denotes the minimal Euclidean distance that separates the MICCA Pareto front's non-dominated ith solution from its closest on the true Pareto front.

- Diversity indicator Δ: it assesses the nature and variety of solution distribution on the Pareto front [7]:

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} \left| d_i - \bar{d} \right|}{d_f + d_l + (N-1)\bar{d}} \tag{10}$$

where, d_i represents the Euclidean distance that separates the resulting non-dominated front's successive solutions and $\bar{d}$ designates the mean of these Euclidean distances. D_f and d_l indicate the Euclidean distance separating the boundary outcomes of the acquired non-dominated front and the most extreme outcomes of a true Pareto front.

- Computational Time (CT): this metric measures the runtime of an algorithm for different challenges.
- Standard deviation (STD) calculated over the outcomes of 10 runs.

$$\mathrm{STD}(P) = \sqrt{\frac{1}{R} \sum_{i=1}^{R} \left(P_i - \bar{P} \right)^2} \tag{11}$$

where, R and P_i are the number of runs and the performance indicator's value, respectively.

4.2 Benchmark functions

To assess the MICCA performances to handle multi-objective challenges, different benchmark problems are used. These challenges are divided into two groups [30-34]:

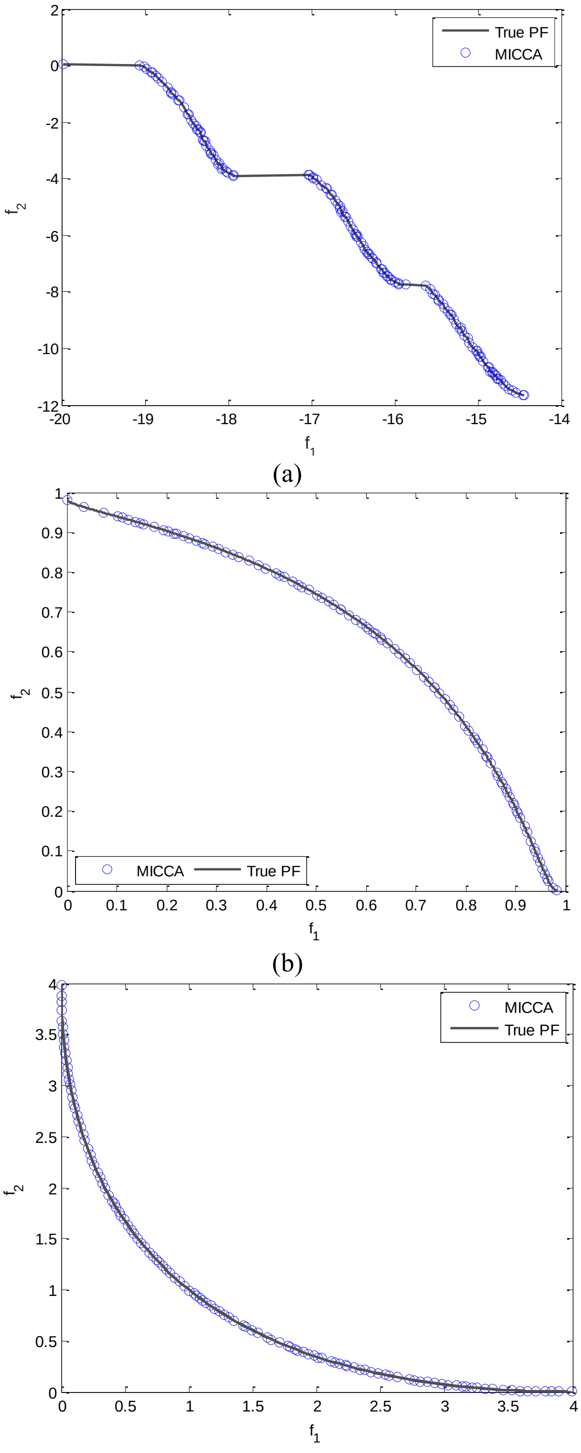
- Unconstrained multi-objective problems: KUR, FON, SCH1, SCH2 and ZDT1 – ZDT4.
- Constrained multi-objective problems: DTLZ8, DTLZ9, BINH2, OSY, CONSTR, TNK, SRN and KITA.

For the sake of comparison, the number of function evaluations is set to 25000 and the MICCA parameters are set as (β=1.2 and ζ=0.02).

4.3 Obtained results

Figures 6 and 7 show the outcomes of the MICCA approach and their related true Pareto fronts for both unconstrained and constrained issues. Tables 1-6 provide the statistical findings achieved by MICCA and other well-known algorithms. The ϒ, Δ and CT metrics are used to present the findings. Moreover, the minimum performance improvement (MPI) of MICCA results, in terms of (ϒ, Δ and CT) metrics, compared to other literature methods are highlighted in Tables 1-6.

