Research article

Multi-objective liver cancer algorithm: A novel algorithm for solving engineering design problems

Kanak Kalita [a,f,*], Janjhyam Venkata Naga Ramesh [b], Robert Čep [c], Sundaram B. Pandya [d], Pradeep Jangir [e], Laith Abualigah [g,h,i,j,k,l,m,n]

[a] Department of Mechanical Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, 600 062, India
[b] Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, 522502, India
[c] Department of Machining, Assembly and Engineering Metrology, Faculty of Mechanical Engineering, VSB-Technical University of Ostrava, 70800, Ostrava, Czech Republic
[d] Department of Electrical Engineering, Shri K.J. Polytechnic, Bharuch, 392 001, India
[e] Department of Biosciences, Saveetha School of Engineering. Saveetha Institute of Medical and Technical Sciences, Chennai, 602 105, India
[f] University Centre for Research & Development, Chandigarh University, Mohali, 140413, India
[g] Computer Science Department, Al al-Bayt University, Mafraq, 25113, Jordan
[h] Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman, 19328, Jordan
[i] MEU Research Unit, Middle East University, Amman, 11831, Jordan
[j] Department of Electrical and Computer Engineering, Lebanese American University, Byblos, 13-5053, Lebanon
[k] School of Computer Sciences, Universiti Sains Malaysia, Pulau, Pinang, 11800, Malaysia
[l] School of Engineering and Technology, Sunway University Malaysia, Petaling Jaya, 27500, Malaysia
[m] Applied Science Research Center, Applied Science Private University, Amman, 11931, Jordan
[n] Artificial Intelligence and Sensing Technologies (AIST) Research Center, University of Tabuk, Tabuk, 71491, Saudi Arabia

ARTICLE INFO

ABSTRACT

This research introduces the Multi-Objective Liver Cancer Algorithm (MOLCA), a novel approach inspired by the growth and proliferation patterns of liver tumors. MOLCA emulates the evolutionary tendencies of liver tumors, leveraging their expansion dynamics as a model for solving multi-objective optimization problems in engineering design. The algorithm uniquely combines genetic operators with the Random Opposition-Based Learning (ROBL) strategy, optimizing both local and global search capabilities. Further enhancement is achieved through the integration of elitist non-dominated sorting (NDS), information feedback mechanism (IFM) and Crowding Distance (CD) selection method, which collectively aim to efficiently identify the Pareto optimal front. The performance of MOLCA is rigorously assessed using a comprehensive set of standard multi-objective test benchmarks, including ZDT, DTLZ and various Constraint (CONSTR, TNK, SRN, BNH, OSY and KITA) and real-world engineering design problems like Brushless DC wheel motor, Safety isolating transformer, Helical spring, Two-bar truss and Welded beam. Its efficacy is benchmarked against prominent algorithms such as the non-dominated sorting grey wolf optimizer (NSGWO), multiobjective multi-verse optimization (MOMVO), non-dominated sorting genetic algorithm (NSGA-II), decomposition-based multiobjective evolutionary algorithm (MOEA/D) and multiobjective marine predator algorithm (MOMPA). Quantitative analysis is conducted using GD, IGD, SP, SD, HV and RT metrics to represent convergence and distribution, while

* Corresponding author. Department of Mechanical Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, 600 062, India.

E-mail addresses: drkanakkalita@veltech.edu.in, kanakkalita02@gmail.com (K. Kalita), jvnramesh@gmail.com (J.V. Naga Ramesh), robert.cep@vsb.cz (R. Čep), sundarampandya@gmail.com (S.B. Pandya), pkjmtech@gmail.com (P. Jangir), aligah.2020@gmail.com (L. Abualigah).

qualitative aspects are presented through graphical representations of the Pareto fronts. The MOLCA source code is available at: https://github.com/kanak02/MOLCA.

1. Introduction

Long before human civilization, nature had been adeptly addressing intricate issues through evolutionary processes. It is only logical to seek inspiration from nature when confronted with complex problems. In the realm of optimization, 1977 marked a pivotal moment when Holland introduced the idea of emulating natural evolutionary principles on computers to address optimization challenges [1]. This gave birth to the renowned heuristic method known as Genetic Algorithms (GA) [2], offering a fresh perspective on addressing multifaceted problems across various disciplines. The GA's core principle is straightforward, emulating natural processes like selection, gene recombination and mutation. Darwin's evolutionary theory was the cornerstone for this algorithm. In GA, the optimization journey commences with a random assortment of potential solutions for a specified problem. Each problem variable is perceived as a gene, with the collective set likened to chromosomes. A cost function determines each chromosome's viability, with the entire solution set viewed as a population. The most viable chromosomes are then randomly chosen to form the subsequent population, mirroring natural processes. This is followed by the fusion of selected individuals, where genes from pairs are randomly amalgamated to spawn new entities. Some genes then undergo random alterations, simulating mutation. Over time, GA emerged as a preferred optimization method over deterministic strategies [3], primarily due to its ability to bypass local solutions [4]. One limitation of GA was its inherent randomness, leading to varied solutions upon each execution. While multiple runs could potentially address this, the extensive function evaluations per run remain a concern. GA's widespread applicability across scientific and industrial sectors has catalyzed the development of other heuristic strategies. Notable algorithms like Ant Colony Optimization (ACO) [5], Particle Swarm Optimizer (PSO) [6] and Differential Evolution algorithm (DE) [7] are testament to GA's success. This domain continues to be a focal point in the computational intelligence arena. Beyond the aforementioned merit of bypassing local optima, several other factors underscore the growing appeal of heuristic methods. For instance, they have used in diverse fields ranging from robotics [8] to vehicle testing [9] to topology optimization [10]. They also find ample application in next-generation applications like digital twining [11], electric vehicles [12], blockchains [13] etc.

Heuristic algorithms operate without the necessity to compute the derivation of a problem for its resolution. This is attributed to their design, which does not rely on gradient descent to pinpoint the global optimum. Instead, they perceive a problem as a closed system characterized by specific inputs and outputs. The inputs correspond to the problem's variables, while the outputs relate to the objectives. The heuristic exploration initiates by generating a random set of inputs, representing potential solutions. This exploration progresses by assessing each solution, monitoring the objective values and subsequently adapting, merging, or evolving the solutions based on their results. This cycle persists until a predefined termination criterion is achieved, which could be a set iteration limit or a maximum count of function evaluations. While heuristics demonstrate proficiency in addressing intricate real-world challenges, they encounter numerous obstacles in optimization tasks [14]. Moreover, optimization challenges are diverse, each with unique traits. Some complexities or features include dynamicity [15], unpredictability [16], constraints [17], multiple objectives [18] and extensive objectives [19]. Each of these challenges has carved out its niche within the heuristic domain, drawing significant research interest. In dynamic scenarios, the global optimum's position is mutable. Hence, a heuristic must incorporate appropriate mechanisms to monitor these shifts without losing sight of the global optimum [20]. Real-world challenges often present varying degrees of uncertainty across components. Addressing these demands heuristics that can identify resilient solutions with a high tolerance for errors. Constraints, which delineate the search domain, further complicate matters. They categorize solutions into feasible and non-feasible groups. Consequently, heuristics must be equipped with mechanisms to filter out non-feasible solutions during the optimization process, ultimately zeroing in on the optimal feasible solution. Numerous effective techniques for handling constraints exist in scholarly literature [21]. However, as this study's proposed method pertains exclusively to unconstrained challenges, a detailed review of such techniques is omitted, directing interested readers to Refs. [18–21]. Numerous effective techniques for handling constraints exist in scholarly literature. For instance, the Dynamic Levy Flight Chimp Optimization technique discussed [22] and the Weighted Chimp Optimization Algorithm by Ref. [23] are notable examples. These methods demonstrate advanced approaches in the categorization of feasible solutions. Additionally, the Multi-objective Chimp Optimizer proposed [24] offers insights into handling multiple objectives, which can be pertinent in distinguishing between feasible and non-feasible solutions. While this study's proposed method pertains exclusively to unconstrained challenges and thus a detailed review of constraint-handling techniques is omitted as well as earlier works [25–27], which further elaborate on the evolution and applications of chimp optimization algorithms in various contexts.

Among the aforementioned challenges, managing multiple objectives stands out. When multiple objectives are in play, heuristic algorithms grapple with solution comparisons. To navigate this, researchers have employed the PD operator [28]. Given the nature of such challenges, there often exists more than a single optimal solution for a multi-objective problem [29]. Many heuristics in scholarly literature are equipped with specialized operators tailored for multi-objective challenges. The foundational structure of these multi-objective heuristics is strikingly similar. At their core lies an archive or storage system dedicated to preserving the NDS throughout the optimization journey [30]. As the process unfolds, these heuristics consistently refine this archive, enhancing both the caliber and volume of the non-dominated solutions it contains [31] a post-optimization activity. Researchers have employed the PD operator, as exemplified in the work of multi-objective mutation-based dynamic Harris Hawks optimization for IoT botnet detection [32]. Given the nature of such challenges, there often exists more than a single optimal solution for a multi-objective problem, as demonstrated in study on improved African vulture's optimization algorithms for image segmentation [33]. Many heuristics in

scholarly literature are equipped with specialized operators tailored for multi-objective challenges. The foundational structure of these multi-objective heuristics is strikingly similar. At their core lies an archive or storage system dedicated to preserving the NDS throughout the optimization journey, a concept explored in the context of community detection in social networks [34]. As the process unfolds, these heuristics consistently refine this archive, enhancing both the caliber and volume of the non-dominated solutions it contains. This is evident in the comprehensive survey of slime mould algorithm variants and applications, which highlights the importance of such archives in multi-objective optimization [35]. Finally, the post-optimization activity is crucial, as seen in the work of Piri et al. (2022) on using artificial gorilla troop optimization for feature selection in biomedical data, including COVID-19 data [36].

The academic realm is replete with algorithms tailored for multi-objective challenges [37–39]. In the context of Genetic Algorithms (GA), the NSGA [40] stands out as a prominent multi-objective (MO) variant. Other noteworthy algorithms encompass MOPSO [41], MOACO [42], MO Differential Evolution [43] and Multi-objective Evolution Strategy [44]. These algorithms have consistently demonstrated their efficacy in identifying non-dominated solutions for multi-objective dilemmas. Yet, a pertinent question arises: is there a need for additional algorithms? The No Free Lunch theorem for optimization [45] posits that no singular algorithm can universally address all optimization challenges. This theorem, grounded in logic, paves the way for the introduction of novel algorithms or the enhancement of existing ones. Consequently, there remains ample scope for the development or refinement of algorithms to adeptly address contemporary challenges and those that remain elusive to current methodologies.

In metaheuristic approaches three primary strategies have been identified for addressing multi-objective problems using metaheuristics: a priori [46], a posteriori [47] and interactive [48]. The a priori method consolidates multiple objectives into a singular objective. This transformation essentially reframes the multi-objective challenge into a single-objective one. By aggregating objectives, single-objective optimizers can adeptly pinpoint Pareto optimal solutions. Yet, this approach is not without its limitations. It necessitates multiple algorithm executions to discern various Pareto optimal solutions, confronts challenges in each run, lacks inter-solution communication during optimization, demands expert consultation for optimal weight determination and struggles with identifying concave regions of the POF due to objective amalgamation. As the name suggests, decision-making in the a priori approach precedes optimization, primarily during weight determination. The inherent challenges of this method often overshadow its advantages. Typically, designers employing this strategy must execute the algorithm repeatedly, adjusting weights to delineate the Pareto optimal front. Conversely, the a posteriori approach defers decision-making until post-optimization. This method abstains from aggregation, preserving the multi-objective nature of the problem. Its strengths lie in its capacity to identify the Pareto optimal solution (POS) set in a singular run, facilitate information exchange between POS during optimization and ascertain any type of Pareto optimal front. However, posteriori techniques demand specialized mechanisms to manage multiple, often divergent, objectives. Moreover, their computational demands usually surpass those of aggregation-based methods. Lastly, the interactive approach to multi-objective optimization implies concurrent decision-making during the optimization process. Often termed "human-in-the-loop" optimization, this method continually integrates expert preferences throughout the optimization journey, aiming to pinpoint the most suitable Pareto optimal solutions.

Existing literature underscores the dominance of a posteriori methods in multi-objective optimization. Many esteemed algorithms, originally designed for single-objective, have been adapted to cater to a posteriori MOO. It is worth noting that these algorithms predominantly rely on Pareto dominance for solution comparisons and utilize an archive to retain the top-tier POS discovered. The foundational blueprint of all a posteriori methodology remains consistent. They embark on the optimization journey with a collection of random solutions. Once the POS are identified and archived, efforts are channeled towards refining these solutions to unearth superior Pareto optimal outcomes. This refinement phase concludes upon meeting a specific criterion. The primary goal of an a posteriori MO is to derive an exceptionally precise approximation of the genuine POS for a designated MO challenge. Given that decision-making transpires post-optimization, It is imperative for solutions to be uniformly distributed across all objectives. A significant hurdle in this domain is the inherent tension between achieving precise Pareto optimal solutions (convergence) and ensuring solution distribution (coverage). An adept multi-objective optimization should strike a harmonious balance between these contrasting elements. To enhance coverage, various strategies are employed. For instance, in MOPSO, Pareto optimal solutions in sparsely populated archive segments are more likely to be selected as leaders for other particles. Meanwhile, NSGA-II employs non-dominated sorting to rank Pareto optimal solutions, assigning them a specific value. This ranking system amplifies the likelihood of superior Pareto optimal solutions contributing to the genesis of the subsequent generation.

Other popular Multi-Objective (MO) Algorithms include MO ant lion optimizer (MOALO) [49], MO equilibrium optimizer (MOEO) [50], MO slime mould algorithm (MOSMA) [51], MO arithmetic optimization algorithm (MOAOA) [52], non-dominated sorting ions motion algorithm (NSIMO) [53], MO evolutionary algorithm based on decomposition (MOEA/D) [54], Non-dominated sorting genetic algorithm (NSGA-II) [55], multi-objective multi-verse optimization (MOMVO) [56], non-dominated sorting grey wolf optimizer (NS-GWO) [57], MO Gradient-Based Optimizer (MOGBO) [58], MO plasma generation optimizer (MOPGO) [59], non-dominated sorting Harris hawks optimization (NSHHO) [60], MO thermal exchange optimization (MOTEO) [61], decomposition based multi-objective heat transfer search (MOHTS/D) [62], Decomposition-Based Multi-Objective Symbiotic Organism Search (MOSOS/D) [63], MOGNDO Algorithm [64], Non-dominated sorting moth flame optimizer (NSMFO) [65], Non-dominated sorting whale optimization algorithm (NSWOA) [66] and Non-Dominated Sorting Dragonfly Algorithm (NSDA) [67]. We aimed to gauge their capabilities in swiftly converging to the true Pareto optimal front and the distribution of the obtained non-dominated solutions. Upon assessing their convergence and coverage using MO metrics and the produced Pareto optimal fronts on benchmark suites (ZDT [54], DTLZ [54], Constraint [68,69] and engineering design problems [55,56]), we discerned that these algorithms still exhibited shortcomings in convergence and coverage using metrics like generational distance (GD) [70], inverse generational distance (IGD) [71], hypervolume [72], Spacing [73], Spread [72] and run time (RT). The approximations of the Pareto-front produced by our method are

evaluated using these metrics. The field of optimization encompasses various challenges like dynamicity, unpredictability, constraints and multiple objectives [74]. MOLCA is designed to handle these complexities more effectively, particularly in dynamic and unpredictable environments. Traditional multi-objective algorithms often struggle to maintain a balance between converging to optimal solutions and preserving diversity among solutions. MOLCA introduces novel mechanisms like the Information Feedback Mechanism (IFM) and Crowding Distance, which collectively enhance this balance. The MOLCA algorithm divides the search process into distinct phases – an initial phase focusing on convergence and a subsequent phase emphasizing diversity. This structured approach allows for a more efficient search process.

This paper presents several key findings and contributions:

- A new Multi-objective Liver Cancer Algorithm (MOLCA) proposed.
- An Information Feedback Mechanism (IFM) is introduced to balance convergence and diversity. This strategy divides the search process into two phases: the initial phase emphasizes convergence, while the subsequent phase emphasizes diversity.
- The rapid non-dominated sorting technique determines the current phase of the algorithm. If the non-dominated solutions make up less than 90% of all solutions, the algorithm is in the initial phase; otherwise, it transitions to the subsequent phase.
- We used Crowding Distance mechanism to refine the positioning of solutions in the space. This enhances solution diversity and ensures a more uniform distribution along the Pareto front.
- The crowding distance metric is utilized to preserve population diversity. Notably, the elitism strategy is employed.

Despite the progressive strides in multi-objective optimization, the research community remains fervently committed to refining existing methodologies or pioneering novel ones to more effectively address current multi-objective optimization challenges. The emulation of liver tumor growth patterns in the context of multi-objective optimization Firstly, the complexity of liver tumors, such as hepatocellular carcinoma, is a significant challenge in biomedicine. These tumors exhibit complex behaviors, including interactions with the microenvironment, angiogenesis, immune evasion and genetic changes, all of which are crucial factors in their progression. By studying these mechanisms, insights can be gained into developing optimization algorithms that mimic these adaptive and robust biological processes. The Liver Cancer Algorithm (LCA) [75], inspired by the behavior of liver tumors, serves as an innovative approach in optimization. It emulates the growth and spread of liver tumors, utilizing this biological model to address optimization problems effectively. The LCA employs an evolutionary search approach, simulating tumor behavior and incorporates genetic operators and a Random Opposition-Based Learning (ROBL) strategy. These elements enable the LCA to balance exploration and exploitation, which is crucial in finding optimal solutions in complex optimization landscapes. This algorithm's unique adaptive mechanism seamlessly harmonizes exploration and exploitation. Driven by the notable advantages of LCA, this paper introduces its multi-objective version, MOLCA. MOLCA employs a rapid non-dominated sorting technique to determine the current phase of the algorithm. This allows for a dynamic adjustment between focusing on convergence and promoting diversity, based on the composition of the solution set. IFM in MOLCA ensures a dynamic response to the changing landscape of the problem space. By adjusting the focus between convergence and diversity, MOLCA can effectively navigate the complex multi-objective optimization terrain. This mechanism refines the positioning of solutions, enhancing the diversity and ensuring a more uniform distribution along the Pareto front. This is crucial for achieving a comprehensive exploration of the solution space. MOLCA incorporates an elitism strategy, which helps in preserving the quality of solutions while maintaining population diversity. This is critical for ensuring that the algorithm does not converge prematurely to suboptimal solutions. This is benchmarked against NSGWO, MOMVO, NSGA-II, MEA/D and MOMPA algorithms.

The subsequent sections of this paper are structured as follows: Section 2 offers an in-depth exploration of the LCA. Section 3 present proposed multi-objective LCA algorithm. Section 4 presents, scrutinizes and interprets the results, encompassing the experimental framework, test functions and performance metrics. Lastly, Section 5 encapsulates the study's conclusions and outlines potential avenues for future research.