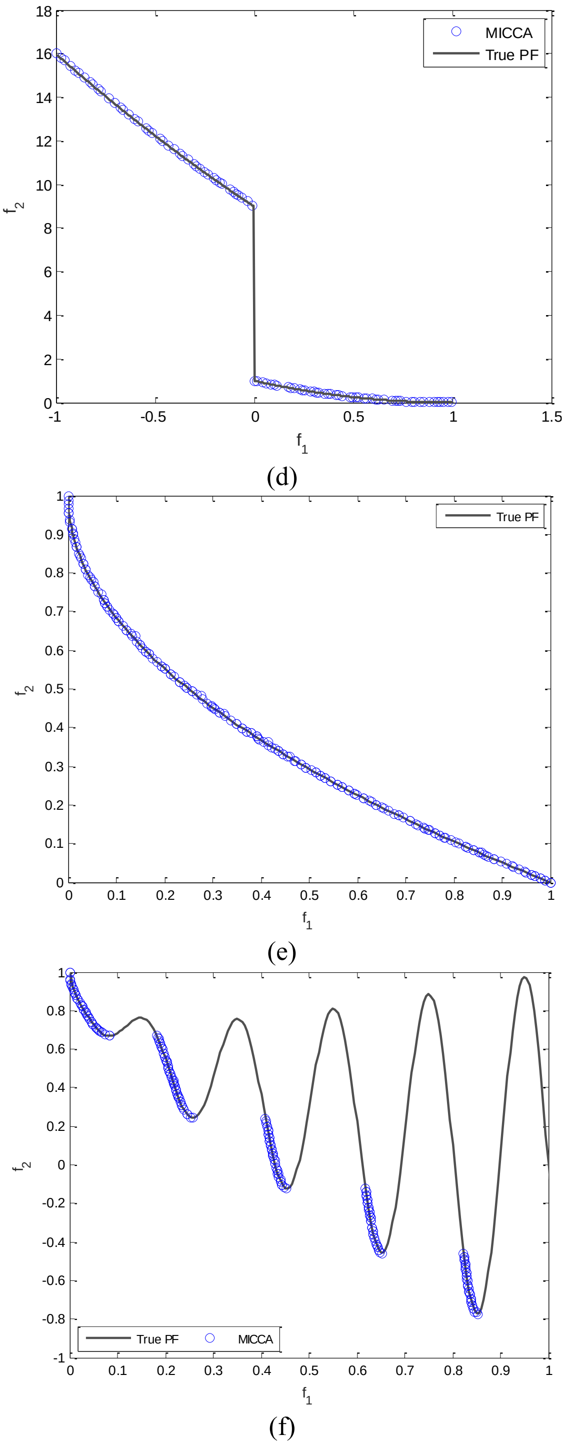
From Figures 6 and 7, the MICCA method's high convergence and solution diversity skills can be shown in all test issues. The MICCA non-dominated solutions are, in fact, extremely close to the true Pareto front.



(a)

(b)

(c)



(d)

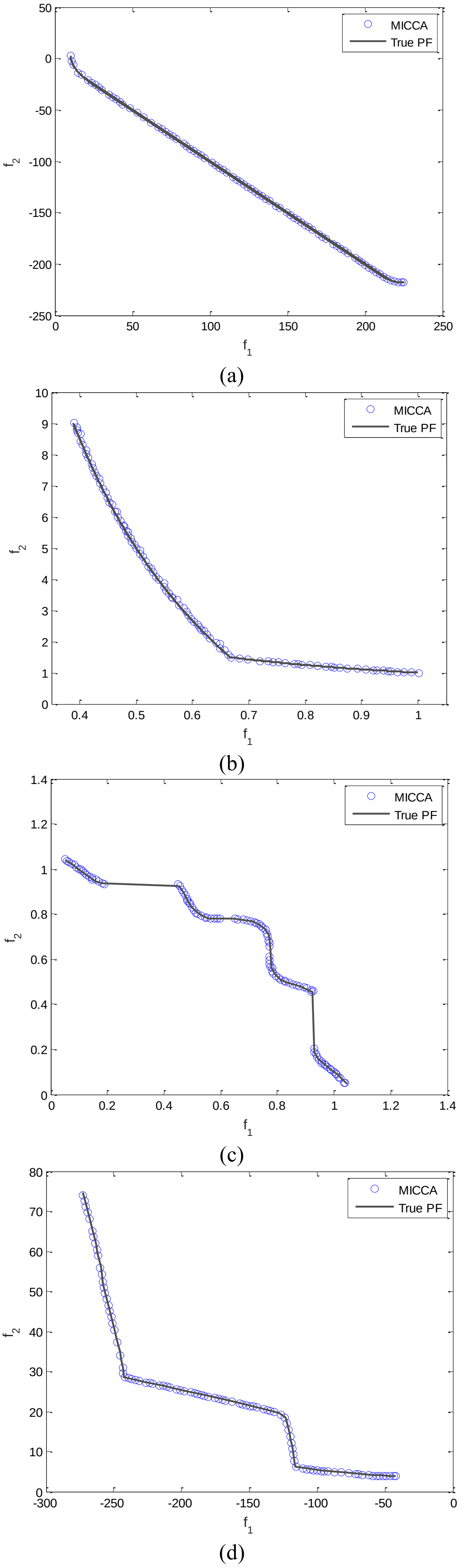
(e)

(f)

Figure 6. True and MICCA Pareto fronts of unconstrained functions: a) KUR; b) FON; c) SCH1; d) SCH2; e) ZDT1; f) ZDT3

From Tables 1-6 and Figures 8 and 9, the small ϒ, Δ and CT values are obtained by MICCA for all unconstrained and constrained problems. Moreover, from the MPI results, reported in Tables 1-6, one can note that the MICCA algorithm is capable of finding reliable and best solutions for all multi-objective optimization problems compared to other literature methods. Furthermore, when compared to the variances (determined based on STD) acquired by the other methods, the variance values achieved by the MICCA are quite tiny in all circumstances.

MICCA surpasses the other algorithms in all measures (ϒ, Δ, CT and STD), demonstrating its high convergence, diversity, robustness with small computational time values for nearly unconstrained and constrained benchmark problems.

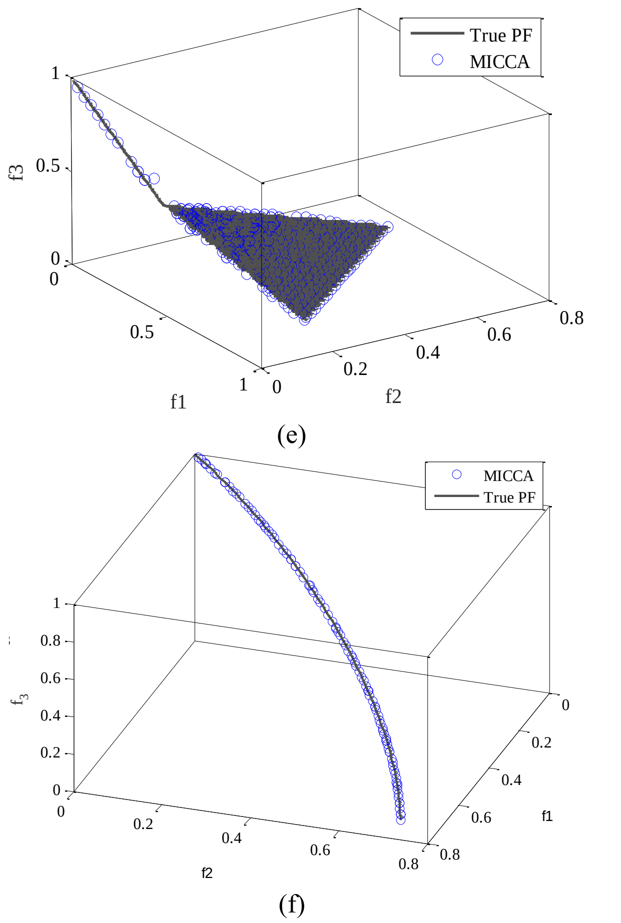


(a)

(b)

(c)

(d)



(e)

(f)

Figure 7. True and MICCA Pareto fronts of constrained issues: a) SRN; b) CONSTR; c) TNK; d) OSY; e) DTLZ8; f) DTLZ9

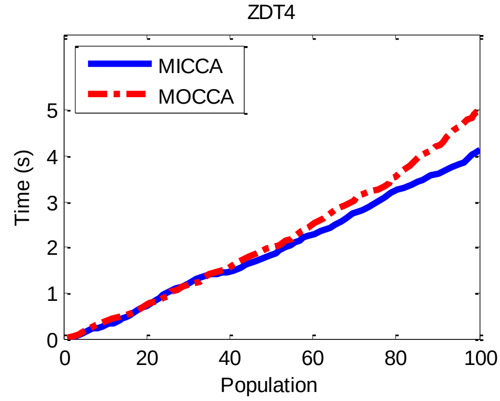


Figure 8. Comparing computational time between MICCA and MOCCA for ZDT4 problem

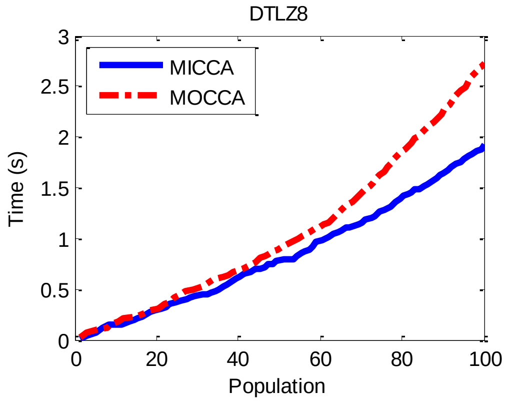


Figure 9. Comparing computational time between MICCA and MOCCA for DTLZ8 problem

Table 1. Mean and STD values ϒ on unconstrained challenges

Function	MOCCA [29]	NSGWO [30]	MOCBO [31]	MOAAA [32]	MOSGA [33]	MOSOS [31]	MICCA	MPI of MICCA (%)
KUR	N/A	0.0073 ± 0.0026	0.0083 ± 0.0062	1.59E−04 ± 8.1E−06	1.967E−04 ± 1.160E−05	0.0075 ± 0.0042	**7.929E-05 ± 4.701E-06**	50.13
FON	0.001185 ± 0.00000	0.0017 ± 0.0001	0.0022 ± 0.0003	2.82E−04 ± 1.0E−05	1.922E−04 ± 6.272E−06	0.0019 ± 0.0002	**6.438E-05 ± 3.233E-06**	66.5
SCH1	0.00046 ± 0.000001	0.0072 ± 0.0006	0.0031 ± 0.0032	3.13E−02 ± 2.3E−02	1.995E−04 ± 1.125E−05	0.0028 ± 0.0024	**8.102E-05 ± 1.061E-05**	59.38
SCH2	N/A	0.0489 ± 0.0017	0.0932 ± 0.0228	N/A	N/A	0.0705 ± 0.0215	**3.277E-04 ± 8.816E-04**	99.1
ZDT1	N/A	0.4568 ± 0.0654	0.3337 ± 0.0319	2.30E−04 ± 9.8E−06	1.846E−04 ± 8.510E−06	0.3325 ± 0.0256	**6.011E-05 ± 1.772E-06**	67.43
ZDT2	N/A	0.0700 ± 0.0008	0.0729 ± 0.0005	2.38E−04 ± 1.4E−05	1.902E−04 ± 6.674E−06	0.0731 ± 0.0010	**5.904E-05 ± 5.707E-06**	68.96
ZDT3	0.00019 ± 0.000004	0.0867 ± 0.0451	0.0982 ± 0.5007	**1.58E−04 ± 9.7E−06**	7.695E−03 ± 1.916E−05	0.1022 ± 0.5187	2.103E-04 ± 1.825E-05	N/A
ZDT4	0.00012 ± 0.000000	0.4785 ± 0.0001	0.5078 ± 0.0013	2.40E−03 ± 2.9E−03	5.868E−05 ± 2.819E−06	0.5015 ± 0.0006	**1.468E-05 ± 2.095E-06**	74.97

N/A=Not Available. MPI of MICCA=Minimum performance improvement of MICCA

Table 2. Mean and STD values Δ on unconstrained challenges

Function	MOCCA [29]	NSGWO [30]	MOCBO [31]	MOAAA [32]	MOSGA [33]	MOSOS [31]	MICCA	MPI of MICCA (%)
KUR	N/A	0.0271 ± 0.0001	0.0357 ± 0.0236	4.98E−01 ± 1.9E−02	5.74E−01 ± 2.27E−02	0.0295 ± 0.012	**7.641E-03 ± 9.836E-05**	71.82
FON	0.00818 ± 0.000014	0.3578 ± 0.0478	0.3955 ± 0.0068	2.89E−01 ± 2.2E−02	3.80E−01 ± 2.74E−02	0.3875 ± 0.006	**6.282E-03 ± 4.009E-03**	23.20
SCH1	0.00704 ±	0.5000 ±	0.5302 ±	6.43E−01 ±	3.52E−01 ±	0.5295 ±	**1.505E-03 ±**	78.62