2. Liver cancer algorithm (LCA)

The Liver Cancer Algorithm (LCA) [75] emerges as a groundbreaking bio-inspired optimization tool proposed by Essam H. Houssein et all, with its roots in the intricate behaviors exhibited by liver tumors, specifically hepatocellular carcinoma. This form of liver cancer, marked by its complexity, presents numerous challenges in both detection and management. The expansion trajectory of these tumors is influenced by a myriad of factors, such as their position, magnitude and inherent virulence. By translating these tumor behaviors into optimization paradigms, the LCA mirrors the tumor's adaptive spread and growth within the liver's confines. The algorithm's architecture is shaped by the tumor's adaptive tendencies, where it seeks optimal growth conditions within the liver. Incorporating both genetic operators and the Random Opposition-Based Learning (ROBL) strategy, the LCA achieves a fine equilibrium between localized and expansive search methodologies, facilitating a comprehensive search space exploration. The algorithm's fitness function evaluates candidate solutions across various tumor stages. In instances where a candidate solution exhibits suboptimal fitness, the LCA permits an alternate tumor to recalibrate the position based on the optimal tumor size. Empirical results underscore the LCA's adeptness in balancing expansive exploration with focused exploitation, culminating in efficient optimization solutions. This section describes the LCA algorithmic structure and offering an in-depth mathematical breakdown of its various phases. Drawing inspiration from the dynamics of liver tumors, the LCA incorporates biological concepts into its optimization mechanism, positioning it as a unique and effective method for feature selection.

2.1. Integration of genetic operators in LCA

2.1.1. Role of genetic operators

In the LCA, genetic operators are critical for maintaining diversity among candidate solutions and preventing premature convergence. This is analogous to biological evolution, where genetic variation is key to the adaptability and survival of species.

2.1.2. Implementation of genetic operators

2.1.2.1. Selection. The LCA selects the fittest individuals from the current population to form a mating pool. This selection is based on a fitness function that evaluates how well each solution performs in solving the optimization problem.

2.1.2.2. Crossover. Crossover operators are used to combine genetic information from two parent solutions to produce new offspring. This mimics natural gene recombination and contributes to the exploration of new areas in the search space.
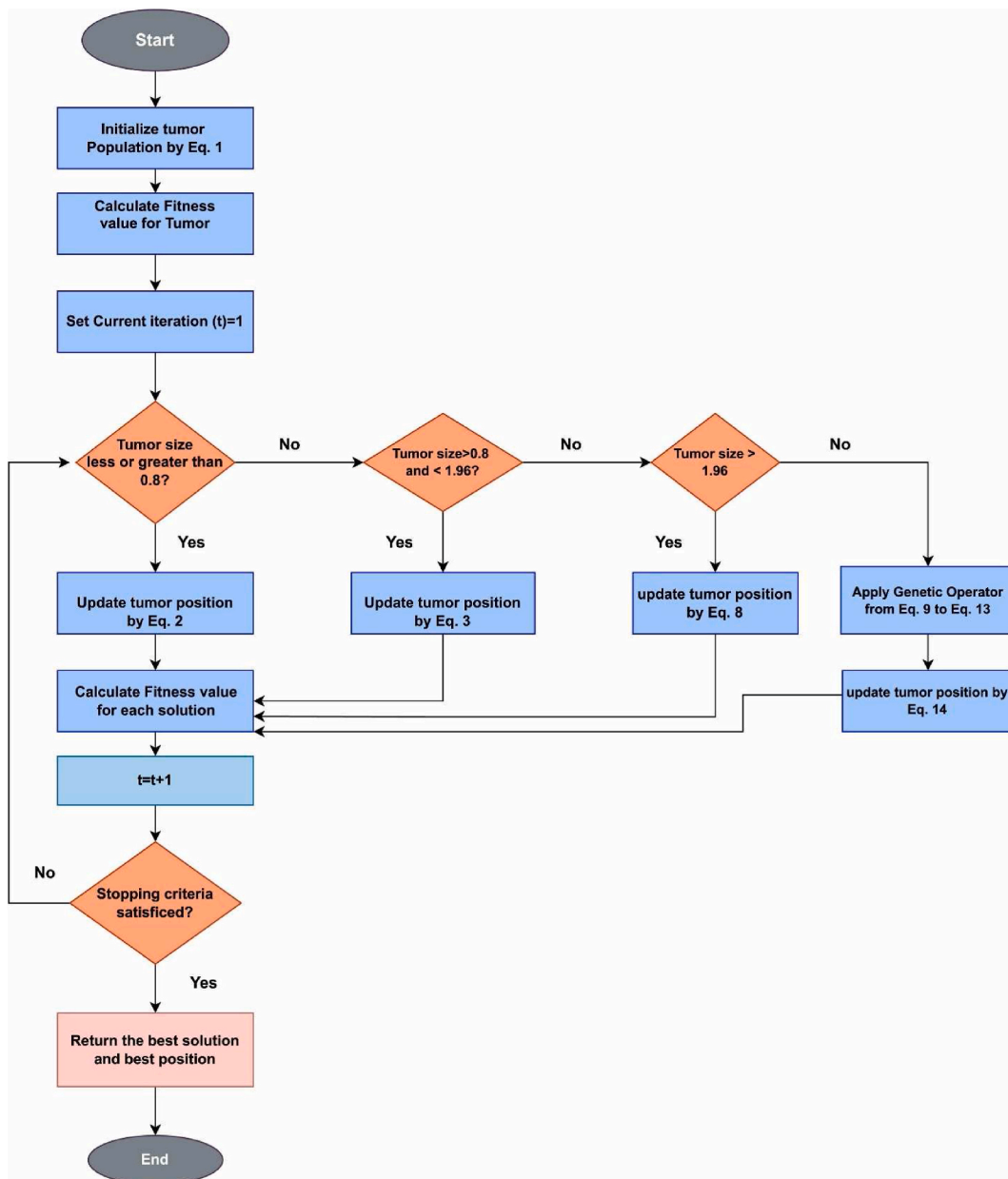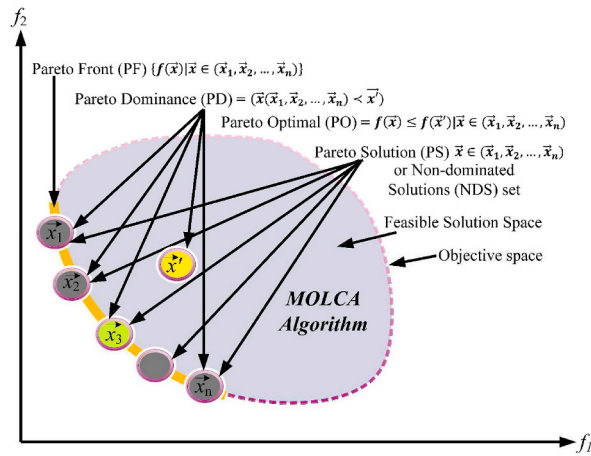

Fig. 1. LCA flowchart.

Fig. 2. Multi-Objective all definitions in search space of MO-Problem.

2.1.2.3. Mutation. Mutation operators introduce random changes in the offspring's genetic makeup. This process ensures diversity in the population and helps in exploring unvisited regions of the search space, thereby avoiding local optima.

2.1.3. Adaptive nature

The algorithm's design, inspired by the adaptive nature of liver tumors, utilizes these genetic operators to adaptively find the most favorable solutions in the optimization landscape, similar to how tumors adaptively find favorable environments for growth.

2.2. Integration of random opposition-based learning (ROBL) in LCA

2.2.1. Concept of ROBL

ROBL is an innovative approach for initializing and diversifying the population in optimization algorithms. It involves generating solutions that are opposite to the existing ones in the search space.

2.2.2. Implementation in LCA

2.2.2.1. Initialization phase. During the initial population setup, alongside generating random solutions, the LCA algorithm also creates opposite solutions. This is achieved by calculating the opposite of each individual's position in the search space.

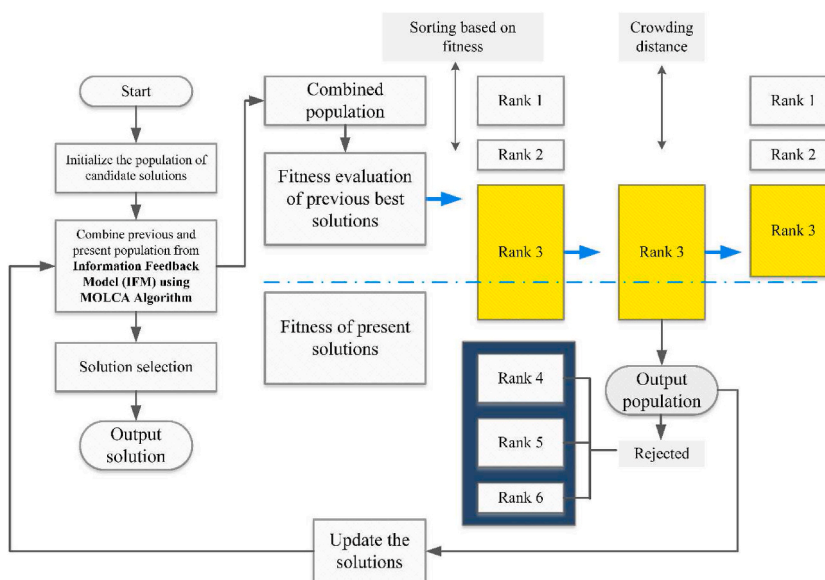**Fig. 3.** The procedure of the NDS approach based on MOLCA algorithm.

2.2.2.2. Diversification. By incorporating these opposite solutions into the population, the LCA algorithm enhances the diversity of the search space explored. This approach is particularly effective in avoiding early convergence to local optima and helps in a more comprehensive exploration of the search space.
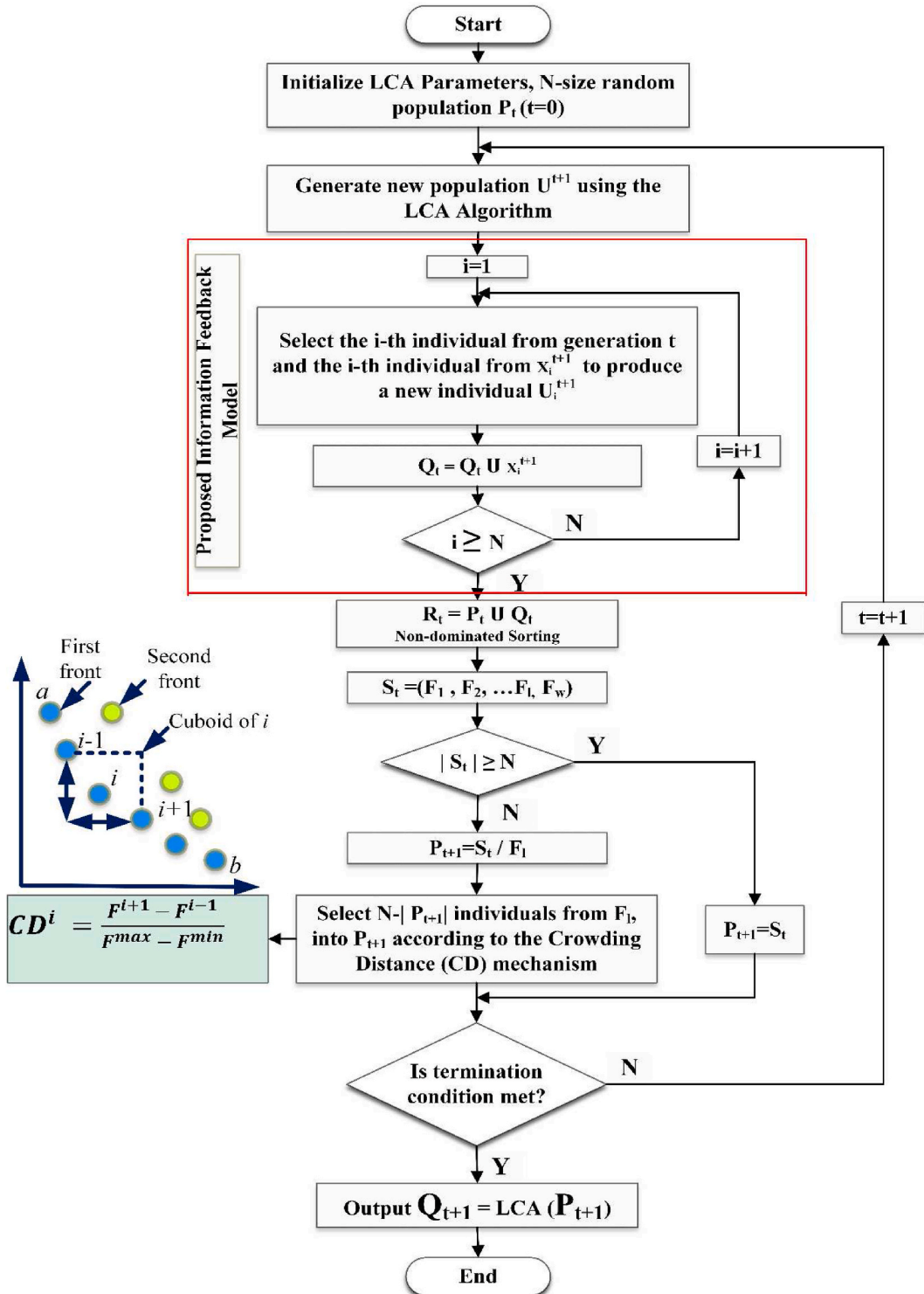

Fig. 4. Flowchart of MOLCA algorithm.

2.2.3. Balancing exploration and exploitation

The integration of ROBL in LCA facilitates a balanced trade-off between exploration (searching new areas) and exploitation (refining known good areas). This balance is crucial for effectively navigating complex optimization landscapes and finding high-quality solutions.

2.2.4. Dynamic adaptation

The LCA continuously applies ROBL throughout the optimization process, dynamically adapting its search strategy based on the evolving population. This ensures that the algorithm remains robust against stagnation and is capable of discovering optimal or near-optimal solutions across diverse problem domains.

2.3. Mathematical representation of the liver cancer algorithm (LCA)

The LCA is conceptualized to simulate the expansion and proliferation patterns of liver tumors, which are malignant growths in the
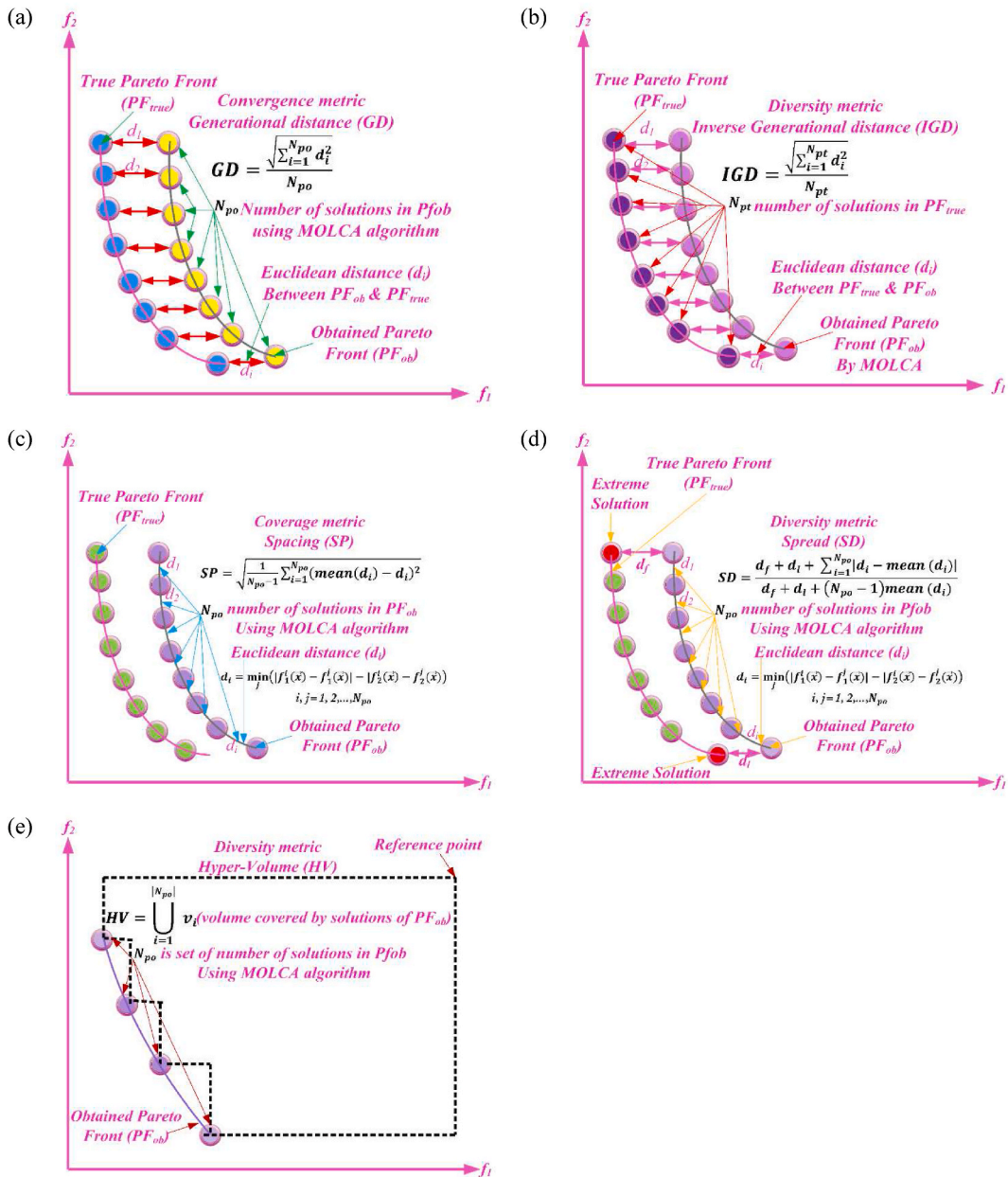
Fig. 5. Mathematical and Schematic view of the (a) GD, (b) IGD, (c) SP, (d) SD and (e) HV metrics.

liver that can significantly compromise bodily functions. To emulate the tumor's dynamics, the LCA is structured into multiple phases, each characterized by specific mathematical models.

2.3.1. Estimation of tumor dimensions

Central to the LCA is the assessment of the tumor's dimensions, which is pivotal for the phases that follow. The tumor's size is gauged using a mathematical framework that presumes tumors adopt a hemi-ellipsoid form, a characteristic seen in various cancer forms, including liver cancer. The tumor is delineated by three parameters: width, length and depth. Given the complexity of directly measuring the depth, we resort to a mathematical approximation. The initial estimation of the tumor's size (or position) utilizes Eq. (1), which integrates random opposition-based learning (ROBL) to initiate a population with a broad spectrum of search exploration potentials.

$$\text{Position}_i^{*j} = \frac{\pi}{6} \left(\text{length}^j \right) \cdot \left(\text{width}^j \right) \cdot \left(\text{height}^j \right) - \left(lb + (ub - lb) - r_d * \text{Position}_i^j \right) \tag{1}$$

In this context $i = 1, 2, …, N, j = 1, 2, …, D. lb$ and ub signify the maximum and minimum limits of the decision variables, D represents the search area's dimension and r_d is a random value between [0, 1] influenced by the optimal objective of the current Position_i and its opposite Position_i^j, thus setting the initial population. The tumor's position size is deduced from the dimensions of the tumor (width, depth and length); where width, depth and length are random values between [0,1]. This study mathematically portrays the growth in the liver tumor's position to craft the LCA and facilitate optimization. The growth in the tumor's position size is determined using Eq. (2),

$$\text{Position} = \frac{\pi}{6} \cdot f \cdot (l \cdot w)^{3/2} \tag{2}$$

where $f = 1$ represents a constant specific to a tumor type.