	0.000281	0.0124	0.1356	2.8E−01	2.19E−02	0.131	**1.071E-02**	
SCH2	N/A	0.7589 ± 0.0166	0.8010 ± 0.0832	N/A	N/A	0.7821 ± 0.051	**5.794E-01 ± 9.882E-03**	23.65
ZDT1	N/A	0.3651 ± 0.0068	0.3825 ± 0.0125	6.40E−01 ± 5.1E−02	3.53E−01 ± 2.34E−02	0.380 ± 0.0122	**2.615E-01 ± 5.013E-03**	26.05
ZDT2	N/A	**0.0405 ± 0.0758**	0.4316 ± 0.0007	6.79E−01 ± 6.5E−02	3.24E−01 ± 1.32E−02	0.430 ± 0.0007	4.839E-02 ± 1.392E-04	N/A
ZDT3	0.00974 ± 0.000019	0.6887 ± 0.2164	0.6532 ± 0.0020	8.08E−01 ± 2.4E−02	7.85E−01 ± 1.24E−02	0.6537 ± 0.005	**3.949E-03 ± 1.057E-03**	59.45
ZDT4	0.06208 ± 0.000008	0.4256 ± 0.0124	0.4795 ± 0.0079	5.85E−01 ± 1.7E−01	3.82E−01 ± 3.52E−02	0.4585 ± 0.007	**1.248E-02 ± 4.153E-04**	79.89

Table 3. Mean and STD values CT on unconstrained challenges

Function	MOCCA [29]	NSGWO [30]	MOCBO [31]	MOPSO [30]	NSGAII [30]	MOSOS [31]	MICCA	MPI of MICCA (%)
KUR	7.7013 ± 0.1502	7.68719 ± 0.39875	7.9531 ± 0.58234	8.0532 ± 0.621	20.4368 ± 3.102	10.7413 ± 0.82246	**5.18401 ± 0.10546**	32.56
FON	7.8935 ± 0.3201	9.7895 ± 0.45897	8.6606 ± 0.88622	8.732 ± 0.9134	22.0323 ± 4.522	11.4013 ± 1.14004	**4.82353 ± 0.10972**	38.89
SCH1	4.9692 ± 0.1094	11.8500 ± 1.21458	5.4845 ± 1.13208	5.5721 ± 1.133	17.9121 ± 2.162	8.2135 ± 1.12122	**3.56902 ± 0.09104**	28.17
SCH2	5.0117 ± 0.2194	5.77985 ± 0.12447	5.9751 ± 0.28216	6.0272 ± 0.582	18.421 ± 2.1802	8.7015 ± 0.45322	**3.87040 ± 0.09273**	22.77
ZDT1	3.4507 ± 0.0168	5.45345 ± 0.00295	3.1435 ± 0.0193	3.7533 ± 0.006	11.2681 ± 0.364	8.2351 ± 0.0204	**1.49195 ± 0.00101**	52.53
ZDT2	6.4711 ± 0.1006	3.13452 ± 0.03000	3.1502 ± 0.0130	3.6113 ± 0.014	11.2811 ± 0.024	8.2345 ± 0.0457	**1.07933 ± 0.00728**	65.56
ZDT3	5.0163 ± 0.1172	10.6325 ± 0.32245	6.2846 ± 0.1059	8.3764 ± 0.231	14.3406 ± 0.144	13.4567 ± 0.12984	**4.00121 ± 0.02711**	20.23
ZDT4	4.9602 ± 0.0914	7.11523 ± 0.45698	6.6922 ± 0.1440	8.8203 ± 0.218	14.8102 ± 0.170	13.9022 ± 0.12110	**4.10480 ± 0.03170**	17.24

Table 4. Mean and STD values Υ on constrained challenges

Function	NSGWO [30]	MOSGA [33]	paε-ODEMO [34]	NSGAII [34]	MOSOS [31]	MICCA	MPI of MICCA (%)
BINH2	1.66E-01 ± 5.97E-03	7.00E-05 ± 5.78E-06	N/A	N/A	1.43E-01 ± 6.24E-03	**3.929E-05 ± 9.701E-08**	43.87
CONSTR	4.89E-01 ± 1.80E-02	2.05E-04 ± 7.018E-06	N/A	N/A	5.16E-01 ± 2.14E-03	**6.134E-05 ± 1.328E-07**	70.07
KITA	N/A	2.19E-04 ± 9.26E-06	0.0196 ± 0.0014	0.0719 ± 0.0764	3.68E-02 ± 8.30E-03	**8.102E-05 ± 4.960E-07**	63
SRN	6.98E-02 ± 1.78E-02	9.49E-05 ± 7.17E-06	0.0657 ± 0.0088	0.2397 ± 0.0345	9.88E-02 ± 1.47E-03	**3.277E-05 ± 2.816E-07**	65.46
TNK	1.47E-01 ± 3.35E-03	2.25E-04 ± 1.34E-05	0.0014 ± 2.28E-4	0.002 ± 3.59E-4	1.50E-01 ± 4.04E-03	**3.510E-05 ± 1.772E-06**	84.4
OSY	0.10002 ± 0.0005	N/A	1.3183 ± 0.1635	1.5614 ± 0.2231	0.1196 ± 0.0031	**7.904E-05 ± 3.707E-05**	99
DTLZ8	N/A	N/A	0.0098 ± 7.86E-4	0.0354 ± 0.0059	N/A	**5.403E-04 ± 8.825E-05**	94.4
DTLZ9	N/A	N/A	0.0201 ± 0.0057	0.0253 ± 0.0035	N/A	**1.468E-04 ± 2.995E-04**	99

N/A=Not Available. MPI of MICCA=Minimum performance improvement of MICCA

Table 5. Mean and STD values Δ on constrained challenges

Function	NSGWO [30]	MOSGA [33]	paε-ODEMO [34]	NSGAII [34]	MOSOS [31]	MICCA	MPI of MICCA (%)
BINH2	4.879E-01 ± 8.965E-02	4.369E-01 ± 3.265E-02	N/A	N/A	0.4288 ± 0.0625	**1.641E-01 ± 2.936E-03**	61.73
CONSTR	6.598E-01 ± 5.69E-04	4.263E-01 ± 1.824E-02	N/A	N/A	0.7122 ± 0.0072	**2.002E-01 ± 1.449E-04**	53.03
KITA	N/A	3.754E-01 ± 5.987E-02	0.29728 ± 0.0324	0.8106 ± 0.14839	0.6832 ± 0.0072	**6.175E-02 ± 8.717E-04**	79.22
SRN	2.001E−01 ± 6.5E−04	3.892E-01 ± 3.142E-02	0.15080 ± 0.0163	0.3906 ± 0.04516	0.2295 ± 0.0017	**8.794E-02 ± 7.992E-04**	41.68
TNK	9.955E-02 ±	7.425E-01 ±	0.3674 ±	0.8480 ±	0.1206 ±	**4.415E-02 ±**	55.65

	2.568E-02	2.750E-02	0.04221	0.10055	0.0423	**5.013E-03**	
OSY	0.65897 ± 0.06779	N/A	0.8010 ± 0.07555	0.7481 ± 0.08818	0.5354 ± 0.0616	**3.129E-01 ± 1.392E-03**	41.55
DTLZ8	N/A	N/A	**0.3484 ± 0.02166**	0.5842 ± 0.03546	N/A	3.520E-01 ± 4.977E-03	N/A
DTLZ9	N/A	N/A	0.7459 ± 0.02456	0.7863 ± 0.03416	N/A	**3.087E-01 ± 1.307E-03**	58.61

Table 6. Mean and STD values CT on constrained challenges

Function	MOCCA [29]	NSGWO [30]	MOCBO [31]	paε-ODEMO [34]	NSGAII [34]	MOSOS [31]	MICCA	MPI of MICCA (%)
BINH2	2.7305±0.0952	8.5685 ± 0.06895	9.1544 ± 0.0420	N/A	N/A	16.2664 ± 0.054	**0.8417 ± 0.0052**	69.17
CONSTR	3.0835±0.1756	12.895 ± 0.00711	5.2252 ± 0.0028	N/A	N/A	10.0112 ± 0.003	**0.9282 ± 0.0019**	69.89
KITA	1.0112±0.1044	N/A	10.3245 0.10168	0.11532 ± 0.00767	0.1734 ± 0.0100	14.3821 ± 0.12046	**0.1059 ±0 .0071**	8.16
SRN	1.2507±0.0194	7.2440 ± 0.00119	7.3251 ± 0.0082	0.14470 ± 0.00697	0.2218 ± 0.0141	12.3254 ± 0.01274	**0.1194 ± 0.0028**	17.48
TNK	3.0413±0.0618	9.1156 ± 0.05899	11.0104 ± 0.052	0.42648 ± 0.02604	0.4993 ± 0.0215	15.1286 ± 0.06332	**0.4151 ± 0.0013**	2.74
OSY	1.9711±0.1006	16.4698 ± 0.02018	12.2104 ± 0.030	0.64000 ± 0.02640	0.7264 ± 0.0236	20.2124 ± 0.03242	**0.6023 ± 0.0092**	6.25
DTLZ8	2.6920±0.1132	N/A	N/A	2.03898 ± 0.07374	2.7562 ± 0.1868	N/A	**1.7949 ± 0.0097**	11.97
DTLZ9	3.0591±0.0904	N/A	N/A	2.09844 ± 0.05457	2.5643 ± 0.1225	N/A	**1.2480 ± 0.0057**	40.52

5. PERFORMANCE EVALUATION FOR MULTI-CRITERIA ENGINEERING CHALLENGES

The MICCA technique will be used to tackle multi-criteria engineering design challenges in this part. After that, the collected outcomes will be compared to literature.

5.1 Bi-objective speed reducer issue

A speed reducer is part of the gear box of mechanical system, and it is used in many other types of applications. The challenge of speed reducer design (Figure10) is a well-known bi-objective problem in the area of mechanical engineering, in which the weight (F_1) (includes both the weight of the gears as well as the weight of the shafts) and the stress (F_2) of a speed reducer should be minimized, simultaneously.

There are seven design variables: gear face width (x_1), teeth module (x_2), number of teeth of pinion (x_3 integer variable), distance between bearings 1 (x_4), distance between bearings 2 (x_5), diameter of shaft 1 (x_6), and diameter of shaft 2 (x_7) as well as eleven constraints. The problem can be stated as follows [34, 35]:

$$\text{Minimize}\begin{cases} F_1 = 0.7854x_1x_2^2(\frac{10x_3^2}{3} + 14.933x_3 - 43.0934) \\ -1.508x_1\left(x_6^2 + x_7^2\right) + 7.477x_1\left(x_6^3 + x_7^3\right) \\ +0.7854\left(x_4x_6^2 + x_5x_7^2\right) \\ F_2 = \dfrac{\sqrt{\left(745\dfrac{x_4}{x_2x_3}\right)^2 + 1.69\times10^7}}{0.1x_6^3} \end{cases} \tag{12}$$

$$\text{subject to}\begin{cases} g_1 = \dfrac{1}{x_1x_2^2x_3} - \dfrac{1}{27} \le 0; \\ g_2 = \dfrac{1}{x_1x_2^2x_3^2} - \dfrac{1}{397.5} \le 0; \\ g_3 = \dfrac{x_4^3}{x_2x_3x_6^4} - \dfrac{1}{1.93} \le 0 \\ g_4 = \dfrac{x_5^3}{x_2x_3x_7^4} - \dfrac{1}{1.93} \le 0; \\ g_5 = x_2x_3 - 40 \le 0;\ g_6 = \dfrac{x_1}{x_2} - 12 \le 0 \\ g_7 = 5 - \dfrac{x_1}{x_2} \le 0;\ g_8 = 1.9 - x_4 - 1.5x_6 \le 0; \\ g_9 = 1.9 - x_5 - 1.1x_7 \le 0 \\ g_{10} = \dfrac{\sqrt{\left(745\dfrac{x_4}{x_2x_3}\right)^2 + 1.69\times10^7}}{0.1x_6^3} - 1300 \le 0; \\ g_{11} = \dfrac{\sqrt{\left(745\dfrac{x_5}{x_2x_3}\right)^2 + 1.575\times10^8}}{0.1x_7^3} - 1100 \le 0 \end{cases} \tag{13}$$

$$\text{where}\begin{cases} 2.6 \le x_1 \le 3.6;\ 0.7 \le x_2 \le 0.8; \\ 17 \le x_3 \le 28;\ 7.3 \le x_4 \le 8.3; \\ 7.3 \le x_5 \le 8.3;\ \ 2.9 \le x_6 \le 3.9; \\ 5 \le x_7 \le 5.5 \end{cases} \tag{14}$$