Table 1
Results of GD metric of different multi-objective algorithms on ZDT and DTLZ, benchmark problems.

Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
ZDT1	2	8.7675e-4 (±1.54e-4) +	2.9643e-2 (±4.85e-3) -	3.3414e-3 (±9.67e-4) =	1.2003e-3 (±4.01e-4) +	2.2214e-3 (±8.08e-4) =	4.0499e-3 (±2.52e-3)
ZDT2	2	6.5216e-4 (±3.71e-4) +	4.1249e-2 (±1.34e-2) =	7.2967e-3 (±5.09e-3) +	1.0337e-2 (±8.86e-3) =	4.4292e-3 (±3.35e-3) +	3.0494e-2 (±2.37e-2)
ZDT3	2	5.1979e-4 (±1.25e-4) +	5.7743e-2 (±2.10e-2) -	1.9822e-3 (±4.18e-4) =	8.9194e-4 (±4.50e-4) =	1.3473e-2 (±7.49e-4) =	2.7297e-3 (±1.29e-3)
ZDT4	2	3.5684e-3 (±5.71e-3) =	5.7787e-2 (±3.41e-2) =	1.0049e-2 (±1.04e-2) =	3.8763e-3 (±6.91e-3) =	1.6251e-3 (±8.37e-4) =	8.4223e-3 (±1.01e-2)
ZDT6	2	1.0722e-3 (±5.78e-4) =	4.0135e-2 (±1.19e-2) -	9.7698e-3 (±2.47e-3) =	1.9075e-3 (±9.29e-4) =	2.3352e-3 (±6.83e-4) =	3.8651e-3 (±4.30e-3)
DTLZ1	2	4.0568e-4 (±3.30e-4) =	2.7363e-2 (±3.75e-2) -	1.0563e-3 (±1.03e-3) =	1.0074e-2 (±1.89e-2) =	7.8004e-4 (±3.65e-4) =	4.2362e-4 (±2.59e-4)
	3	2.2961e-2 (±3.52e-2) =	5.3174e-3 (±5.07e-3) =	4.1232e-2 (±3.29e-2) =	9.4125e-4 (±2.96e-4) =	3.1451e-2 (±3.19e-2) =	3.8623e-1 (±7.19e-1)
DTLZ2	2	1.3673e-5 (±2.09e-6) +	2.5826e-4 (±9.97e-5) =	7.9963e-5 (±2.85e-5) =	1.1263e-4 (±1.80e-5) =	2.0039e-4 (±1.57e-5) =	2.2872e-3 (±2.51e-3)
	3	4.7746e-4 (±5.12e-6) +	5.8552e-4 (±2.11e-5) +	7.0563e-4 (±1.89e-4) +	1.3096e-3 (±1.79e-4) =	2.2396e-3 (±1.56e-4) -	1.1159e-3 (±9.11e-5)
DTLZ3	2	9.6007e-1 (±3.56e-1) =	2.1304e+0 (±5.54e-1) -	1.4074e+0 (±1.28e+0) =	6.2777e-1 (±3.30e-1) =	1.0995e+0 (±8.91e-1) =	8.9649e-1 (±5.40e-1)
	3	2.6689e-1 (±1.21e-1) =	2.0899e+0 (±1.10e+0) =	8.6509e-1 (±5.49e-1) =	7.7575e-1 (±6.92e-1) =	1.1528e+0 (±1.02e+0) =	1.2206e+0 (±9.99e-1)
DTLZ4	2	1.7263e-5 (±4.54e-6) =	2.6500e-4 (±8.24e-5) =	3.4396e-5 (±2.81e-5) =	7.1913e-5 (±5.18e-5) =	1.3560e-4 (±1.17e-4) =	4.3533e-3 (±4.28e-3)
	3	3.8818e-4 (±8.14e-5) =	5.9420e-4 (±1.03e-5) =	5.9794e-4 (±1.51e-5) =	1.3074e-3 (±1.81e-4) =	1.3429e-3 (±7.76e-4) =	6.0688e-3 (±3.34e-3)
DTLZ5	2	1.2702e-5 (±3.51e-6) =	3.4014e-4 (±1.74e-4) =	5.8399e-5 (±1.20e-5) =	1.1579e-4 (±1.95e-5) =	2.2649e-4 (±3.30e-5) =	8.7670e-5 (±2.35e-5)
	3	1.5901e-4 (±3.47e-5) =	5.8576e-2 (±2.78e-2) =	1.8026e-4 (±1.31e-5) =	2.4462e-4 (±1.98e-5) =	5.2623e-4 (±1.33e-4) =	7.4104e-4 (±4.27e-4)
DTLZ6	2	4.5676e-6 (±3.32e-7) =	5.6867e-5 (±6.40e-5) =	4.7033e-6 (±1.59e-7) =	4.7450e-6 (±1.77e-7) =	4.3955e-6 (±3.42e-7) =	5.9014e-2 (±5.12e-2)
	3	4.9259e-6 (±1.74e-7) =	1.2547e-1 (±1.58e-1) =	4.4588e-6 (±5.09e-7) =	5.1619e-6 (±1.96e-7) =	5.0840e-6 (±8.42e-8) =	5.6963e-6 (±2.29e-7)
DTLZ7	2	1.2114e-4 (±5.16e-5) =	8.4829e-3 (±1.24e-3) =	2.0324e-4 (±1.00e-4) =	1.3795e-4 (±4.32e-5) =	3.9810e-4 (±4.27e-5) =	9.3196e-5 (±3.91e-5)
	3	1.8816e-3 (±6.51e-5) =	1.1236e-2 (±1.06e-3) =	3.2221e-3 (±5.98e-4) =	3.4792e-3 (±1.11e-4) =	7.1230e-3 (±1.10e-3) =	3.4115e-3 (±1.36e-3)

Table 2
Results of IGD metric of different multi-objective algorithms on ZDT and DTLZ, benchmark problems.

Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
ZDT1	2	5.4023e-2 (±7.71e-2) =	3.3067e-1 (±7.12e-2) -	7.0648e-2 (±4.92e-2) =	1.1412e-1 (±9.42e-2) =	1.4638e-1 (±1.16e-1) =	1.5460e-1 (±1.01e-1)
ZDT2	2	4.9232e-1 (±1.14e-1) =	1.3738e-1 (±8.17e-2) +	2.9003e-1 (±1.40e-1) +	3.6520e-1 (±1.61e-1) =	4.0609e-1 (±5.28e-2) =	5.3069e-1 (±1.21e-1)
ZDT3	2	2.1423e-1 (±3.25e-2) =	2.0548e-2 (±1.68e-2) +	1.0791e-1 (±1.08e-1) +	9.0405e-2 (±3.94e-2) +	1.7507e-1 (±9.67e-2) =	3.1687e-1 (±1.54e-1)
ZDT4	2	4.6353e-2 (±5.24e-2) +	4.8557e-1 (±2.69e-1) =	2.3999e-1 (±1.36e-1) =	1.2309e-1 (±6.26e-2) +	2.0149e-1 (±1.51e-1) =	2.2709e-1 (±4.20e-2)
ZDT6	2	1.1049e-2 (±4.97e-3) =	1.8204e-1 (±5.62e-2) -	6.8082e-2 (±1.50e-2) -	1.8323e-2 (±8.25e-3) =	2.0725e-2 (±5.42e-3) =	3.2556e-2 (±2.43e-2)
DTLZ1	2	5.6177e-3 (±2.61e-3) =	1.1239e-1 (±1.92e-1) -	1.1570e-2 (±1.11e-2) =	9.6068e-2 (±1.72e-1) =	7.7516e-3 (±3.21e-3) =	5.2738e-3 (±2.09e-3)
	3	1.2296e-1 (±1.83e-1) =	8.3588e-2 (±2.90e-2) =	1.4858e-1 (±1.32e-1) =	3.0571e-2 (±2.77e-3) =	1.9111e-1 (±1.67e-1) =	9.6056e-2 (±1.37e-1)
DTLZ2	2	4.0276e-3 (±4.33e-5) +	5.3732e-3 (±4.55e-4) =	4.8678e-3 (±4.75e-4) +	5.0937e-3 (±1.12e-4) +	5.1877e-3 (±1.41e-4) +	5.8068e-3 (±1.30e-4)
	3	5.4558e-2 (±5.78e-5) +	5.4893e-2 (±1.96e-4) +	5.7645e-2 (±2.35e-3) +	6.8567e-2 (±2.19e-3) =	6.9630e-2 (±1.47e-3) =	6.6784e-2 (±2.02e-3)
DTLZ3	2	2.0893e+0 (±5.74e-1) =	5.2161e+0 (±2.25e+0) =	4.6150e+0 (±3.24e+0) =	2.5535e+0 (±1.52e+0) =	3.3415e+0 (±2.54e+0) =	3.7628e+0 (±1.78e+0)
	3	1.8723e+0 (±6.89e-1) =	8.0894e+0 (±4.15e+0) =	5.0417e+0 (±3.53e+0) =	5.1527e+0 (±4.08e+0) =	9.2679e+0 (±9.20e+0) =	6.0276e+0 (±2.97e+0)
DTLZ4	2	4.0318e-3 (±4.55e-5) =	5.4589e-3 (±8.48e-4) =	1.8888e-1 (±3.69e-1) =	1.8930e-1 (±3.69e-1) =	2.5102e-1 (±4.25e-1) =	2.5155e-1 (±4.25e-1)
	3	3.7881e-1 (±2.81e-1) =	5.4995e-2 (±2.95e-4) =	5.6948e-2 (±1.13e-3) =	6.7180e-2 (±3.08e-4) =	2.2566e-1 (±2.73e-1) =	2.2579e-1 (±2.73e-1)
DTLZ5	2	3.9997e-3 (±1.15e-5) =	6.6857e-3 (±1.56e-3) =	4.5553e-3 (±1.86e-4) =	5.1760e-3 (±5.35e-5) =	5.4136e-3 (±2.66e-4) =	5.9856e-3 (±4.71e-4)
	3	5.4540e-3 (±2.61e-5) =	8.0434e-2 (±1.10e-2) =	1.1413e-2 (±5.01e-4) =	6.1093e-3 (±3.86e-4) =	6.8058e-3 (±3.52e-4) =	1.1479e-2 (±6.08e-4)
DTLZ6	2	3.9745e-3 (±1.00e-5) =	4.0703e-3 (±6.95e-5) =	4.4929e-3 (±2.50e-4) =	5.7368e-3 (±1.92e-4) =	5.3769e-3 (±2.76e-4) =	1.5914e-2 (±3.70e-3)
	3	5.2848e-3 (±4.39e-4) =	9.8160e-2 (±2.09e-2) =	1.4158e-2 (±1.49e-3) =	5.6812e-3 (±1.91e-4) =	5.5162e-3 (±5.47e-5) =	1.4137e-2 (±2.72e-4)
DTLZ7	2	5.6154e-3 (±2.17e-4) =	5.3661e-2 (±7.30e-3) =	5.6368e-3 (±5.05e-4) =	5.3668e-3 (±2.18e-4) =	7.1034e-3 (±3.15e-4) =	1.6780e-1 (±2.50e-1)
	3	6.8965e-2 (±8.15e-4) =	1.2323e-1 (±7.90e-4) =	7.9463e-2 (±4.76e-3) =	7.5169e-2 (±2.07e-3) =	9.4369e-2 (±2.04e-3) =	1.4590e-1 (±2.23e-2)

2.3.2. Tumor duplication

This phase signifies the perilous progression of the disease, ensuing once the tumor's size has evolved. Tumors have the ability to reproduce in another location within the same liver. the exponential growth formula, applicable to breast cancer and liver tumors (hepatocellular carcinoma), is apt for gauging the growth of the tumor's position, as depicted in Eq. (3). The consistency in cell division ensures that the tumor's position V size remains consistent, irrespective of the tumor's actual size.

$$(PG)^i = \frac{dV}{dt} = r * \text{Positiont} \in [1\dots T] \text{ and } i \in [1\dots N] \tag{3}$$

Where PG represents the tumor position's growth, signifies the r radius, taking the form of an ellipse and is defined within [0,1]. The tumor's size and position are derived from the preceding equation. T denotes the maximum iteration count and N stands for the number of search agents. The tumor's proliferation intensifies to pinpoint the best location for disease management within the liver. This mechanism is depicted by the L'evy flight function, as documented and is illustrated in Eqs. (4) and (5).

$$Lv(D) = 0.01 \times \frac{\text{rand}(1,D) \times \sigma}{|\text{rand}(1,D)|^{\frac{1}{\beta}}} \tag{4}$$

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2} \right)^{\frac{1}{\beta}} \tag{5}$$

At this juncture of the LCA, inspired by real-world tumor propagation patterns, it is postulated that tumors can strategically select optimal liver sections for dominance in competitive scenarios. As a result, it is believed that the tumor can determine its subsequent move based on a specific rule to finalize this segment. This process is outlined in Eqs. (6)–(8).

Table 3
Results of SP metric of different multi-objective algorithms on ZDT and DTLZ, benchmark problems.

Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
ZDT1	2	5.9825e-3 (±1.37e-3) +	2.0148e-2 (±2.88e-3) =	7.9935e-3 (±8.35e-4) +	9.7799e-3 (±3.15e-3) +	6.2573e-3 (±1.32e-3) +	5.8727e-2 (±5.96e-2)
ZDT2	2	5.2658e-3 (±2.87e-3) =	3.7342e-2 (±2.08e-2) =	1.6943e-2 (±1.43e-2) =	2.4635e-2 (±3.41e-2) =	2.6723e-2 (±2.38e-2) =	4.2508e-2 (±4.53e-2)
ZDT3	2	6.8647e-3 (±7.45e-4) +	6.1931e-2 (±2.32e-2) =	7.5627e-3 (±1.00e-3) +	1.0160e-2 (±1.45e-3) +	1.5509e-2 (±2.22e-2) =	3.6570e-2 (±3.39e-2)
ZDT4	2	3.9553e-2 (±4.06e-2)	4.3735e-2 (±2.63e-2) =	1.0387e-2 (±4.15e-3) =	1.4384e-2 (±8.68e-3) =	7.2322e-3 (±4.02e-3) +	6.5081e-3 (±7.70e-4) +
ZDT6	2	3.0909e-2 (±8.25e-3) =	3.1727e-2 (±1.27e-2) =	1.6553e-2 (±5.93e-3) +	5.5241e-3 (±4.29e-4) +	6.4902e-3 (±1.16e-3) +	6.7695e-3 (±1.19e-3) +
DTLZ1	2	4.1513e-3 (±1.92e-3) =	4.7272e-2 (±4.72e-2) -	6.3036e-3 (±5.23e-3) =	5.1322e-3 (±2.77e-3) =	4.1170e-3 (±1.33e-3) =	5.1457e-3 (±3.85e-4)
	3	2.1477e-2 (±3.17e-3) =	5.2907e-2 (±1.82e-2) =	4.8904e-2 (±1.69e-2) =	4.8500e-2 (±3.59e-2) =	5.4268e-2 (±2.99e-2) =	3.7813e+0 (±7.09e+0)
DTLZ2	2	6.9890e-3 (±5.10e-4) +	5.6192e-3 (±1.45e-4) +	7.1417e-3 (±1.14e-4) =	6.5408e-3 (±1.21e-4) +	6.3595e-3 (±6.59e-4) +	2.6726e-2 (±2.08e-2)
	3	6.0344e-2 (±4.57e-3) =	5.6667e-2 (±4.35e-4) =	5.2238e-2 (±8.13e-3) =	8.2043e-2 (±1.39e-3) -	5.8508e-2 (±2.31e-3) =	5.7817e-2 (±5.43e-3)
DTLZ3	2	3.0375e-1 (±2.05e-1) =	9.8587e-1 (±4.03e-1) -	6.0747e-1 (±3.91e-1) =	1.4810e+0 (±1.78e+0) =	1.2159e+0 (±1.88e+0) =	2.5172e-1 (±1.91e-1)
	3	5.4941e-1 (±5.42e-1) =	1.2707e+0 (±5.45e-1) =	6.9643e-1 (±6.54e-1) =	1.9799e-1 (±9.99e-2) +	7.0098e-1 (±5.63e-1) =	5.6124e+0 (±9.51e+0)
DTLZ4	2	5.0969e-3 (±3.41e-3) =	6.1294e-3 (±1.04e-3) =	4.9496e-3 (±3.34e-3) =	6.2592e-3 (±2.42e-4) =	None	4.2788e-2 (±4.24e-2)
	3	3.8998e-2 (±2.68e-2) =	5.6529e-2 (±5.10e-4) =	5.2465e-2 (±3.55e-3) =	3.3870e-2 (±4.27e-2) =	5.7076e-2 (±3.50e-3) =	5.9688e-2 (±2.55e-2)
DTLZ5	2	6.2359e-3 (±5.22e-4) =	6.4085e-3 (±2.59e-4) =	6.5371e-3 (±7.04e-4) =	6.5774e-3 (±1.42e-5) =	7.1233e-3 (±9.22e-4) =	8.3850e-3 (±9.62e-4)
	3	9.4384e-3 (±8.64e-4) =	1.2378e-1 (±5.05e-2) =	1.2432e-2 (±4.98e-4) =	1.3168e-2 (±6.84e-4) =	1.1406e-2 (±2.02e-3) =	1.9430e-2 (±2.66e-3)
DTLZ6	2	9.5041e-3 (±7.12e-4) =	5.9759e-3 (±3.20e-4) =	6.7074e-3 (±1.09e-3) =	6.1974e-3 (±2.45e-4) =	1.0161e-2 (±2.69e-4) =	5.9554e-1 (±5.01e-1)
	3	1.0318e-2 (±1.64e-4) =	3.5774e-1 (±4.40e-1) =	1.3279e-2 (±2.71e-3) =	1.2204e-2 (±5.13e-4) =	1.1486e-2 (±3.81e-4) =	2.4481e-2 (±2.92e-3)
DTLZ7	2	7.1490e-3 (±8.90e-4) =	3.0553e-2 (±8.30e-3) =	1.0228e-2 (±5.23e-4) =	1.0693e-2 (±3.54e-4) =	7.6615e-3 (±4.05e-4) =	3.4424e-2 (±3.87e-3)
	3	7.0546e-2 (±8.85e-3) =	1.2855e-1 (±5.56e-3) =	6.6264e-2 (±8.69e-3) =	9.9521e-2 (±2.44e-3) =	7.2643e-2 (±1.19e-2) =	9.6565e-2 (±9.33e-3)

$$y = \text{Position} + PG \tag{6}$$

$$Z = Y + S \times LF(D) \tag{7}$$

$$\text{Position}_{t+1}^{i} = \begin{cases} y & \text{iff fit } (y) < fit\left(\text{Position}_{t}^{i}\right) \\ z & \text{iff fit } (z) < fit\left(\text{Position}_{t}^{i}\right) \end{cases} \tag{8}$$

Here, D symbolizes the dimensional space, S contains D components randomly generated within the [0,1] range. Lastly, $fit(\cdot)$ represents the optimization target fitness function.

2.3.3. Tumor proliferation via crossover and mutation

This concluding phase of the LCA depicts the tumor's evolution into metastatic liver cancer, a severe variant of the ailment that, while originating in the liver, can invade other body parts. To simulate this, the LCA employs genetic mechanisms like mutation and crossover to infuse further variability.

Mutation:

Within the LCA's framework, the mutation function serves as an objective solution. Each element is allocated a random value between [0,1]. If this value nears the mutation rate (zeta), the target agent's tumor position is taken into account. If the value is below the mutation rate (zeta), a component from either the y or z vectors is leveraged to refresh the preceding vector. The mutation function is constructed using equations from Eq. (9) to Eq. (12), showcasing tumor growth via the mutation mechanism. Here, the j th dimension is characterized by lb^j and ub^j, S have D elements sourced from random values in $[0,1]$.

$$y_{\text{Mut}} = \begin{cases} Position & \text{if } \text{rand}_1 \geq \zeta \\ y & \text{else} \end{cases} \text{ and} \tag{9}$$

Table 4
Results of SD metric of different multi-objective algorithms on ZDT and DTLZ, benchmark problems.

Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
ZDT1	2	4.7802e-1 (±1.08e-1) +	8.0837e-1 (±4.89e-2) =	6.8355e-1 (±1.17e-1) +	7.8088e-1 (±1.13e-1) =	7.2890e-1 (±1.46e-1) =	1.0366e+0 (±1.84e-1)
ZDT2	2	7.5071e-1 (±1.03e-1) +	1.0098e+0 (±4.00e-2) =	9.4826e-1 (±8.86e-2) =	1.0004e+0 (±5.31e-2) =	1.0439e+0 (±3.90e-2) =	1.0470e+0 (±5.86e-2)
ZDT3	2	7.0111e-1 (±1.41e-1) +	7.4244e-1 (±1.08e-1) =	4.5277e-1 (±7.82e-2) +	8.2562e-1 (±1.93e-2) =	8.4610e-1 (±1.06e-1) =	1.0069e+0 (±1.46e-1)
ZDT4	2	9.0627e-1 (±5.24e-2) =	4.4233e-1 (±8.54e-2) +	8.4033e-1 (±7.36e-2) +	8.9663e-1 (±1.16e-1) =	7.8403e-1 (±1.67e-1) +	1.1296e+0 (±2.14e-1)
ZDT6	2	3.6335e-1 (±3.68e-2) +	6.8969e-1 (±8.00e-2) +	7.0019e-1 (±5.46e-2) +	5.0279e-1 (±1.26e-1) +	4.8333e-1 (±1.18e-1) +	9.9479e-1 (±1.26e-1)
DTLZ1	2	6.8778e-1 (±2.87e-1) =	4.7003e-1 (±1.42e-1) =	4.5581e-1 (±1.30e-1) =	5.2548e-1 (±2.03e-1) =	4.1533e-1 (±6.64e-2) =	5.2889e-1 (±5.31e-2)
	3	6.6789e-1 (±2.26e-1) =	4.6368e-1 (±4.88e-2) =	5.4355e-1 (±1.20e-1) =	6.2983e-1 (±2.75e-1) =	7.6277e-1 (±3.57e-1) =	1.0334e+0 (±7.14e-1)
DTLZ2	2	1.8060e-1 (±1.09e-2) +	3.6464e-1 (±4.25e-2) +	3.3382e-1 (±5.94e-2) +	2.2514e-1 (±1.57e-2) +	3.4741e-1 (±2.70e-2) +	6.1401e-1 (±1.61e-1)
	3	1.6991e-1 (±6.48e-4) +	5.3590e-1 (±2.85e-2) +	2.2046e-1 (±3.84e-2) +	3.3655e-1 (±1.27e-2) =	4.4036e-1 (±3.94e-2) =	4.1408e-1 (±9.77e-2)
DTLZ3	2	9.1370e-1 (±4.41e-2) =	1.0518e+0 (±1.45e-1) =	9.8004e-1 (±8.98e-2) =	1.2578e+0 (±4.01e-1) =	1.1451e+0 (±1.04e-1) -	8.9381e-1 (±4.12e-2)
	3	6.0172e-1 (±8.20e-2) +	7.9390e-1 (±1.69e-1) =	9.2637e-1 (±7.08e-2) =	9.3720e-1 (±1.11e-1) =	9.9143e-1 (±1.88e-1) =	1.0122e+0 (±4.46e-1)
DTLZ4	2	1.8891e-1 (±2.82e-2) =	5.4754e-1 (±3.03e-1) =	4.6363e-1 (±3.60e-1) =	2.1630e-1 (±7.29e-3) =	None	9.1971e-1 (±1.49e-1)
	3	1.7141e-1 (±4.42e-3) =	4.7323e-1 (±2.34e-2) =	2.0203e-1 (±3.26e-2) =	6.2345e-1 (±2.45e-1) =	5.1924e-1 (±2.18e-1) =	6.1761e-1 (±1.79e-1)
DTLZ5	2	2.0216e-1 (±2.87e-2) =	4.0317e-1 (±2.81e-2) =	2.7386e-1 (±1.31e-2) =	2.1674e-1 (±1.66e-3) =	3.2529e-1 (±3.44e-2) =	5.3147e-1 (±2.24e-2)
	3	3.5241e-1 (±1.26e-1) =	6.2566e-1 (±1.56e-1) =	9.0745e-1 (±4.36e-2) =	6.5332e-1 (±4.66e-2) =	4.3317e-1 (±4.65e-2) =	5.1640e-1 (±7.15e-2)
DTLZ6	2	2.0363e-1 (±1.97e-2) =	8.2522e-1 (±8.27e-2) =	2.8071e-1 (±7.70e-2) =	1.9955e-1 (±1.80e-2) =	6.4855e-1 (±1.22e-1) =	1.5025e+0 (±5.19e-1)
	3	6.6391e-1 (±4.10e-1) =	6.6251e-1 (±3.86e-2) =	1.1918e+0 (±3.79e-2) =	6.3822e-1 (±8.24e-2) =	5.7159e-1 (±3.87e-2) =	1.0154e+0 (±9.81e-2)
DTLZ7	2	5.3147e-1 (±5.05e-2) =	4.2281e-1 (±2.59e-2) =	5.1877e-1 (±2.52e-2) =	4.8007e-1 (±7.25e-3) =	3.7800e-1 (±3.53e-2) =	1.1532e+0 (±8.83e-2)
	3	4.0826e-1 (±2.59e-2) =	5.0904e-1 (±6.81e-2) =	5.5059e-1 (±9.57e-2) =	6.1254e-1 (±4.19e-2) =	5.0074e-1 (±4.45e-2) =	7.8352e-1 (±1.57e-1)

$$\text{Where}: \begin{cases} \zeta = \dfrac{1}{T}; \\ y = |Position - Position_i^j|; \\ z = y - S \end{cases} \tag{10}$$

$$y_{\text{Mut}} = \begin{cases} Position & \text{if } \rho_1 \geq \zeta \\ y & \text{else} \end{cases} \text{ and} \tag{11}$$

$$\text{Where}: \begin{cases} \zeta = \dfrac{t}{T}; \\ y = |Position - Position_i^j|; \\ z = y - S \end{cases} \tag{12}$$

Crossover:

This function amalgamates two entities to generate a new progeny. Within the LCA, a blend of random values τ and τ' is employed to conceive the new progeny w_{Cross}, as detailed in Eq. (13).

$$w_{\text{Cross}} = \tau * y_{Mut} + (1 - \tau') * z_{Mut}, \tau \neq \tau' \tag{13}$$

Selection:

The LCA's adoption of the greedy selection technique is rooted in differential evolution. Offsprings are birthed once the evaluation functions (mutation and crossover) are executed. The efficacy of both progenitors and offsprings is assessed concurrently, allowing superior-performing progenitors to persist in the population. Eq. (14) lays down the criteria for this greedy selection.

Table 5
Results of HV metric of different multi-objective algorithms on ZDT and DTLZ, benchmark problems.

Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
ZDT1	2	8.2615e-1 (±5. 46e-2) =	5.0369e-1 (±3. 76e-2) -	7.9054e-1 (±3. 82e-2) =	7.7861e-1 (±6. 38e-2) =	7.4434e-1 (±7. 90e-2) =	6.9218e-1 (±1. 04e-1)
ZDT2	2	3.6904e-1 (±8. 03e-2) +	5.4981e-2 (±4. 57e-2) =	2.2476e-1 (±1. 03e-1) =	1.8576e-1 (±8. 07e-2) =	1.5409e-1 (±3. 01e-2) =	8.2371e-2 (±6. 01e-2)
ZDT3	2	8.6887e-1 (±1. 31e-1) =	6.1035e-1 (±6. 15e-2) =	9.9527e-1 (±1. 83e-2) +	9.0531e-1 (±4. 84e-2) +	7.9616e-1 (±1. 20e-1) =	6.2783e-1 (±1. 77e-1)
ZDT4	2	8.0908e-1 (±8. 12e-2) +	3.5971e-1 (±2. 44e-1) =	6.2793e-1 (±1. 43e-1) =	7.6493e-1 (±5. 76e-2) +	7.0826e-1 (±1. 11e-1) =	5.2961e-1 (±1. 15e-1)
ZDT6	2	4.1922e-1 (±7. 72e-3) =	1.9897e-1 (±5. 16e-2) -	3.3600e-1 (±1. 93e-2) =	4.0816e-1 (±1. 24e-2) =	4.0481e-1 (±8. 17e-3) =	3.8686e-1 (±3. 93e-2)
DTLZ1	2	1.7258e-1 (±2. 29e-3) =	1.2158e-1 (±8. 12e-2) -	1.6734e-1 (±9. 51e-3) =	1.2795e-1 (±8. 25e-2) =	1.7054e-1 (±2. 77e-3) =	1.7264e-1 (±1. 90e-3)
	3	1.0195e-1 (±6. 57e-2) =	1.1646e-1 (±8. 95e-3) =	8.4064e-2 (±5. 61e-2) =	1.3502e-1 (±1. 15e-3) =	7.2666e-2 (±6. 15e-2) =	1.0949e-1 (±5. 40e-2)
DTLZ2	2	4.2003e-1 (±3. 45e-5) +	4.1750e-1 (±8. 96e-4) =	4.1882e-1 (±7. 42e-4) =	4.1903e-1 (±2. 92e-4) =	4.1817e-1 (±3. 19e-4) =	4.1778e-1 (±7. 69e-4)
	3	7.4323e-1 (±2. 95e-4) +	7.4172e-1 (±4. 46e-4) +	7.3444e-1 (±6. 17e-3) =	7.0699e-1 (±8. 25e-4) -	6.9555e-1 (±2. 22e-3) -	7.1844e-1 (±7. 10e-3)
DTLZ3	2	0.0000e+0 (±0. 00e+0) =	0.0000e+0 (±0. 00e+0) =	0.0000e+0 (±0. 00e+0) =	3.9047e-3 (±7. 81e-3) =	0.0000e+0 (±0. 00e+0) =	0.0000e+0 (±0. 00e+0)
	3	0.0000e+0 (±0. 00e+0) =	0.0000e+0 (±0. 00e+0) =	0.0000e+0 (±0. 00e+0) =	0.0000e+0 (±0. 00e+0) =	0.0000e+0 (±0. 00e+0) =	0.0000e+0 (±0. 00e+0)
DTLZ4	2	4.2003e-1 (±4. 12e-5) =	4.1750e-1 (±1. 17e-3) =	3.4202e-1 (±1. 55e-1) =	3.4197e-1 (±1. 55e-1) =	3.1507e-1 (±1. 78e-1) =	3.1517e-1 (±1. 78e-1)
	3	5.5529e-1 (±1. 63e-1) =	7.4159e-1 (±2. 42e-4) =	7.3871e-1 (±2. 67e-3) =	7.0851e-1 (±4. 39e-3) =	6.2451e-1 (±1. 44e-1) =	6.3350e-1 (±1. 50e-1)
DTLZ5	2	4.1999e-1 (±5. 07e-5) =	4.1617e-1 (±1. 70e-3) =	4.1925e-1 (±5. 10e-4) =	4.1923e-1 (±2. 58e-4) =	4.1792e-1 (±3. 01e-4) =	4.1823e-1 (±2. 38e-4)
	3	1.3233e-1 (±8. 99e-5) =	9.9530e-2 (±6. 31e-3) =	1.2944e-1 (±1. 38e-4) =	1.3223e-1 (±1. 63e-4) =	1.3135e-1 (±2. 82e-4) =	1.2981e-1 (±3. 01e-4)
DTLZ6	2	4.2012e-1 (±4. 10e-6) =	4.1995e-1 (±1. 28e-4) =	4.1964e-1 (±1. 64e-4) =	4.1925e-1 (±2. 94e-4) =	4.1939e-1 (±2. 88e-4) =	4.0910e-1 (±3. 59e-3)
	3	1.3266e-1 (±2. 63e-4) =	8.4606e-2 (±9. 59e-3) =	1.2902e-1 (±8. 67e-4) =	1.3281e-1 (±8. 36e-5) =	1.3291e-1 (±6. 21e-5) =	1.2826e-1 (±2. 25e-4)
DTLZ7	2	1.0066e+0 (±1. 67e-3) =	8.9567e-1 (±1. 18e-2) =	1.0055e+0 (±1. 54e-3) =	1.0081e+0 (±8. 21e-4) =	1.0019e+0 (±3. 79e-4) =	8.9930e-1 (±1. 56e-1)
	3	1.5947e+0 (±4. 52e-3) =	1.4326e+0 (±1. 39e-2) =	1.5456e+0 (±1. 00e-2) =	1.5525e+0 (±1. 09e-2) =	1.4447e+0 (±4. 58e-3) =	1.4712e+0 (±3. 36e-2)

$$\text{Position}^{i}_{t+1} = \begin{cases} y_{Mut} & \text{if fit} (y_{Mut}) < fit\left(\text{Position}^{i}\right) \\ z_{Mut} & \text{if fit} (z_{Mut}) < fit\left(\text{Position}^{i}\right) \\ w_{\text{Cross}} & \text{if fit} (w_{\text{Cross}}) < fit\left(\text{Position}^{i}\right) \end{cases} \tag{14}$$

2.3.4. In developing the LCA, we focused on several key aspects of liver tumor behavior

2.3.4.1. Adaptive growth. The LCA algorithm mimics the liver tumor's adaptive nature, where it seeks the most favorable environment for growth. This feature is translated into the algorithm through an evolutionary search approach, simulating tumor behavior in overtaking liver tissue. It contributes to the LCA's ability to explore and exploit the search space effectively, improving convergence rates.

2.3.4.2. Genetic operators and random opposition-based learning (ROBL). The integration of genetic operators ensures diversity among candidate solutions and prevents premature convergence. ROBL, meanwhile, helps in initializing the population with diverse solutions, avoiding traps in local optima and improving the probability of discovering optimal solutions. These features are inspired by the tumor's genetic variability and its ability to adapt to different environments.

2.3.4.3. Mutation rate adjustment. The LCA employs a mutation rate that adjusts based on the number of iterations. This aspect is inspired by the tumor's ability to mutate and adapt over time. It allows the LCA to explore new and potentially promising locations in the search space, enhancing its overall effectiveness in problem-solving. The pseudocode of the LCA algorithm is discussed in Algorithm 1, while Fig. 1 illustrates the corresponding flowchart.

Table 6
Results of RT metric of different multi-objective algorithms on ZDT and DTLZ, benchmark problems.

Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
ZDT1	2	9.2197e-1 (±7. 15e-2) +	1.1157e+0 (±1. 73e-2) +	1.8911e+0 (±2. 01e-1) +	2.5630e+0 (±2. 98e-1) +	1.6069e+1 (±4. 13e-1) +	1.9214e+1 (±3. 47e-1)
ZDT2	2	1.1827e+0 (±2. 00e-1) +	1.0898e+0 (±3. 77e-2) +	2.0934e+0 (±1. 38e-1) +	1.7793e+0 (±1. 47e-1) +	1.6392e+1 (±4. 43e-1) +	1.9639e+1 (±3. 24e-1)
ZDT3	2	8.4897e-1 (±9. 03e-3) +	1.1154e+0 (±1. 37e-2) +	1.8293e+0 (±4. 60e-2) +	2.7735e+0 (±1. 67e-1) +	1.6296e+1 (±9. 39e-1) +	1.9085e+1 (±3. 98e-1)
ZDT4	2	8.1078e-1 (±3. 22e-2) +	1.0117e+0 (±4. 38e-3) +	1.6608e+0 (±4. 11e-2) +	2.2318e+0 (±2. 46e-1) +	1.6302e+1 (±3. 78e-1) +	1.8696e+1 (±3. 01e-1)
ZDT6	2	7.8030e-1 (±1. 98e-2) +	1.0292e+0 (±6. 92e-3) +	1.8161e+0 (±3. 17e-2) +	2.7229e+0 (±4. 27e-1) +	1.6178e+1 (±4. 11e-1) +	1.8836e+1 (±5. 68e-1)
DTLZ1	2	7.2309e-1 (±2. 40e-2) +	1.0334e+0 (±3. 94e-2) +	1.8301e+0 (±7. 03e-2) +	3.1618e+0 (±6. 70e-1) +	1.5964e+1 (±7. 80e-1) +	1.8677e+1 (±7. 75e-1)
	3	7.5678e-1 (±6. 63e-3) +	1.0691e+0 (±1. 37e-2) +	2.0250e+0 (±6. 48e-2) +	6.3860e+0 (±7. 50e-1) +	1.5929e+1 (±3. 43e-1) +	2.0086e+1 (±6. 88e-1)
DTLZ2	2	7.5435e-1 (±6. 96e-2) +	1.1287e+0 (±9. 15e-2) +	1.7434e+0 (±1. 35e-1) +	1.1150e+1 (±4. 59e-1) +	1.5828e+1 (±4. 69e-1) +	1.9180e+1 (±2. 32e-1)
	3	1.1568e+0 (±1. 03e-1) +	8.1521e-1 (±4. 03e-2) +	1.8336e+0 (±1. 76e-1) +	1.6731e+1 (±4. 15e-1) +	1.6281e+1 (±5. 50e-1) +	2.1337e+1 (±1. 28e+0)
DTLZ3	2	1.0834e+0 (±8. 15e-3) +	8.5100e-1 (±2. 16e-2) +	1.9440e+0 (±1. 61e-1) +	1.4647e+0 (±4. 83e-2) +	1.6207e+1 (±6. 56e-1) +	1.9356e+1 (±2. 58e-1)
	3	1.1510e+0 (±5. 94e-2) +	8.1790e-1 (±1. 62e-2) +	1.9250e+0 (±4. 59e-2) +	2.3774e+0 (±4. 61e-1) +	1.6354e+1 (±2. 75e-1) +	1.9893e+1 (±1. 90e-1)
DTLZ4	2	8.9720e-1 (±3. 49e-1) =	1.0808e+0 (±9. 86e-3) =	1.9382e+0 (±6. 47e-1) =	1.0718e+1 (±1. 99e-1) =	1.6816e+1 (±2. 57e-1) =	2.1993e+1 (±4. 39e+0)
	3	7.9120e-1 (±2. 68e-3) =	1.1275e+0 (±1. 11e-2) =	1.8688e+0 (±2. 94e-1) =	1.2576e+1 (±2. 81e+0) =	1.6355e+1 (±6. 60e-1) =	1.9965e+1 (±2. 38e-1)
DTLZ5	2	7.2161e-1 (±1. 61e-2) =	1.0940e+0 (±1. 55e-2) =	1.6360e+0 (±6. 52e-2) =	1.0840e+1 (±8. 38e-2) =	1.6278e+1 (±1. 21e-1) =	1.9040e+1 (±7. 16e-2)
	3	7.7659e-1 (±1. 09e-2) =	9.7995e-1 (±8. 19e-2) =	2.7205e+0 (±1. 51e-1) =	1.2705e+1 (±4. 44e-2) =	1.6156e+1 (±5. 84e-1) =	1.9674e+1 (±1. 65e-1)
DTLZ6	2	7.5398e-1 (±4. 02e-3) =	1.0955e+0 (±1. 05e-2) =	1.7218e+0 (±2. 76e-2) =	1.2838e+1 (±1. 40e+0) =	1.6244e+1 (±6. 06e-1) =	2.1363e+1 (±9. 04e-1)
	3	7.8930e-1 (±4. 09e-3) =	1.1175e+0 (±1. 10e-1) =	2.7258e+0 (±3. 11e-1) =	1.3780e+1 (±6. 92e-1) =	1.6853e+1 (±1. 40e-1) =	2.2286e+1 (±1. 82e-1)
DTLZ7	2	7.8340e-1 (±1. 71e-2) =	1.1218e+0 (±2. 98e-3) =	1.9262e+0 (±3. 46e-2) =	7.5225e+0 (±4. 23e-1) =	1.5878e+1 (±5. 93e-1) =	1.8842e+1 (±2. 54e-1)
	3	8.3308e-1 (±8. 08e-3) =	1.0326e+0 (±5. 26e-3) =	2.1798e+0 (±1. 02e-1) =	1.3302e+1 (±2. 45e-1) =	1.6456e+1 (±4. 57e-1) =	1.9821e+1 (±1. 86e-1)