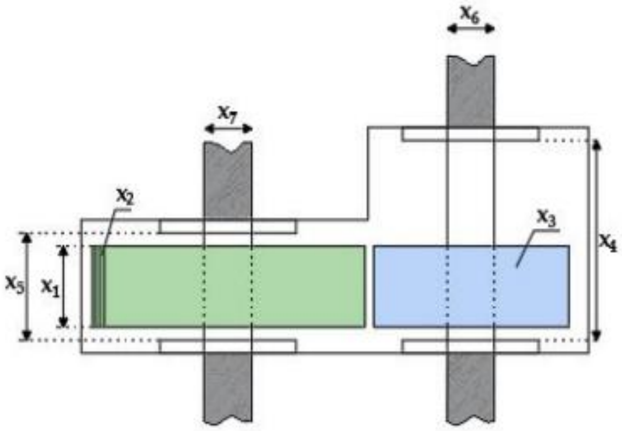


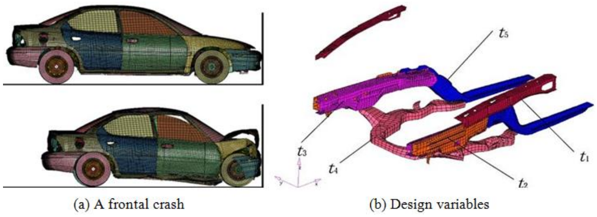
Figure 10. Speed reducer design

5.2 Tri-objective vehicle crashworthiness issue

The vehicle crashworthiness design problem (Figure 11) can be formulated as the structural optimization on the frontal structure of vehicle for crash-worthiness [36]. Thickness of five reinforced members t_i (i=1…5) around the frontal structure are chosen as the design variables, while mass of vehicle f_1, deceleration during the full frontal crash f_2 and toe board intrusion in the offset-frontal crash f_3 are considered as three objectives (Eqs. (15) and (16)). More detailed mathematical formulation can be found in [36]. The optimization problem can be stated as follows [35, 36]:

$$\text{Minimize}\begin{cases} f_1 = 1640.2823 + 2.3573285t_1 + 2.3220035t_2 \\ +4.5688768t_3 + 7.7213633t_4 + 4.45595t_5 \\ f_2 = 6.5856 + 1.15t_1 - 1.0427t_2 + 0.9738t_3 \\ +0.8364t_4 - 0.3695t_1t_4 + 0.0861t_1t_5 + 0.3628t_2t_4 \\ -0.1106t_1^2 - 0.3437t_3^2 + 0.1764t_4^2 \\ f_3 = -0.0551 + 0.0181t_1 + 0.1024t_2 + 0.0421t_3 \\ -0.0073t_1t_2 + 0.024t_2t_3 - 0.0118t_2t_4 - 0.0204t_3t_4 \\ -0.008t_3t_5 - 0.0241t_2^2 + 0.0109t_4^2 \end{cases} \quad (15)$$

$$\text{subject to} \quad 1 \le t_i \left(i = 1..5\right) \le 3 \quad (16)$$



(a) A frontal crash

(b) Design variables

Figure 11. Crashworthiness design

5.3 Results and discussion

Figures 12 and 13 show the Pareto fronts that represent the optimization outcomes of the MICCA algorithm formulti-criteria engineering design challenges.

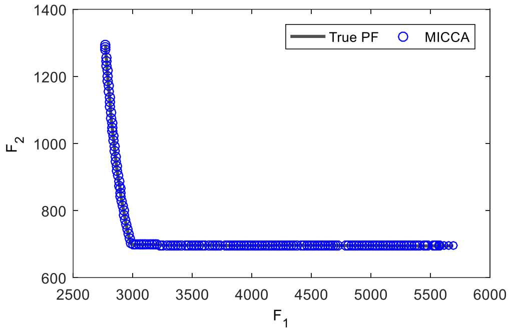


Figure 12. True Pareto Front and MICCA speed reducer design findings

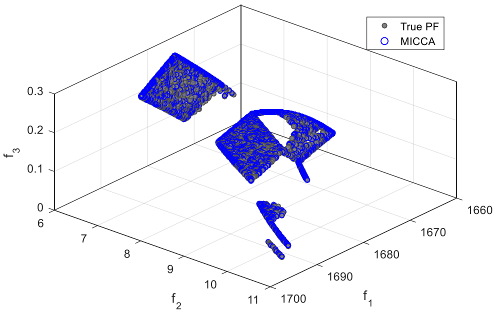


Figure 13. True Pareto Front and MICCA vehicle crashworthiness design findings

Table 7. ϒ, Δ and CT metrics findings

Algorithms	ϒ		Δ		CT	
	Mean	STD	Mean	STD	Mean	STD
Speed Reducer Design						
NSGAII [35]	1.5301E-01	8.43E-03	6.4010E-01	1.65E-02	6.4719	5.27E-01
MOSMA [36]	2.4403E-02	2.26E-02	6.9187E−01	1.19E−02	7.0924E-01	2.41E-02
paε-ODEMO [34]	2.69281	0.24051	0.84041	0.20085	0.42500	0.00671
MOEO [35]	1.5567E-01	1.16E-02	7.2463E-01	1.44E-03	5.7312	4.40E-01
MICCA	7. 0143E-03	1.21E-04	4.1652E-01	1.01E-03	0.3072	0.00095
MPI of MICCA (%)	71.25		34.92		29.12	
Car Crashworthiness						
NSGAII [35]	1.2407E-02	1.73E-03	7.0033E-01	9.47E-02	6.7597	6.49E-01
MOSMA [36]	1.1583E-02	3.68E-04	4.6747E-01	1.31E-01	6.2795E-01	4.27E-03
MOEO [35]	8.6479E-03	1.97E-03	8.2143E-01	1.49E-02	5.9990	3.85E-01
MICCA	8.0031E-04	2.56E-04	6.1205E-02	8.93E-03	0.6197	0.00105
MPI of MICCA (%)	90.73		86.9		1.31	

MPI of MICCA=Minimum performance improvement of MICCA

The non dominated solutions provided by MICCA are near to the true Pareto fronts, as shown in Figures 12 and 13. This demonstrates that the MICCA method shows evidence of convergence to true Pareto-optimal solutions with great diversity.