Table 7
SP metric of different multi-objective algorithms on constraint and engineering design problems.

| Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
CONSTR	2	4.5352e-2 (±3. 21e-3) +	2.2914e-1 (±1. 12e-1) =	6.5565e-2 (±1. 94e-2) +	5.0697e-2 (±3. 53e-3) +	4.6065e-2 (±5. 73e-3) +	1.8810e-1 (±1. 30e-1)
TNK	2	7.1201e-3 (±1. 77e-4) +	3.9958e-2 (±1. 52e-2) +	8.5420e-3 (±3. 61e-4) +	8.0098e-3 (±7. 10e-4) +	6.6398e-3 (±6. 62e-4) +	3.2620e-1 (±1. 38e-1)
SRN	2	8.7215e-1 (±1. 29e-1) +	1.0229e+1 (±3. 97e+0) -	1.6313e+0 (±1. 35e-1) +	1.0154e+0 (±1. 67e-1) +	1.3888e+0 (±6. 72e-2) +	3.1931e+0 (±4. 13e-1)
OSY	2	2.8411e-1 (±1. 05e-1) +	1.0715e+1 (±1. 42e+1) =	4.2273e-1 (±2. 99e-1) +	3.9182e-1 (±1. 24e-1) +	7.9117e-1 (±4. 47e-1) +	2.7181e+1 (±1. 29e+1)
BNH	2	1.7531e+0 (±7. 37e-1) =	2.5278e+0 (±1. 07e-1) +	8.1832e-1 (±7. 02e-2) +	2.5280e+0 (±7. 36e-3) =	7.2497e-1 (±2. 92e-2) +	2.4408e+0 (±1. 04e+0)
KITA	2	1.4642e-1 (±6. 40e-4) =	3.1138e-1 (±4. 68e-2) =	1.7520e-1 (±1. 61e-2) =	1.4798e-1 (±2. 69e-4) =	1.6259e-1 (±1. 65e-2) =	4.3087e-1 (±3. 92e-1)
RWMOP1	2	1.9165e-2 (±5. 10e-4)	9.7063e-2 (±4. 83e-2)	1.9790e-2 (±2. 76e-3)	1.6289e-2 (±1. 91e-3)	1.8753e-2 (±1. 57e-3)	None
RWMOP2	2	6.7119e-2 (±1. 71e-3)	3.3116e-1 (±5. 91e-2)	9.1180e-2 (±2. 82e-2)	1.5006e-1 (±6. 06e-2)	6.9509e-2 (±9. 39e-2)	None
RWMOP3	2	5.8671e+2 (±2. 11e+2) =	3.5331e+3 (±2. 67e+3) =	1.2076e+3 (±1. 84e+2) =	1.2156e+3 (±2. 29e+2) =	8.3048e+2 (±3. 09e+2) =	3.7413e+4 (±2. 25e+4)
RWMOP4	2	5.8062e+2 (±3. 05e+1) +	3.8455e+3 (±1. 38e+3) =	2.6018e+3 (±5. 68e+2) =	1.4854e+3 (±1. 03e+3) =	5.8084e+2 (±2. 62e+1) +	2.3841e+3 (±1. 54e+3)
RWMOP5	2	2.0704e-1 (±1. 49e-2) +	None	3.8973e-1 (±7. 35e-2) =	4.9349e-1 (±5. 04e-2) =	2.2116e-1 (±1. 19e-2) +	1.8724e+0 (±1. 43e+0)
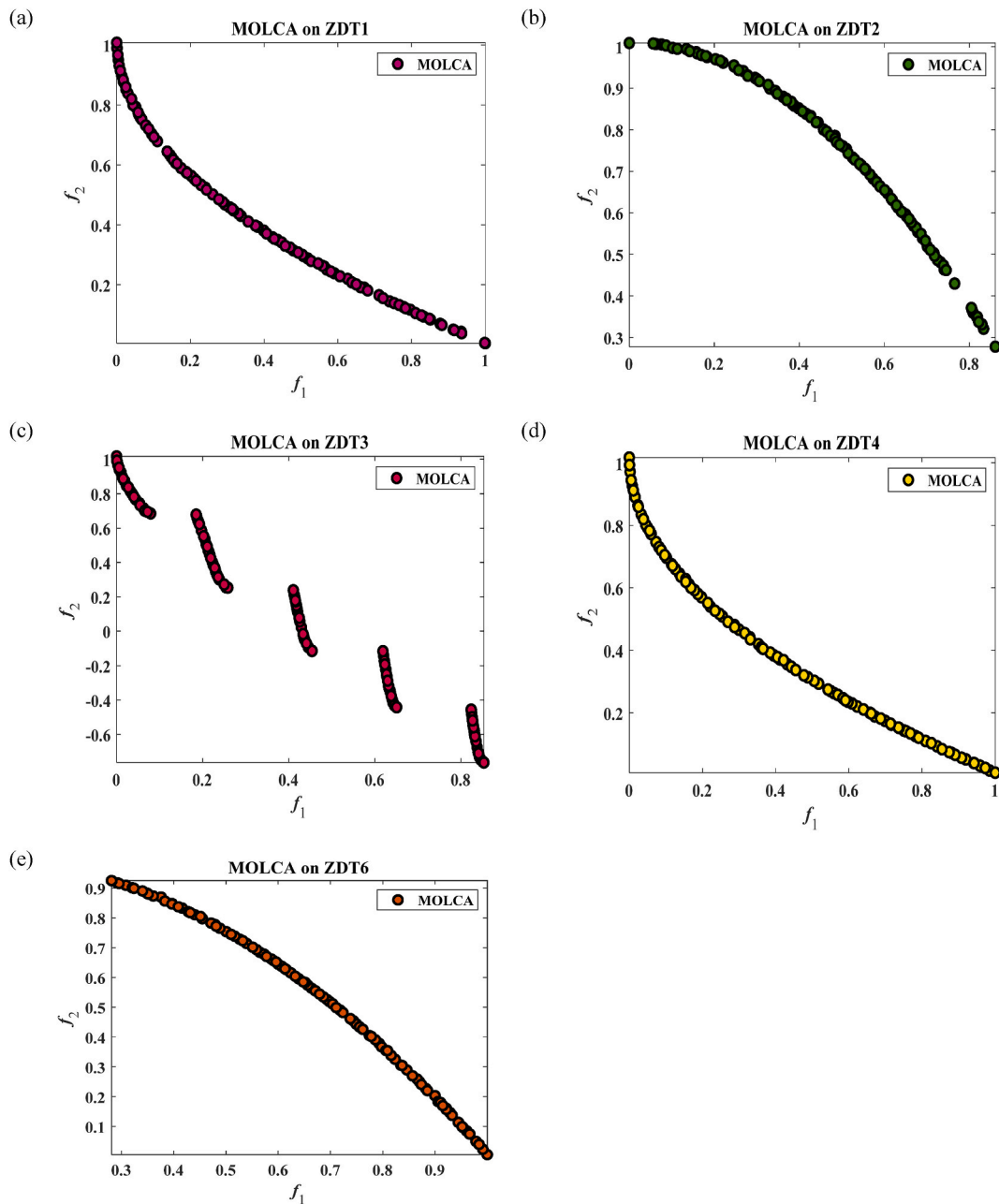
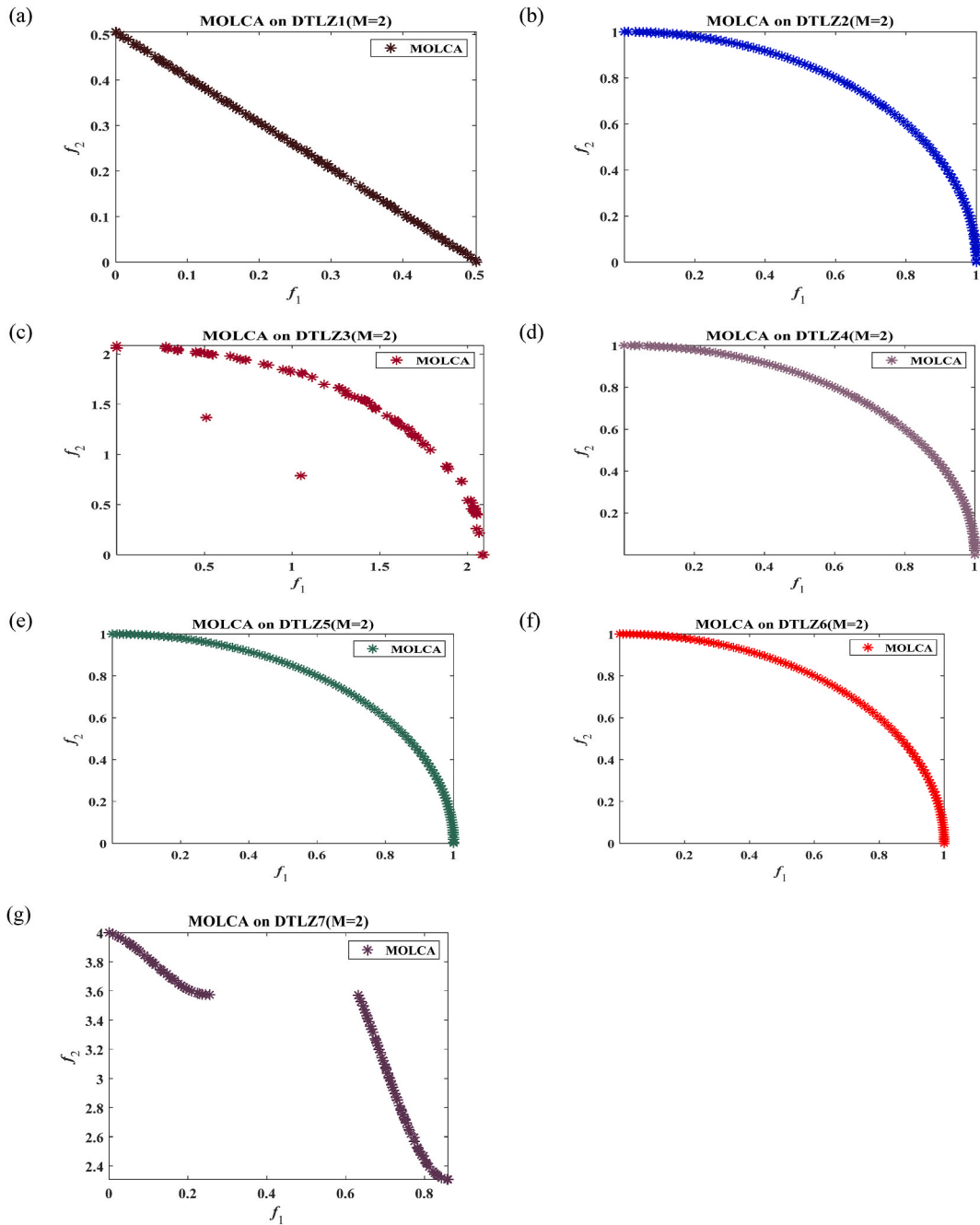**Fig. 6.** Best Pareto optimal front obtained by the MOLCA algorithm on (a) ZDT1, (b) ZDT2, (c) ZDT3, (d) ZDT4, (e) ZDT5, (f) ZDT6 problems.

3. Proposed multi-objective liver cancer algorithm (MOLCA)

3.1. Basic definitions of multi-objective optimization

Multi-objective optimization, as the name suggests, focuses on optimizing across multiple objectives. Within academic discussions, the term multi-objective typically pertains to problems encompassing up to three objectives. Given the intricate nature of problems with more than three objectives, a distinct domain known as many-objective optimization has emerged to address challenges with a multitude of objectives. A primary challenge in the multi-objective domain is the potential conflict between objectives, necessitating specialized strategies. In this context, straightforward relational comparisons between solutions become untenable due to the presence of multiple comparison criteria. As a result, alternative operators are required to gauge the relative superiority of one solution over another. The Pareto dominance operator has gained widespread acceptance for this purpose. It is worth noting that even if circles were to dominate squares in certain visual representations, they would be considered non-dominated relative to each other. This implies
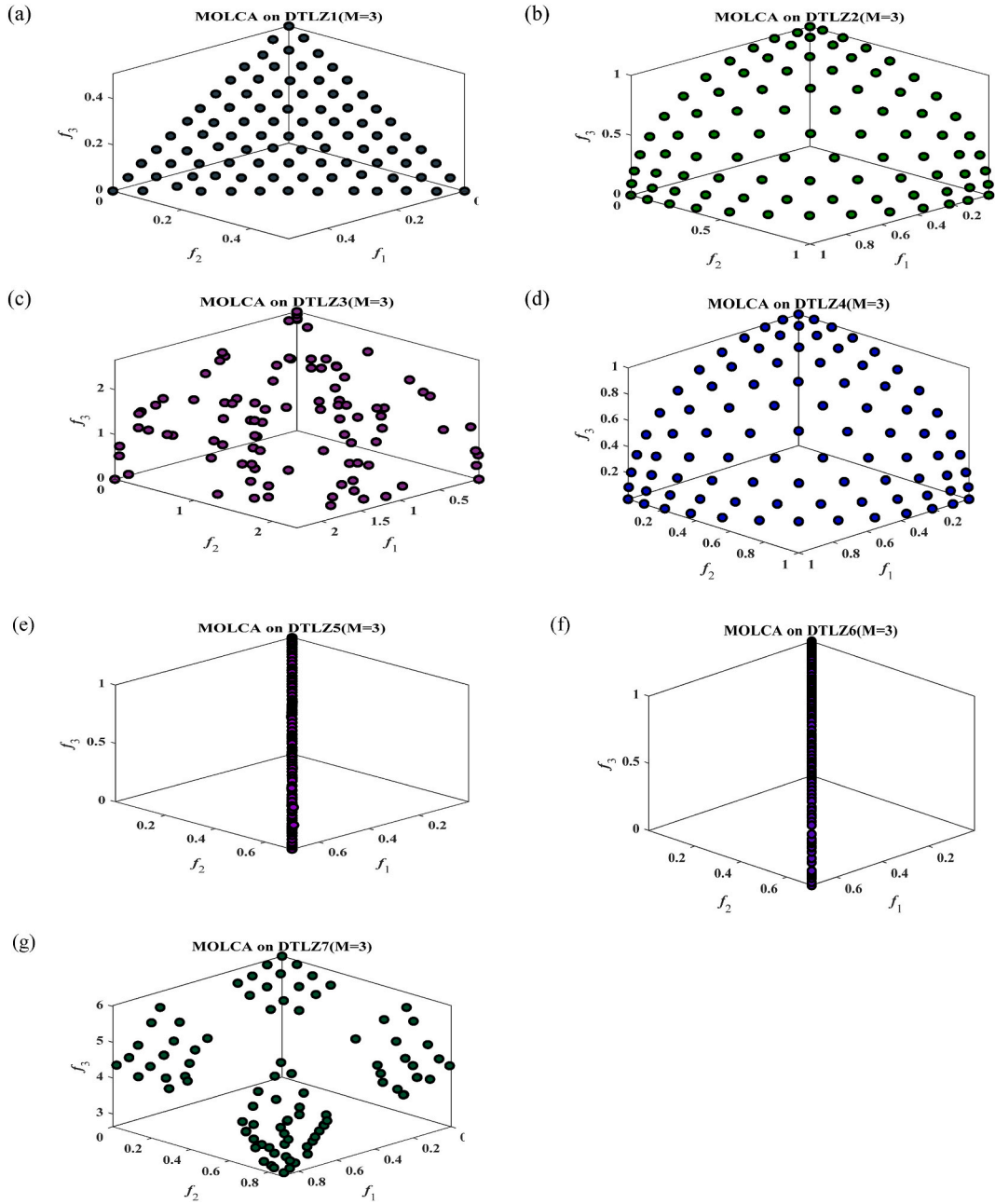
Fig. 7. Best Pareto optimal front obtained by the MOLCA algorithm on (a) DTLZ1, (b) DTLZ2, (c) DTLZ3, (d) DTLZ4, (e) DTLZ5, (f) DTLZ6, (g) DTLZ7 problems with 2-objectives.

that while one might excel in a particular objective, it might lag in another. Mathematically, Pareto optimality can be defined in specific terms: For any given problem, a collection of optimal non-dominated solutions exists. This collection serves as the solution set for MOO. Consequently, the representation of these Pareto optimal solutions within the objective domain is consolidated into a set termed the Pareto optimal front. An elaboration on the idea of domination and associated terminologies are illustrated in Fig. 2.

3.2. Multi-objective liver cancer algorithm (MOLCA)

MOLCA algorithm starts with a random population of size. the current generation is t, x_i^t and x_i^{t+1} the i th individual at t and $(t+1)$ generation. u_i^{t+1} the i th individual at the $(t+1)$ generation generated through the LCA algorithm and parent population P_t. the fitness

Fig. 8. Best Pareto optimal front obtained by the MOLCA algorithm on (a) DTLZ1, (b) DTLZ2, (c) DTLZ3, (d) DTLZ4, (e) DTLZ5, (f) DTLZ6, (g) DTLZ7 problems with 3-objectives.

value of u_i^{t+1} is f_i^{t+1} and U^{t+1} is the set of u_i^{t+1}. Then, we can calculate x_i^{t+1} according to u_i^{t+1} generated through the LCA algorithm and Information Feedback Mechanism (IFM) Eq. (15)

$$x_i^{t+1} = \partial_1 u_i^{t+1} + \partial_2 x_k^t; \ \partial_1 = \frac{f_k^t}{f_i^{t+1} + f_k^t}, \partial_2 = \frac{f_i^{t+1}}{f_i^{t+1} + f_k^t}, \partial_1 + \partial_2 = 1 \qquad (15)$$

where x_k^t is the k th individual we chose from the t th generation, the fitness value of x_k^t is f_k^t, ∂_1 and ∂_2 are weight coefficients. Generate offspring population Q_t. Q_t is the set of x_i^{t+1}. The combined population $R_t = P_t \cup Q_t$ is sorted into different w-non-dominant levels $(F_1, F_2, \ldots, F_l \ldots, F_w)$. Begin from F_1, all individuals in level 1 to l are added to $S_t = \bigcup_{i=1}^{l} F_i$ and remaining members of R_t are rejected illustrated in Fig. 3. If $|S_t| = N$ no other actions are required and the next generation is begun with $P_{t+1} = S_t$ directly. Otherwise, solutions in S_t / F_l are included in P_{t+1} and the remaining solutions $N - \sum_{i=0}^{l-1} |F_i|$ are selected from F_l according to the Crowding Distance
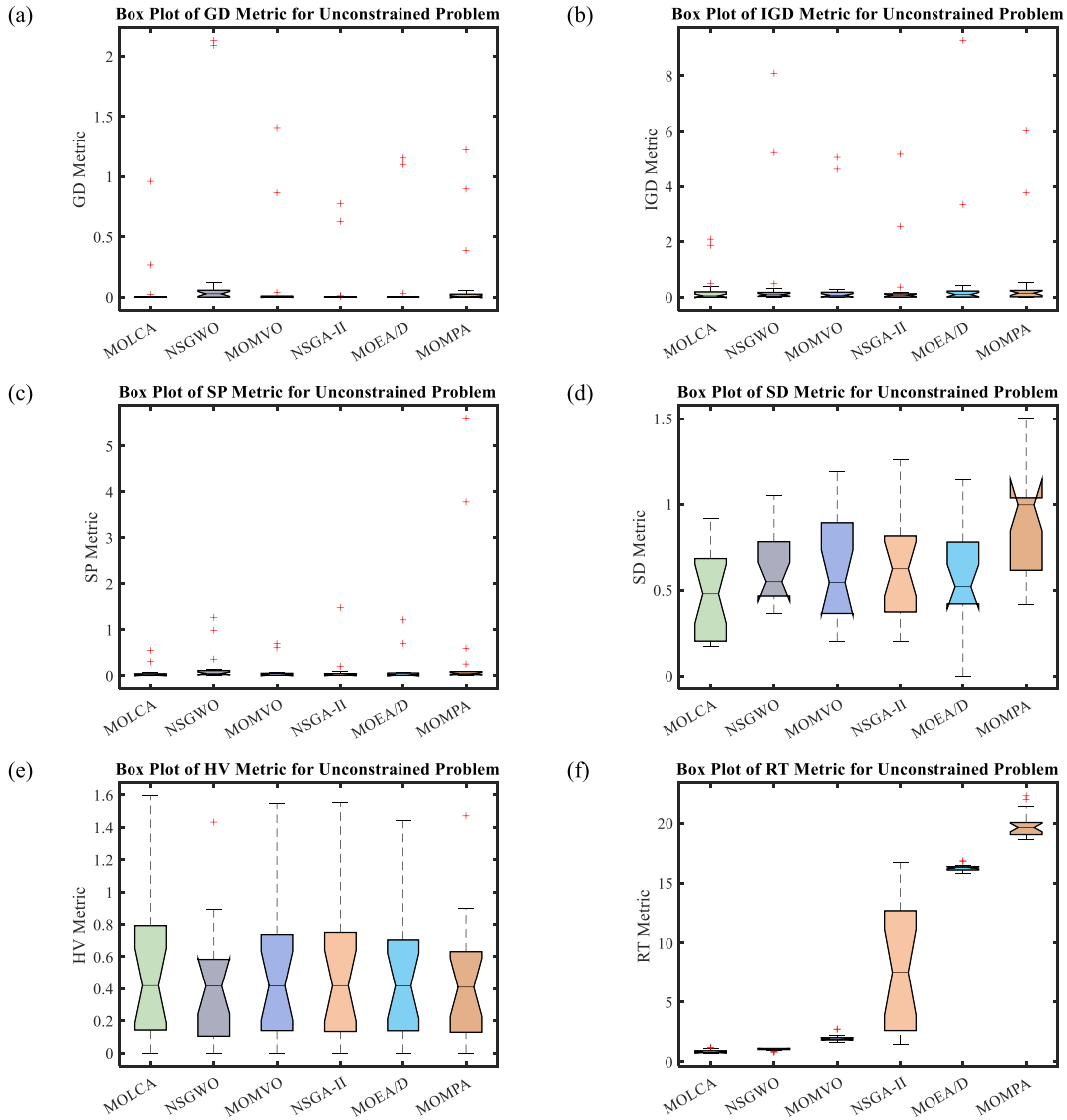
Fig. 9. Box plot of different algorithms on unconstrained problems based on various metrics i.e., (a) GD, (b) IGD, (c) SP, (d) SD, (e) HV and (f) RT metric.

(CD) mechanism, the way to select solutions is according to the CD of solutions in F_l. The larger the crowding distance, the higher the probability of selection and check termination condition is met. If the termination condition is not satisfied, $t = t + 1$ than repeat and if it is satisfied, P_{t+1} is generated represent in **Algorithm-2**, it is then applied to generate a new population Q_{t+1} by LCA algorithm. Such a careful selection strategy is found to computational complexity of M-Objectives $O(N^2M)$. MOLCA that incorporates proposed information feedback mechanism to effectively guide the search process, ensuring a balance between exploration and exploitation. This leads to improved convergence, coverage and diversity preservation, which are crucial aspects of multi-objective optimization. MOLCA algorithm does not require to set any new parameter other than the usual LCA parameters such as the population size, termination parameter and their associated parameters. The flow chart of MOLCA algorithm can be shown in Fig. 4.

4. Experimental results

When evaluating MO algorithms experimental setup on Windows 10 Ultimate on a 64-bit system, powered by an Intel® Core™ i5-2330 M CPU @ 2.20 GHz, with 8 GB of RAM, it is crucial to utilize a range of test functions that each algorithm, we conducted 30 independent runs, with a population size of 40 and a maximum of 500 iterations per run. the parameter settings used in MOLCA (parameter less), NSGWO (a = [0,2]), MOMVO (existence probability ∈ [0.2,1], travelling distance rate ∈ [0.6,1]), NSGA-II (crossover rate = 0.9, mutation rate = 0.1), MOEA/D (Number of neighbors = 15, Crossover parameter-0.5) and MOMPA (FADs = 0.2, P = 0.5).
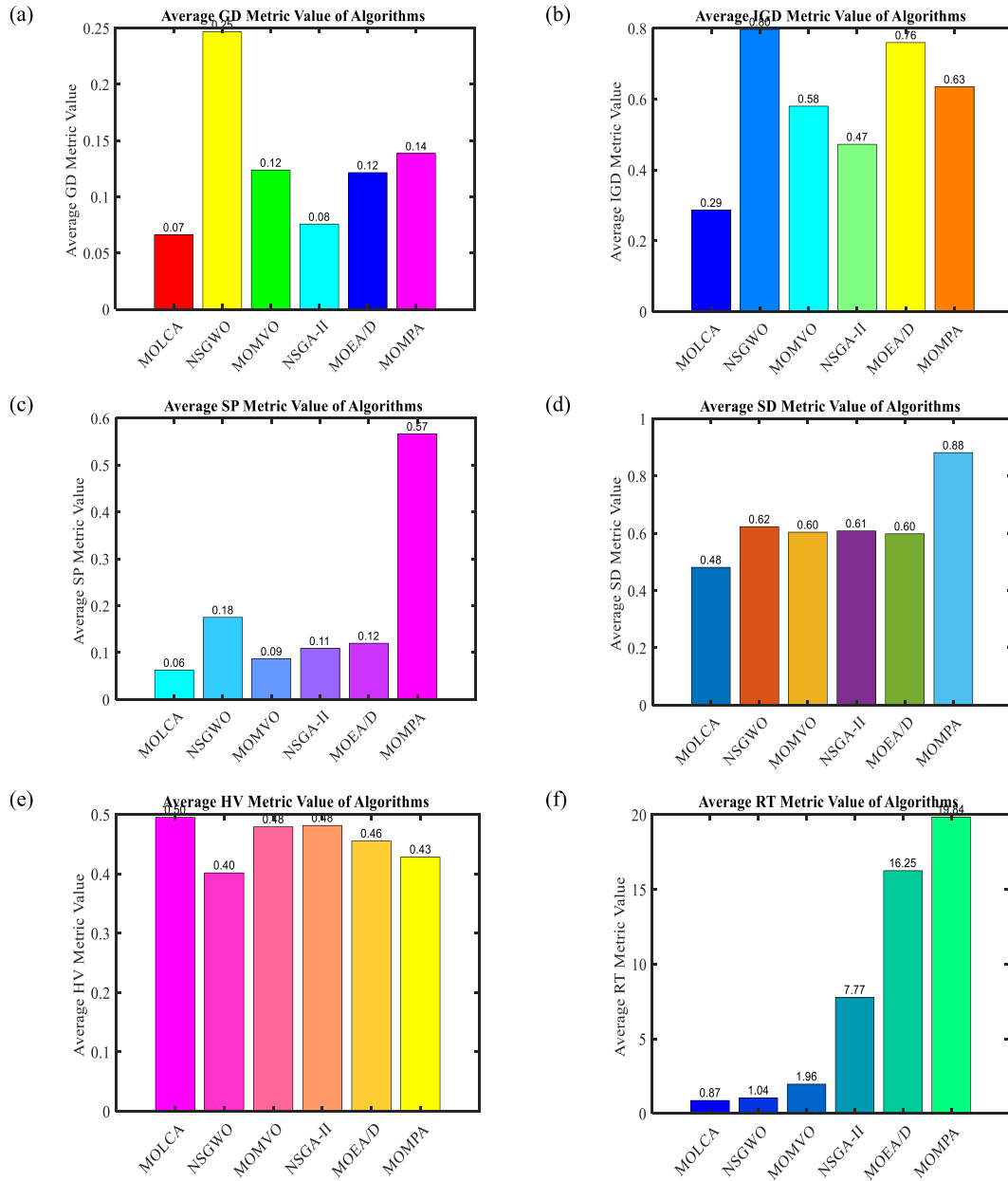
Fig. 10. Bar plot for average different metric values on unconstrained problems on various metrics i.e., (a) GD, (b) IGD, (c) SP, (d) SD, (e) HV and (f) RT metric.

This approach ensures that the algorithm is tested from multiple angles. Several standard test functions are recognized in scholarly literature, including ZDT1-ZDT6 (2-objectives), DTLZ1-DTLZ7 (2 and 3 objectives), Constraint (CONSTR, TNK, SRN, BNH, OSY and KITA) and Real-world engineering design (Brushless DC wheel motor (RWMOP1), Safety isolating transformer (RWMOP2), Helical spring (RWMOP3), Two-bar truss (RWMOP4) and Welded beam (RWMOP5)) problems. The specifics of the test functions used in this study are detailed in Appendix. A closer look reveals that these test functions exhibit diverse Pareto optimal fronts, such as concave, convex, linear and separated. The majority of them operate within a multimodal search space, where multiple local fronts can potentially obstruct solutions from gravitating towards the genuine Pareto optimal front.

To validate the outcomes of the proposed MOLCA algorithm, we juxtaposed its results with esteemed and widely-adopted multi-objective algorithms from the literature, including NSGWO, MOMVO, NSGA-II, MOEA/D and MOMPA algorithms. These results were assessed both qualitatively and quantitatively. On the qualitative front, the optimal Pareto fronts from 30 runs were selected and visualized in the subsequent subsection. Such visual representations offer insights into the relative performance of each algorithm. However, they do not provide a precise measure of superiority. Hence, we incorporated a suite of performance indicators to quantify

Table 8
HV metric of different multi-objective algorithms on constraint and engineering design problems.

Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
CONSTR	2	5.2251e+0 (±7. 03e-4) +	4.8593e+0 (±9. 81e-2) =	5.1768e+0 (±5. 55e-2) +	5.2244e+0 (±5. 56e-3) +	5.2216e+0 (±4. 03e-3) +	4.8179e+0 (±6. 22e-2)
TNK	2	5.2186e-1 (±1. 03e-3) +	4.7175e-1 (±1. 01e-2) +	5.2121e-1 (±7. 30e-4) +	5.2113e-1 (±8. 29e-4) +	5.2101e-1 (±1. 30e-3) +	3.6549e-1 (±3. 52e-2)
SRN	2	2.9875e+4 (±1. 95e+1) +	2.6897e+4 (±4. 43e+2) -	2.9986e+4 (±7. 26e+0) +	2.9974e+4 (±4. 31e+0) +	2.9892e+4 (±9. 90e+0) +	2.9788e+4 (±4. 27e+1)
OSY	2	7.8498e+3 (±3. 56e+3) =	5.5653e+3 (±2. 22e+3) =	3.0966e+3 (±1. 35e+2) -	3.2365e+3 (±3. 31e+2) -	4.7096e+3 (±3. 23e+3) =	7.5598e+3 (±3. 44e+3)
BNH	2	6.4421e+3 (±9. 45e-1) +	6.4257e+3 (±1. 68e+0) =	6.4158e+3 (±8. 28e+0) +	6.4410e+3 (±2. 98e-1) +	6.4396e+3 (±3. 32e+0) +	6.4087e+3 (±1. 59e+1)
KITA	2	4.9690e+1 (±1. 88e-2) +	4.8397e+1 (±3. 62e-1) +	4.9805e+1 (±3. 49e-3) +	4.9806e+1 (±4. 93e-3) +	4.9700e+1 (±3. 15e-2) +	3.5627e+1 (±7. 77e+0)
RWMOP1	2	0.0000e+0 (±0. 00e+0)	0.0000e+0 (±0. 00e+0)	0.0000e+0 (±0. 00e+0)	0.0000e+0 (±0. 00e+0)	0.0000e+0 (±0. 00e+0)	None
RWMOP2	2	0.0000e+0 (±0. 00e+0)	0.0000e+0 (±0. 00e+0)	0.0000e+0 (±0. 00e+0)	0.0000e+0 (±0. 00e+0)	0.0000e+0 (±0. 00e+0)	None
RWMOP3	2	3.4344e+6 (±1. 90e+4) +	3.2540e+6 (±1. 01e+5) +	3.4312e+6 (±2. 10e+4) +	3.4144e+6 (±3. 30e+4) +	3.4480e+6 (±6. 91e+3) +	1.6053e+6 (±5. 82e+5)
RWMOP4	2	4.6799e+3 (±9. 18e-1) +	4.5442e+3 (±1. 92e+1) =	4.6387e+3 (±5. 57e+0) +	4.6428e+3 (±9. 78e+0) +	4.6818e+3 (±1. 95e+0) +	4.1936e+3 (±3. 78e+2)
RWMOP5	2	8.9696e-1 (±2. 12e-3) +	2.7432e-1 (±3. 76e-1) -	8.9117e-1 (±4. 05e-3) +	8.9522e-1 (±2. 96e-3) +	8.9408e-1 (±3. 41e-3)	7.3188e-1 (±2. 02e-1)

Table 9
RT metric of different multi-objective algorithms on constraint and engineering design problems.

Problem	M	MOLCA	NSGWO	MOMVO	NSGA-II	MOEA/D	MOMPA
CONSTR	2	6.8979e-1 (±3. 32e-2) +	9.7621e-1 (±1. 83e-2) +	2.1268e+0 (±9. 80e-2) +	9.4547e+0 (±4. 11e-1) +	1.5308e+1 (±5. 06e-1) +	2.2290e+1 (±6. 10e-1)
TNK	2	7.1622e-1 (±4. 49e-2) +	1.0418e+0 (±3. 97e-2) +	2.0659e+0 (±4. 50e-2) +	3.2000e+0 (±6. 35e-2) +	1.5272e+1 (±4. 68e-1) -	8.5506e+0 (±5. 21e-1)
SRN	2	7.0109e-1 (±5. 28e-2) +	9.3227e-1 (±3. 49e-2) +	1.6041e+0 (±7. 10e-2) +	1.5757e+1 (±6. 41e-1) =	1.5384e+1 (±6. 69e-1) =	1.5545e+1 (±1. 31e-1)
OSY	2	1.6641e+0 (±5. 12e-2) +	1.0014e+0 (±3. 02e-2) +	7.8572e-1 (±2. 05e-2) +	3.4290e+0 (±2. 60e-1) +	1.5841e+1 (±2. 78e-1) +	6.6616e+0 (±5. 53e-1)
BNH	2	1.9059e+0 (±2. 26e-1) +	1.4562e+0 (±1. 05e+0) +	7.9631e-1 (±3. 59e-1) +	1.6710e+1 (±2. 29e+0) =	1.6286e+1 (±1. 99e+0) =	1.5751e+1 (±4. 97e-1)
KITA	2	7.4207e-1 (±9. 96e-2) +	9.9775e-1 (±1. 36e-2) +	1.6468e+0 (±1. 46e-2) +	1.8094e+1 (±2. 05e-1) +	1.6104e+1 (±7. 98e-1) +	2.2623e+1 (±2. 63e-1)
RWMOP1	2	1.9002e+0 (±2. 29e-1) +	1.8710e+0 (±1. 98e-1) +	2.7807e+0 (±1. 98e-1) +	7.1500e+0 (±4. 90e-1) +	1.7648e+1 (±7. 07e-1) -	2.5801e+0 (±2. 18e-1)
RWMOP2	2	5.2129e+0 (±2. 91e-1) +	4.9956e+0 (±2. 18e-1) +	5.8608e+0 (±1. 20e-1) +	7.7272e+0 (±6. 92e-1) =	2.4080e+1 (±5. 97e-1) -	8.1176e+0 (±3. 67e-1)
RWMOP3	2	7.9719e-1 (±3. 39e-2) +	9.2981e-1 (±4. 48e-2) +	1.7136e+0 (±1. 96e-1) +	2.0572e+0 (±1. 68e-1) +	1.5393e+1 (±6. 38e-1) -	3.3884e+0 (±3. 54e-1)
RWMOP4	2	6.2314e-1 (±2. 13e-2) +	8.7154e-1 (±4. 04e-2) +	1.4844e+0 (±5. 37e-2) +	7.9026e+0 (±2. 33e-1) +	1.4741e+1 (±6. 70e-1) =	1.4962e+1 (±8. 30e-1)
RWMOP5	2	8.7699e-1 (±9. 74e-3) +	9.3375e-1 (±1. 29e-2) +	1.6530e+0 (±3. 12e-2) +	7.9414e+0 (±4. 38e-1) -	1.5958e+1 (±3. 14e-1) -	6.6208e+0 (±1. 00e+0)