The results of MICCA will be compared to [34-36]. Table 7 highlights the findings of the MICCA method as well as those found in the literature. Moreover, the minimum performance improvement (MPI) of MICCA results, in terms of (ϒ, Δ and CT) metrics, compared to other literature methods are highlighted in Table 7. One can note that the MICCA has outperformed other methods in all metrics (ϒ, Δ and CT), demonstrating its exploratory skills to find reliable solutions for all multi-criteria engineering design challenges.

In this study, we have analyzed the variance of findings for evaluating MICCA's robustness. In future studies, the influence of setting parameters on algorithm performance will be examined.

6. CONCLUSIONS

In this work, the competitive colonial algorithm (CCA) was expanded to handle multi-objective engineering design challenges. The proposed method is called the multi-objective improved competitive colonial algorithm (MICCA). The original CCA was modified to include the Pareto concept in order to tackle the conflict of the objective functions. Moreover, to ameliorate the quality of non-dominated solutions, the assimilation step of CCA was equipped with a new approach named enhanced assimilation. The suggested algorithm's efficiency was tested on unconstrained, constrained multi-objective issues and multi-criteria engineering design difficulties. Experimental studies were carried out to evaluate the MICCA's performance compared to other well-known methods. Convergence, solution diversity, computational time, and robustness metrics were used to evaluate the performance of the different methods.

These experiments have shown that the MICCA yielded better results compared to those reported in the literature for unconstrained, constrained multi-objective issues and engineering design challenges.

REFERENCES

[1] Li, X., Zhang, J., Yin, M. (2014). Animal migration optimization: An optimization algorithm inspired by animal migration behavior. Neural Computing and Applications, 24: 1867-1877. https://doi.org/10.1007/s00521-013-1433-8

[2] Varadarajan, M., Swarup, K.S. (2008). Solving multi-objective optimal power flow using differential evolution. IET Generation, Transmission & Distribution, 2(5): 720-730. https://doi.org/10.1049/iet-gtd_20070457

[3] Kurz, M.E., Askin, R.G. (2001). An adaptable problem-space-based search method for flexible flow line scheduling. IIE Transactions, 33(8): 691-693. https://doi.org/10.1023/A:1010943502591

[4] Zain, M.Z.B.M., Kanesan, J., Chuah, J.H., Dhanapal, S., Kendall, G. (2018). A multi-objective particle swarm optimization algorithm based on dynamic boundary search for constrained optimization. Applied Soft Computing, 70: 680-700. https://doi.org/10.1016/j.asoc.2018.06.022

[5] Sabeghi, N., Kiani Talaei, B., Ghanbari, R., Ghorbani-Moghadam, K. (2024). Optimizing natural gas liquids (NGL) production process: A multi-objective approach for energy-efficient operations using genetic algorithm and artificial neural networks. Iranian Journal of Numerical Analysis and Optimization, 14(2): 522-544. https://doi.org/10.22067/ijnao.2024.83955.1303

[6] Deb, K., Pratap, A., Moitra, S. (2000). Mechanical component design for multiple ojectives using elitist non-dominated sorting ga. In International conference on parallel problem solving from nature, pp. 859-868. https://doi.org/10.1007/3-540-45356-3_84

[7] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation, 6(2): 182-197. https://doi.org/10.1109/4235.996017

[8] Gupta, R., Khan, D. (2022). Heuristic solutions for interval-valued games. Iranian Journal of Numerical Analysis and Optimization, 12(1): 187-200. https://doi.org/10.22067/ijnao.2021.71321.1048

[9] 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. https://doi.org/10.1109/CEC.2007.4425083

[10] Ahmadi, M.A., Ebadi, M., Shokrollahi, A., Majidi, S. M.J. (2013). Evolving artificial neural network and imperialist competitive algorithm for prediction oil flow rate of the reservoir. Applied Soft Computing, 13(2): 1085-1098. https://doi.org/10.1016/j.asoc.2012.10.009

[11] Ghanavati, M., Gholamian, M.R., Minaei, B., Davoudi, M. (2011). An efficient cost function for imperialist competitive algorithm to find best clusters. Journal of Theoretical & Applied Information Technology, 29(1): 22-31.

[12] Chen, J., Zhang, Y., Wang, G. (2011). A new algorithm for a fuzzy vehicle routing and scheduling problem: Imperialist competitive algorithm. JCIT: Journal of Convergence Information Technology, 6(7): 303-311.

[13] Mohamed, N., Alateyah, A.I., El-Garaihy, W.H. (2021). Defect free optimization of a polycentric prosthetic knee design using imperialist competition-inspired optimization method. Journal of Engineering Research, 11(2B). https://doi.org/10.36909/jer.13063

[14] Muruganantham, P., Balaji, D. (2021). Optimal forecasting of thermal conductivity and storage in parabolic trough collector using phase change materials. Journal of Engineering Research, 9(2): 7706. https://doi.org/10.36909/jer.v9i2.7706

[15] Toghyani, S., Ahmadi, M.H., Kasaeian, A., Mohammadi, A.H. (2016). Artificial neural network, ANN-PSO and ANN-ICA for modelling the Stirling engine. International Journal of Ambient Energy, 37(5): 456-468. https://doi.org/10.1080/01430750.2014.986289

[16] Ardalan, Z., Karimi, S., Poursabzi, O., Naderi, B. (2015). A novel imperialist competitive algorithm for generalized traveling salesman problems. Applied Soft Computing, 26: 546-555. https://doi.org/10.1016/j.asoc.2014.08.033

[17] Nemati, K., Shamsuddin, M., Kamarposhti, M.S. (2011). Using imperial competitive algorithm for solving traveling salesman problem and comparing the efficiency

of the proposed algorithm with methods in use. Australian Journal of Basic and Applied Sciences, 5: 540-543.