both convergence and coverage of the algorithms. The GD, IGD, RT and HV metric gauges algorithm convergence, enabling us to ascertain the proximity of derived Pareto optimal solutions to the true ones. Yet, ensuring a broad solution distribution across all objectives remains pivotal. To quantify this coverage and facilitate algorithmic comparisons, we employed the spacing (SP) and spread (SD) metrics. It is worth noting that lower values for GD, IGD, RT and SP indicate superior outcomes, while a higher SD and HV value signifies enhanced coverage. To provide a qualitative comparison and visualize solution distribution, the optimal Pareto front for each algorithm will also be showcased in the subsequent subsection. Graphical performance indicators shown in Fig. 5., a statistical evaluation. "$+/-/\sim$" Wilcoxon signed-rank test (WSRT) was conducted at a significance level of 0.05 between the total amount of test problems on which the corresponding optimizers has a better performance, a worse performance and an equal performance for solving MO problems.
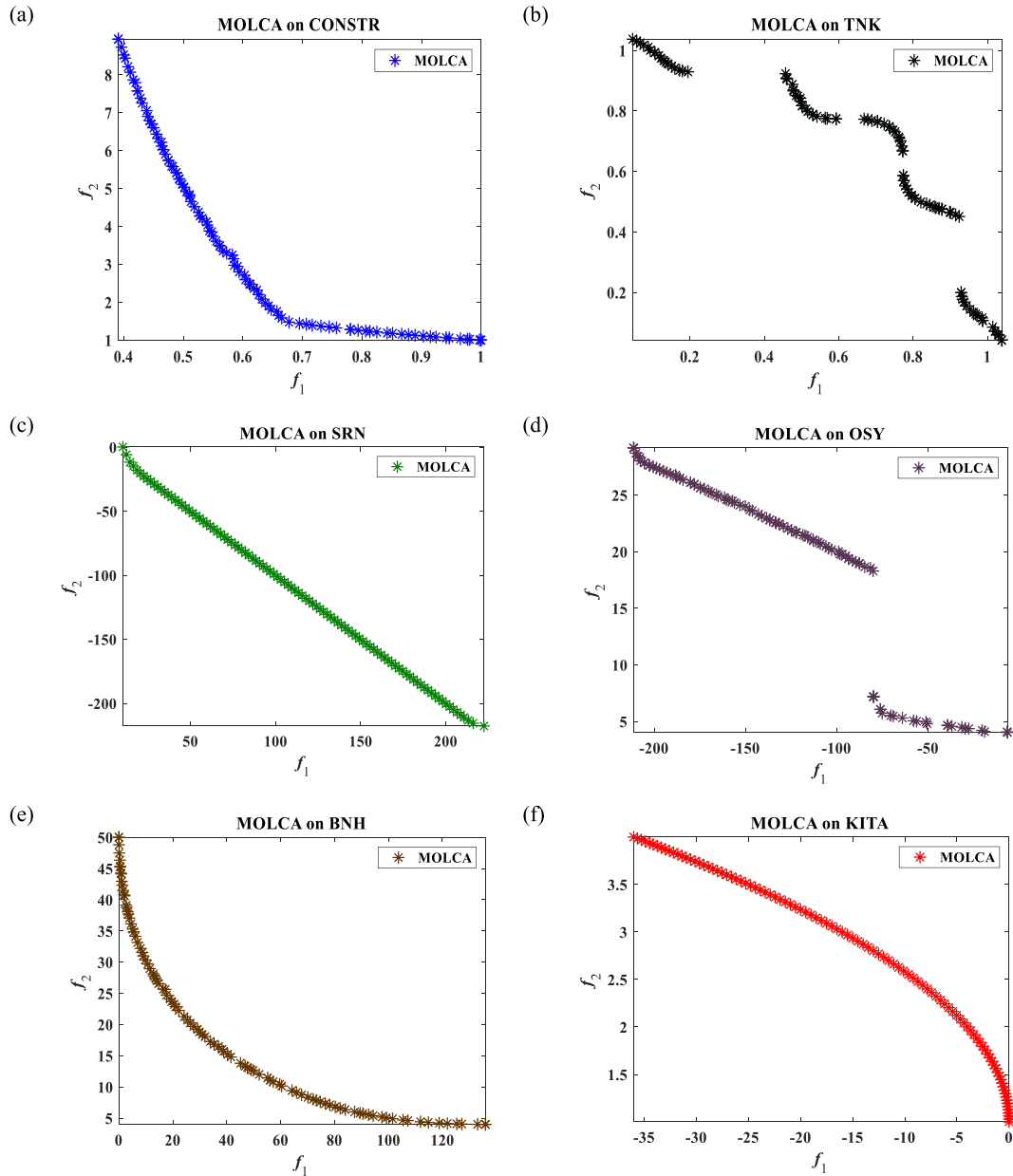
(a)

(b)

(c)

(d)

(e)

(f)

Fig. 11. Best Pareto optimal front obtained by the MOLCA algorithm on various constrained problems i.e., (a) CONSTR, (b) TANK, (c) OSY, (d) KITA, (e) BNH and (f) SRN.

4.1. Performance evaluation on ZDT and DTLZ benchmarks

Table 1 showcases the average GD-metric outcomes from the MOLCA, NSGWO, MOMVO, NSGA-II, MOEA/D and MOMPA algorithms across the 19 test functions. The MOLCA algorithm outperforms in the GD-metric for all functions, with the exceptions being ZDT4 and DTLZ6. the summarized "+/ = /−" outcome of "5/15/0". Furthermore, in many instances, the average GD-metric scores achieved by MOLCA are just 100%. This suggests that the Pareto set derived from MOLCA is more optimally distributed than those from other algorithms. Table 2 presents the IGD-metric values. For this metric, we A reduced IGD-metric signifies a better solution in search space. This data implies that MOLCA excels in four test functions in terms of the IGD-metric. Simultaneously, MOLCA's performance is "3/16/0" summarized "+/ = /−" outcomes. However, it surpasses NSGWO, MOMVO, NSGA-II, MOEA/D and MOMPA algorithms. Broadly speaking, the IGD-metric evaluates an algorithm's diversity, leading to the inference that MOLCA exhibits strong divergence capabilities. Table 3, highlighting the SP-metric results, reveals that MOLCA clinches the top spot in eight of the 19 test functions, while the other algorithms capture the remaining seven. MOLCA's dominance is evident, as shown by the "3/13/3"
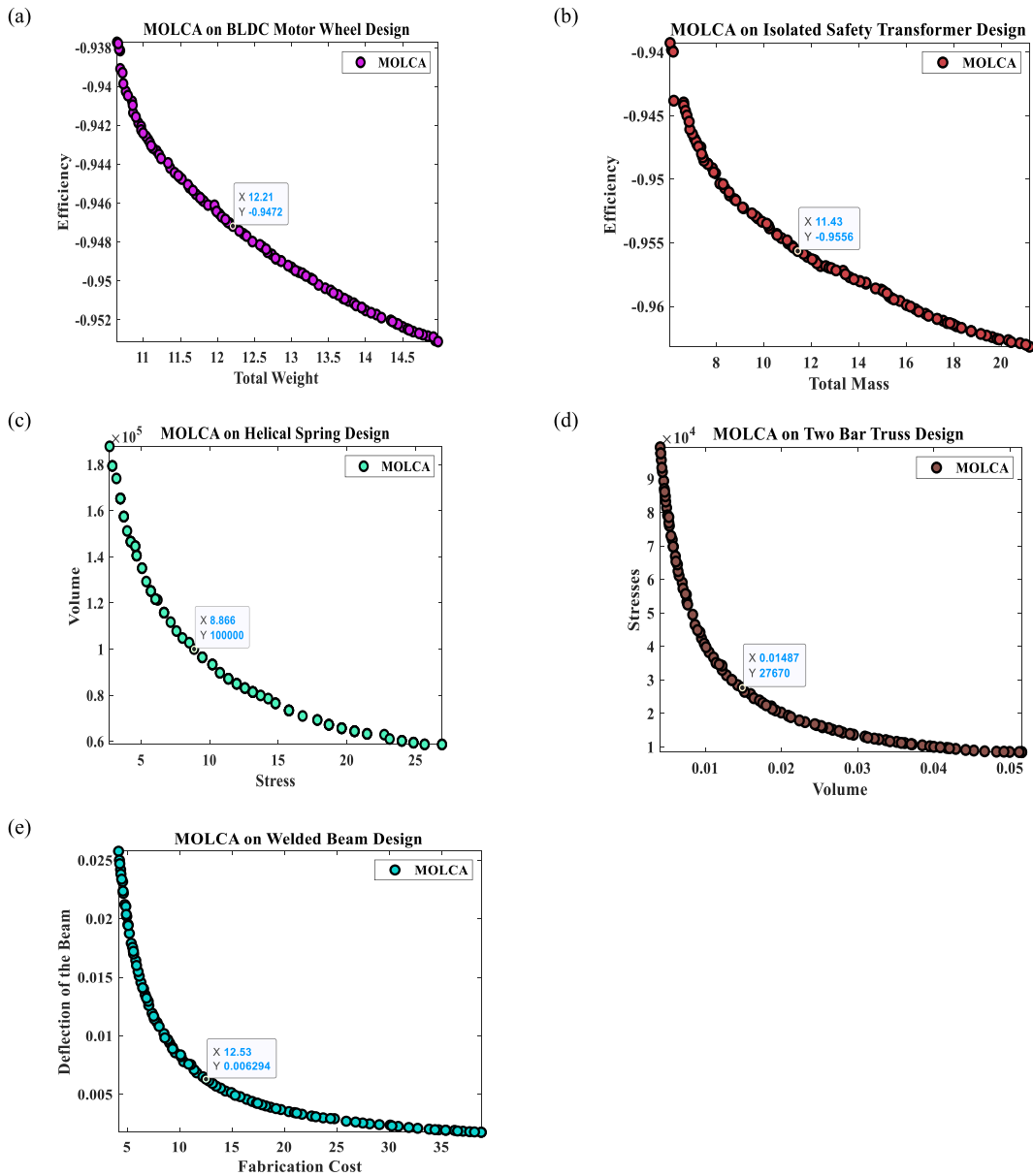
Fig. 12. Best Pareto optimal front and compromise solution obtained by the MOLCA algorithm on real-world engineering problems (a) RWMOP1, (b) RWMOP2, (c) RWMOP3, (d) RWMOP4 and (e) RWMOP5.

summarized "+/ = /−" outcome. These scores underscore the superior distribution of the solution set achieved by MOLCA. Table 4 displays the comparative SD-metric outcomes. Here, we assign as the solution set from these algorithms and as the genuine Pareto set. Based on Table 4's data, MOLCA stands out, securing top scores, suggesting its superior stability and spread. Table 5 presents the HV-metric values. For this metric, we A higher HV-metric signifies a better solution in search space. This data implies that MOLCA excels in four test functions in terms of the HV-metric. Simultaneously, MOLCA's performance is "4/15/0" summarized "+/ = /−" outcomes. This suggests that the Pareto set derived from MOLCA is more optimally distributed and diversified than those from other algorithms. Table 7 displays the comparative RT-metric outcomes. Based on Table 7's data, MOLCA stands out, securing top scores, suggesting its superior speed, stability and less computational burden (see Table 6).

To get a clearer visual representation of each algorithm's efficacy on every test function, Fig. 6, Figs. 7 and 8 visually contrasts the typical non-dominated solution set with the true PF. These results are based on the outcomes of 30 runs across the 19 test functions. From an analysis of Figs. 6, Figs. 7 and 8 It is evident that for the ZDT and DTLZ test functions, MOLCA's performance closely in terms of uniformity. However, it surpasses other algorithms due to its superior distribution concerning universality, uniformity and convergence. Drawing from the box plot shown in Fig. 9 and bar plot shown in Fig. 10, It is clear that the MOLCA algorithm offers a robust and efficient approach to tackle MOPs.
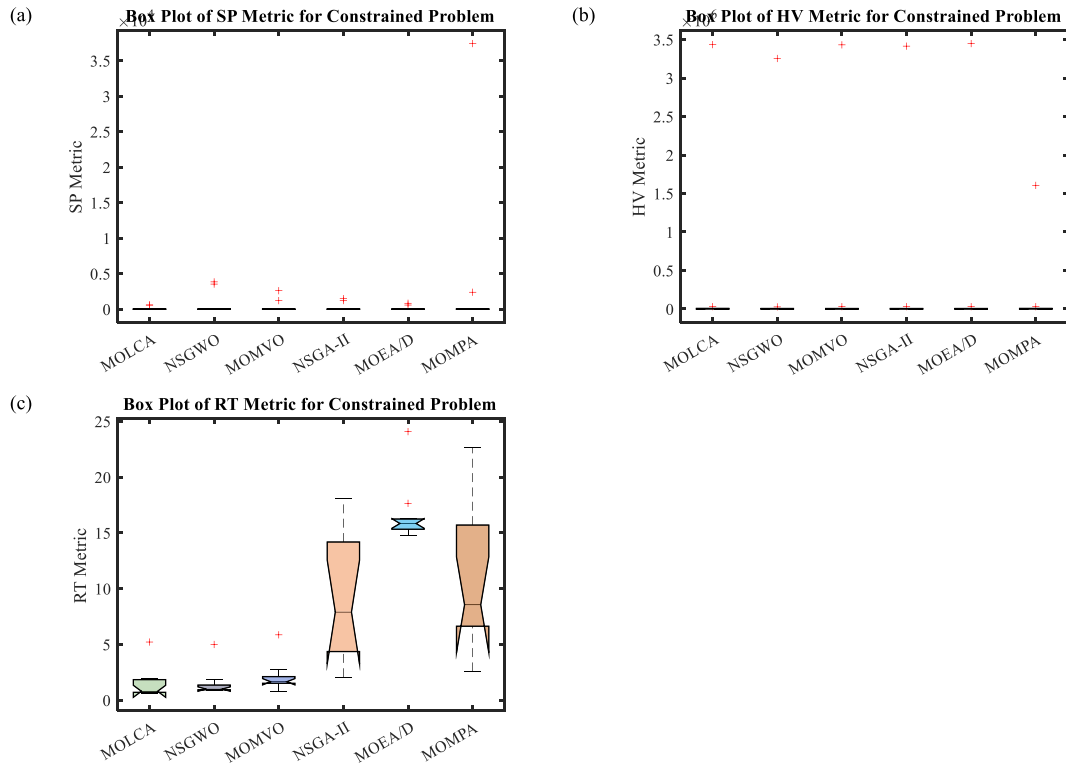
Fig. 13. Box plot by different algorithms on constrained (real-world) problems on various metrics i.e., (a) SP, (b) HV and (c) RT metric.

4.2. Performance evaluation on constraint and real-world engineering design problems

To further assess MOLCA's capabilities, we have incorporated six constraint (CONSTR, TNK, SRN, BNH, OSY and KITA) and five Real-world engineering design (Brushless DC wheel motor (RWMOP1), Safety isolating transformer (RWMOP2), Helical spring (RWMOP3), Two-bar truss (RWMOP4) and Welded beam (RWMOP5)) problems into our experiments. The parameters for NSGWO, MOMVO, NSGA-II, MOEA/D and MOMPA remain consistent with the original papers, with a set population size of 100. In our research, we juxtaposed the outcomes from the suggested MOLCA with those from NSGWO, MOMVO, NSGA-II, MOEA/D and MOMPA. Given the unknown characteristics of the aforementioned engineering challenges and constraint problems, it was imperative to construct a reference PF to gauge the performance metric. Our empirical analysis involved a two-step process for each engineering issue to construct the reference PF: 1) We recorded the non-dominated solutions from all algorithms across 30 separate runs; 2) Using these solutions, a clustering technique was utilized to pick 6000 non-dominated solutions, forming the reference set for performance evaluation. For spacing (SP) and hypervolume (HV) considerations.

Table 7, Table 8 and Table 9 showcases the outcomes from the introduced MOLCA and NSGWO, MOMVO, NSGA-II, MOEA/D and MOMPA based on the performance metrics for the five real-world challenges. Evidently, MOLCA surpassed in SP, HV and RT values compared to NSGWO, MOMVO, NSGA-II, MOEA/D and MOMPA for constraint and engineering tasks. This indicates that the non-dominated solutions by MOLCA demonstrated superior convergence and distribution towards the actual PF. Furthermore, MOLCA notably excelled over other multi-objective strategies, especially for the CONSTR, TNK, KITA, RWMOP2, RWMOP5 test issue, as per the Mann–Whitney–Wilcoxon statistical evaluation with a significance level of 0.05. As previously mentioned, the performance metric offers insights into a multi-objective algorithm's comparative efficacy. To gain a clearer understanding of the distance between a solution set and a reference PF, it is beneficial to use performance metrics rooted in reference sets. Accordingly, the MOEAs were assessed using the performance metric, with the reference being the approximation crafted as previously detailed.

The latter section of Table 7, Table 8, Table 9 presents outcomes related to the SP, HV and RT performance metric. A review of this table reveals that MOLCA, our proposed model, achieved superior performance values across constraint and real-world engineering challenges. This aligns with the earlier assessment made using the performance indicator, where MOLCA surpassed other algorithms. Now, this superiority is evident in the context of the run time (RT) computational burden performance metric. Given the extensive reference set, comprising 6000 solutions for each issue, the SP and HV performance metric offers a more nuanced evaluation of the closeness and spread of the non-dominated solutions along the PF. Hence, it is evident that MOLCA provides a more refined approximation and distribution of solutions on the estimated PF. Turning to Figs. 11 and 12 it visually presents the PF approximations derived from MOLCA. we have employed a fuzzy approach to identify and incorporate a compromise solution for real world engineering problems shown in Fig. 12. The coverage and spacing in obtained PFs show superiority of MOLCA for handling the real-world problems. Based on box plot shown in Fig. 13 and bar plot shown in Fig. 14, we deduce that MOLCA stands out as a promising solution
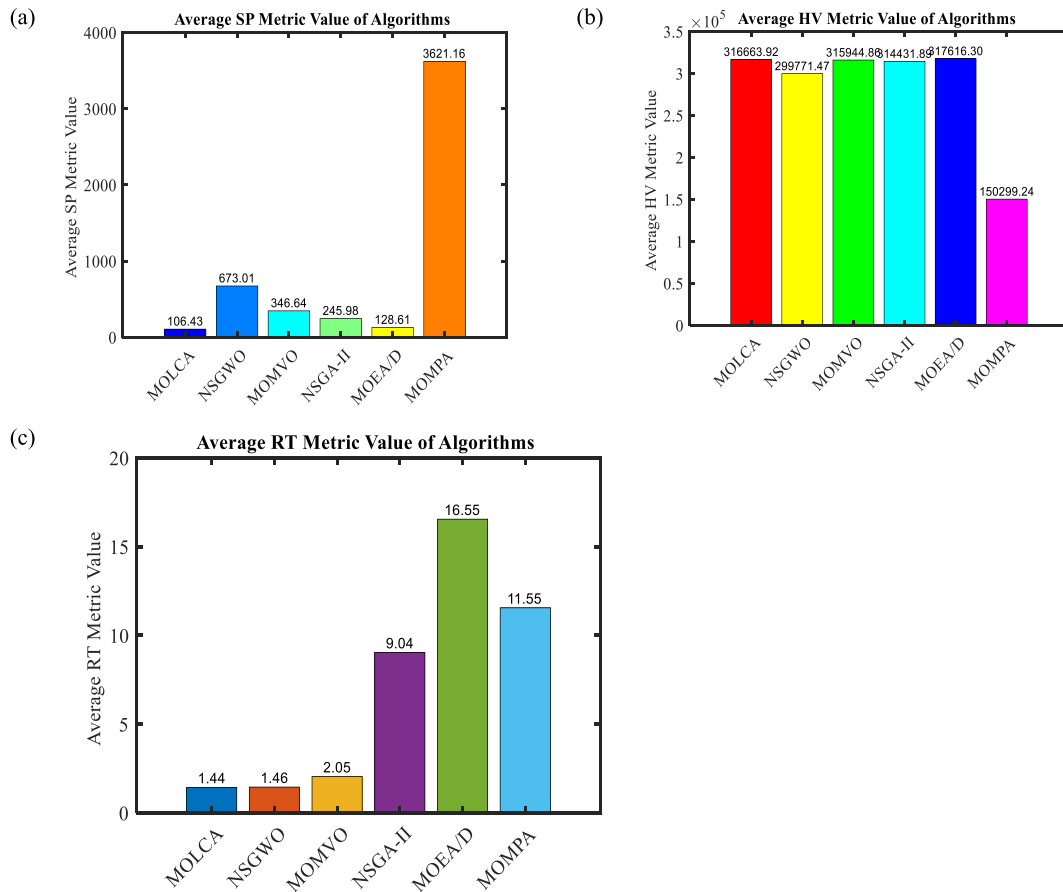
(a)

(b)

(c)

Fig. 14. Bar plot for average different metric values on constrained (real-world) problems on various metrics i.e., (a) SP, (b) HV and (c) RT metric.

for real-world challenges.