[18] Bilel, N., Mohamed, N., Zouhaier, A., Lotfi, R. (2019). A novel approach for robust design of sewing machine. In Advances in Mechanical Engineering and Mechanics: Selected Papers from the 4th Tunisian Congress on Mechanics, CoTuMe 2018, Hammamet, Tunisia, pp. 112-119. https://doi.org/10.1007/978-3-030-19781-0_14

[19] Najlawi, B., Nejlaoui, M., Affi, Z., Romdhane, L. (2019). Multi-objective robust design optimization of a sewing mechanism under uncertainties. Journal of Intelligent Manufacturing, 30: 783-794. https://doi.org/10.1007/s10845-016-1284-0

[20] Abdi, B., Mozafari, H., Ayob, A., Kohandel, R. (2011). Imperialist competitive algorithm and its application in optimization of laminated composite structures. European Journal of Scientific Research, 55(2): 174-187.

[21] Ghaderi, M.R., Aghakhani, M., Eslampanah, A., Ghaderi, K. (2015). The application of imperialist competitive algorithm for optimization of deposition rate in submerged arc welding process using TiO_2 nano particle. Journal of Mechanical Science and Technology, 29: 357-364. https://doi.org/10.1007/s12206-014-1242-8

[22] Hosseini, S., Al Khaled, A. (2014). A survey on the imperialist competitive algorithm metaheuristic: Implementation in engineering domain and directions for future research. Applied Soft Computing, 24: 1078-1094. https://doi.org/10.1016/j.asoc.2014.08.024

[23] Huang, L., Duan, H., Wang, Y. (2014). Hybrid bio-inspired lateral inhibition and imperialist competitive algorithm for complicated image matching. Optik, 125(1): 414-418. https://doi.org/10.1016/j.ijleo.2013.06.085

[24] Ghasemi, M., Ghavidel, S., Ghanbarian, M.M., Gitizadeh, M. (2015). Multi-objective optimal electric power planning in the power system using Gaussian bare-bones imperialist competitive algorithm. Information Sciences, 294: 286-304. https://doi.org/10.1016/j.ins.2014.09.051

[25] Ghasemi, M., Ghavidel, S., Rahmani, S., Roosta, A., Falah, H. (2014). A novel hybrid algorithm of imperialist competitive algorithm and teaching learning algorithm for optimal power flow problem with non-smooth cost functions. Engineering Applications of Artificial Intelligence, 29: 54-69. https://doi.org/10.1016/j.engappai.2013.11.003

[26] Ghadi, M.J., Baghramian, A., Imani, M.H. (2016). An ICA based approach for solving profit based unit commitment problem market. Applied Soft Computing, 38: 487-500. https://doi.org/10.1016/j.asoc.2015.10.026

[27] Armaghani, D.J., Mohamad, E.T., Narayanasamy, M.S., Narita, N., Yagiz, S. (2017). Development of hybrid intelligent models for predicting TBM penetration rate in hard rock condition. Tunnelling and Underground Space Technology, 63: 29-43. https://doi.org/10.1016/j.tust.2016.12.009

[28] Bilel, N., Mohamed, N., Zouhaier, A., Lotfi, R. (2016). An improved imperialist competitive algorithm for multi-objective optimization. Engineering Optimization, 48(11): 1823-1844. https://doi.org/10.1080/0305215X.2016.1141204

[29] Bilel, N., Mohamed, N., Zouhaier, A., Lotfi, R. (2019). An efficient evolutionary algorithm for engineering design problems. Soft Computing, 23: 6197-6213. https://doi.org/10.1007/s00500-018-3273-z

[30] Jangir, P., Jangir, N. (2018). A new non-dominated sorting grey wolf optimizer (NS-GWO) algorithm: Development and application to solve engineering designs and economic constrained emission dispatch problem with integration of wind power. Engineering Applications of Artificial Intelligence, 72: 449-467. https://doi.org/10.1016/j.engappai.2018.04.018

[31] Panda, A., Pani, S. (2016). A symbiotic organisms search algorithm with adaptive penalty function to solve multi-objective constrained optimization problems. Applied Soft Computing, 46: 344-360. https://doi.org/10.1016/j.asoc.2016.04.030

[32] Babalik, A., Ozkis, A., Uymaz, S.A., Kiran, M.S. (2018). A multi-objective artificial algae algorithm. Applied Soft Computing, 68: 377-395. https://doi.org/10.1016/j.asoc.2018.04.009

[33] Huy, T.H.B., Nallagownden, P., Truong, K.H., Kannan, R., Vo, D.N., Ho, N. (2022). Multi-objective search group algorithm for engineering design problems. Applied Soft Computing, 126: 109287. https://doi.org/10.1016/j.asoc.2022.109287

[34] Gong, W., Cai, Z., Zhu, L. (2009). An efficient multiobjective differential evolution algorithm for engineering design. Structural and Multidisciplinary Optimization, 38: 137-157. https://doi.org/10.1007/s00158-008-0269-9

[35] Premkumar, M., Jangir, P., Sowmya, R., Alhelou, H.H., Mirjalili, S., Kumar, B.S. (2022). Multi-objective equilibrium optimizer: Framework and development for solving multi-objective optimization problems. Journal of Computational Design and Engineering, 9(1): 24-50. https://doi.org/10.1093/jcde/qwab065

[36] Premkumar, M., Jangir, P., Sowmya, R., Alhelou, H.H., Heidari, A.A., Chen, H. (2020). MOSMA: Multi-objective slime mould algorithm based on elitist non-dominated sorting. IEEE Access, 9: 3229-3248. https://doi.org/10.1109/ACCESS.2020.3047936