5. Conclusion

In this study, we introduced a multi-objective algorithm from the growth and proliferation patterns of liver tumors. Initially, a mathematical framework was utilized to replicate the dynamics of a liver cancer, leading to the development of LCA algorithm. Subsequently, elitist non-dominated sorting (NDS), information feedback mechanism (IFM) and Crowding Distance (CD) selection process was incorporated to adapt this algorithm for multi-objective challenges. The efficacy of the newly devised MOLCA algorithm was assessed using a variety of test functions. Its performance was benchmarked against established algorithms like NSGWO, MOMVO, NSGA-II, MOEA/D and MOMPA, using performance metrics such as GD, IGD, SP, SD, HV and RT. Analytical findings and statistical insights reveal that MOLCA often outperforms its counterparts in multiple testing scenarios. The evaluations revealed that MOLCA stands out as a robust contender, adept at deriving precise approximations of the Pareto optimal front while ensuring a broad distribution across objectives. The algorithm's precision can be attributed to its impressive convergence rate, while its expansive exploration capabilities contribute to its broad distribution. The integration of the Information Feedback Mechanism (IFM) further enhances the algorithm's exploration and solution distribution capabilities. Looking ahead, it would be valuable to explore the potential of various constraint management techniques in conjunction with MOLCA, especially when addressing multi-objective challenges with equality constraints. The MOLCA source code is available at: https://github.com/kanak02/MOLCA.

Funding

No Funding.

Institutional review board statement

Not applicable.

Informed consent statement

Not applicable.

Data availability statement

The data presented in this study are available through email upon request to the corresponding author.

CRediT authorship contribution statement

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Unconstrained multi-objective ZDT [43] test problems utilized in this work

A.1 ZDT1 with Convex POF

$$F = (f_1(x), f_2(x)), where f_1(x) = x_1, f_2(x, g) = g(x) \cdot \left(1 - \sqrt{\frac{f_1}{g(x)}} \right),$$

$and, g(x) = 1 + \frac{9}{n-1} \cdot \sum_{i=2}^{n} x_i,$ Constraints: $0 \leq x_i \leq 1, n = 30, i = 1, 2, ..., 30$

A.2 ZDT 2 with non-convex POF

$$F = (f_1(x), f_2(x)), where f_1(x) = x_1, f_2(x, g) = g(x) \cdot \left(1 - \left(\sqrt{\frac{f_1}{g(x)}} \right)^2 \right),$$

$and, g(x) = 1 + \frac{9}{n-1} \cdot \sum_{i=2}^{n} x_i,$ Constraints $0 \leq x_i \leq 1, n = 30, i = 1, 2, ..., 30$

A.3 ZDT 3 with convex and discrete POF

$$F = (f_1(x), f_2(x)), where f_1(x) = x_1, f_2(x, g) = g(x) \cdot \left(1 - \sqrt{\frac{f_1}{g(x)}} - \frac{f_1}{g(x)} \cdot \sin(10\pi f_1) \right),$$

$and, g(x) = 1 + \frac{9}{n-1} \cdot \sum_{i=2}^{n} x_i,$ Constraints: $0 \leq x_i \leq 1, n = 30, i = 1, 2, ..., 30$

A.4 ZDT 4 with non-convex POF

$$F = (f_1(x), f_2(x)), where f_1(x) = x_1, f_2(x, g) = g(x) \cdot \left(1 - \sqrt{\frac{f_1}{g(x)}} \right),$$

$$and, g(x) = 1 + 10 \cdot (n - 1) + \sum_{i=2}^{n} (x_i^2 - 10 \cos(4\pi x_i)),$$

Constraints: $0 \leq x_1 \leq 1, -5 \leq x_i \leq 5, n = 10, i = 1, 2, ..., 10$

A.5 ZDT 6 with non-uniformly and non-convex POF

$$F = (f_1(x), f_2(x)), where f_1(x) = 1 - \exp(4x_1).sin^6(6\pi x_1), f_2(x, g) = g(x).\left(1 - \left(\frac{f_1}{g(x)}\right)^2\right),$$

$$and, g(x) = 1 + \frac{9}{n-1}.\left[\frac{\sum_{i=2}^n x_i}{9}\right]^{0.25}, \text{ Constraints: } 0 \le x_i \le 1, n = 10, i = 1, 2, ..., 10$$

Appendix B. Unconstrained multi-objective DTLZ with 2-Dim [45] test problems utilized in this work

B.1 DTLZ 1 with linear POF

$$F = (f_1(x), f_2(x)), where f_1(x) = \frac{1}{2}x_1(1 + g(x)), f_2(x) = \frac{1}{2}(1 - x_1)(1 + g(x))$$

$$and, g(x) = 100\left[1 + \sum_{i=3}^n (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))\right],$$

Constraints: $0 \le x_i \le 1, n = 12, i = 1, 2, ..., 12$

B.2 DTLZ 2 with concave POF

$$F = (f_1(x), f_2(x)), where f_1(x) = cos\left(\frac{\pi}{2}x_1\right)(1 + g(x)), f_2(x) = sin\left(\frac{\pi}{2}x_1\right)(1 + g(x))$$

$$and, g(x) = \sum_{i=3}^n (x_i - 0.5)^2, \text{ Constraints: } 0 \le x_i \le 1, n = 12, i = 1, 2, ..., 12$$

B.3 DTLZ 3 with concave POF

$$F = (f_1(x), f_2(x)), where f_1(x) = cos\left(\frac{\pi}{2}x_1\right)(1 + g(x)), f_2(x) = sin\left(\frac{\pi}{2}x_1\right)(1 + g(x))$$

$$and, g(x) = 100\left[1 + \sum_{i=3}^n (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))\right],$$

Constraints: $0 \le x_i \le 1, n = 12, i = 1, 2, ..., 12$

B.4 DTLZ 4 with concave POF

$$F = (f_1(x), f_2(x)), where f_1(x) = cos\left(\frac{\pi}{2}x_1^a\right)(1 + g(x)), f_2(x) = sin\left(\frac{\pi}{2}x_1^a\right)(1 + g(x))$$

$$and, g(x) = \sum_{i=3}^n (x_i - 0.5)^2, \text{ Constraints: } 0 \le x_i \le 1, n = 12, \alpha = 100, i = 1, 2, ..., 1$$

B.5 DTLZ 5with curve POF

$$F = (f_1(x), f_2(x)), where f_1(x) = cos\left(\frac{\pi}{2}\theta_1\right)(1 + g(x)), f_2(x) = sin\left(\frac{\pi}{2}\theta_1\right)(1 + g(x))$$

$$and, \theta_1 = x_1.\left(\frac{\pi}{2}\right), g(x) = \sum_{i=3}^n (x_i - 0.5)^2,$$

Constraints: $0 \le x_i \le 1, n = 12, i = 1, 2, ..., 12$

B.6 DTLZ 6with curve POF

$$F = (f_1(x), f_2(x)), where f_1(x) = cos\left(\frac{\pi}{2}\theta_1\right)(1 + g(x)), f_2(x) = sin\left(\frac{\pi}{2}\theta_1\right)(1 + g(x))$$

$and, \theta_1 = x_1.\left(\frac{\pi}{2}\right), g(x) = \sum_{i=3}^{n}(x_i)^{0.1},$ Constraints: $0 \leq x_i \leq 1, n = 12, i = 1, 2, ..., 12$

B.7 DTLZ 7 with disconnected POF

$$F = (f_1(x), f_2(x)), where f_1(x) = x_1, f_2(x) = (1 + g(x)).h(f_1, f_2, g(x))$$

$$and, g(x) = 1 + \frac{9}{22} \sum_{i=3}^{n}(x_i), h(f_1, f_2, g(x)) = 2 - \sum_{i=1}^{2} \left(\frac{f_i}{1+g}(1 + \sin(3\pi f_i))\right)$$

Constraints: $0 \leq x_i \leq 1, n = 22, i = 1, 2, ..., 12$

Appendix C. Unconstrained multi-objective DTLZ with 3-Dim [45] test problems utilized in this work

C.1 DTLZ 1 with linear POF

$$F = (f_1(x), f_2(x), f_3(x)), where f_1(x) = \frac{1}{2}x_1 x_2(1 + g(x)), f_2(x) = \frac{1}{2}x_1(1 - x_2)(1 + g(x))$$

$$f_3(x) = \frac{1}{2}(1 - x_1)(1 + g(x)) and, g(x) = 100 \left[10 + \sum_{i=3}^{n}(x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))\right],$$

Constraints: $0 \leq x_i \leq 1, n = 12, i = 1, 2, ..., 12$

C.2 DTLZ 2 with concave POF

$$F = (f_1(x), f_2(x), f_3(x)), where f_1(x) = \cos\left(\frac{\pi}{2}x_1\right)\cos\left(\frac{\pi}{2}x_2\right)(1 + g(x)),$$

$$f_2(x) = \cos\left(\frac{\pi}{2}x_1\right)\sin\left(\frac{\pi}{2}x_2\right)(1 + g(x)), f_3(x) = \sin\left(\frac{\pi}{2}x_1\right)(1 + g(x))$$

$and, g(x) = \sum_{i=3}^{n}(x_i - 0.5)^2,$ Constraints: $0 \leq x_i \leq 1, n = 12, i = 1, 2, ..., 12$

C.3 DTLZ 3 with concave POF

$$F = (f_1(x), f_2(x), f_3(x)), where f_1(x) = \cos\left(\frac{\pi}{2}x_1\right)\cos\left(\frac{\pi}{2}x_2\right)(1 + g(x)),$$

$$f_2(x) = \cos\left(\frac{\pi}{2}x_1\right)\sin\left(\frac{\pi}{2}x_2\right)(1 + g(x)), f_3(x) = \sin\left(\frac{\pi}{2}x_1\right)(1 + g(x))$$

$$and, g(x) = 100 \left[10 + \sum_{i=3}^{n}(x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))\right],$$

Constraints: $0 \leq x_i \leq 1, n = 12, i = 1, 2, ..., 12$

C.4 DTLZ 4 with concave POF

$$F = (f_1(x), f_2(x), f_3(x)), where f_1(x) = \cos\left(\frac{\pi}{2}x_1^\alpha\right)\cos\left(\frac{\pi}{2}x_2^\alpha\right)(1 + g(x)),$$

$$f_2(x) = \cos\left(\frac{\pi}{2}x_1^\alpha\right)\sin\left(\frac{\pi}{2}x_2^\alpha\right)(1 + g(x)), f_3(x) = \sin\left(\frac{\pi}{2}x_1^\alpha\right)(1 + g(x))$$

$and, g(x) = \sum_{i=3}^{n}(x_i - 0.5)^2,$ Constraints: $0 \leq x_i \leq 1, n = 12, \alpha = 100, i = 1, 2, ..., 12$

C.5 DTLZ 5 with curve POF

$$F = (f_1(x), f_2(x), f_3(x)), where f_1(x) = \cos\left(\frac{\pi}{2}\theta_1\right)\cos\left(\frac{\pi}{2}\theta_2\right)(1 + g(x)),$$

$$f_2(x) = cos\left(\frac{\pi}{2}\theta_1\right)sin\left(\frac{\pi}{2}\theta_2\right)(1+g(x)), f_3(x) = sin\left(\frac{\pi}{2}\theta_1\right)(1+g(x))$$

$$and, \theta_1 = x_1.\left(\frac{\pi}{2}\right), \theta_2 = \frac{\pi}{4.(1+g(x))}.(1+2x_2.g(x)), g(x) = \sum_{i=3}^{n}(x_i - 0.5)^2,$$

Constraints: $0 \leq x_i \leq 1, n = 12, i = 1, 2, ..., 12$

C.6 DTLZ 6 with curve POF

$$F = (f_1(x), f_2(x), f_3(x)), where f_1(x) = cos\left(\frac{\pi}{2}\theta_1\right)cos\left(\frac{\pi}{2}\theta_2\right)(1+g(x)),$$

$$f_2(x) = cos\left(\frac{\pi}{2}\theta_1\right)sin\left(\frac{\pi}{2}\theta_2\right)(1+g(x)), f_3(x) = sin\left(\frac{\pi}{2}\theta_1\right)(1+g(x))$$

$$and, \theta_1 = x_1.\left(\frac{\pi}{2}\right), \theta_2 = \frac{\pi}{4.(1+g(x))}.(1+2x_i.g(x)), g(x) = \sum_{i=3}^{n}(x_i)^{0.1},$$

Constraints: $0 \leq x_i \leq 1, n = 12, i = 1, 2, ..., 12$

C.7 DTLZ 7 with disconnected POF

$$F = (f_1(x), f_2(x), f_3(x)), where f_1(x) = x_1, f_2(x) = x_2, f_3(x) = (1+g(x)).h(f_1, f_2, g(x))$$

$$and, g(x) = 1 + \frac{9}{22}\sum_{i=3}^{n}(x_i), h(f_1, f_2, g(x)) = 3 - \sum_{i=1}^{2}\left(\frac{f_i}{1+g}(1+\sin(3\pi f_i))\right)$$

Constraints: $0 \leq x_i \leq 1, n = 22, i = 1, 2, ..., 12$

Appendix D. Constrained multi-objective test problems [57,58] utilized in this work

D.1 CONSTR with convex POF

$$F = (f_1(x, y), f_2(x, y)), where f_1(x, y) = x, f_2(x, y) = (1+y)/x,$$

Constraints: $0.1 \leq x \leq 1, 0 \leq y \leq 5, 0 \geq 6 - y - 9x, 0 \geq 1 + y - 9x$

D.2 TNK with dis-connected and convoluted POF

$$Minimize\ F = (f_1(x, y), f_2(x, y)), where f_1(x, y, z) = x, f_2(x, y, z) = y,$$

Constraints: $0 \leq x, y \leq \pi, 0 \geq -x^2 - y^2 + 1 + 0.1 * cos\left(16\ arctan\frac{x}{y}\right), \frac{1}{2} \geq \left(x - \frac{1}{2}\right)^2 + \left(y - \frac{1}{2}\right)^2,$

D.3 SRN with connected POF

$$F = (f_1(x, y), f_2(x, y)), where f_1(x, y) = (x - 2)^2 + (y - 1)^2 + 2, f_2(x, y) = 9x - (y - 1)^2,$$

Constraints: $-20 \leq x, y \leq 20, 0 \geq x^2 + y^2 - 225, 0 \geq x - 3y + 10$

D.4 OSY with discrete POF

$$F = (f_1(x), f_2(x)), where f_1(x) = -\left(25(x_1 - 2)^2 + (x_2 - 2)^2\right) + (x_3 - 1)^2 + (x_4 - 4)^2 + (x_5 - 1)^2,$$

$$f_2(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 + x_6^2,$$

Constraints: $0 \leq x_1, x_2, x_6 \leq 10, 1 \leq x_3, x_5 \leq 5, 0 \leq x_4 \leq 6, 0 \leq x_1 + x_2 - 2, 0 \leq 6 - x_1 - x_2,$

$$0 \leq 2 - x_2 + x_1, 0 \leq 2 - x_1 + 3x_1, 0 \leq 4 - (x_3 - 3)^2 - x_4, 0 \leq (x_5 - 3)^2 + x_6 - 4,$$

D.5 BNH with convex POF

$$F = (f_1(x,y), f_2(x,y)), where f_1(x,y) = 4x^2 + 4y^2, f_2(x,y) = (x-5)^2 + (y-5)^2,$$

Constraints: $0 \leq x \leq 5, 0 \leq y \leq 3, 0 \geq (x-5)^2 + (y)^2 - 25, 0 \geq -(x-8)^2 + (y+3)^2 + 7.7$

D.6 KITA with discrete concave POF

$$Maximize\ F = (f_1(x,y), f_2(x,y)), where f_1(x,y) = -x^2 + y, f_2(x,y) = \left(\frac{1}{2}\right)x + y + 1,$$

Constraints: $x, y \geq 0, 0 \geq (1/6)x + y - (13/2), 0 \geq (1/2)x + y - (15/2), 0 \geq 5x + y - 30,$

Appendix E. Constrained multi-objective engineering problems used in this work

E.1 Brushless DC Wheel Motor design problem (RWMOP1)

Brushless DC wheel motor design problem is a constrained multi-objective problem in the area of electrical machine design [43]. The objectives are in conflict, minimize the total weight ($f1$) and efficiency ($f2$) should be maximized and there are five design variables: stator diameter (Ds), magnetic induction in the air gap (Be), current density in the conductors (δ), magnetic induction in the teeth (Bd) and magnetic induction in the stator back iron (Bcs) as well as five constraints. The pareto optimal front have non-convex and connected in nature.

$$f_1(x) = Max\ \eta, f_2(x) = Min\ M_{tot}$$

$$Where: G(x) = \begin{matrix} D_{ext} \leq 340mm \\ D_{int} \geq 76mm, I_{max} \geq 125A \\ T_a \leq 120°, discr \geq 0 \end{matrix}$$

$$150mm \leq D_s \leq 330mm, 0.5T \leq B_e \leq 0.76T, 2A/mm^2 \leq \varsigma \leq 5A/mm^2, 0.9T \leq B_d \leq 1.8T$$
$$0.6T \leq B_{cs} \leq 1.6T$$

E.2 Isolated Safety Transformer design problem (RWMOP2)

The Isolated Safety Transformer design problem is a well-known problem in the electrical optimization field [43], in which minimize the total mass ($f1$)and maximize the efficiency ($f2$). As can be seen in the following equations, there are seven discrete design variables ($x1$-$x7$), three variables ($x1, x2, x3$) for the shape of the lamination, one for the frame *(x4),* two for the section of enameled wires($x5, x6$), one for the number of primary turn($x7$).The pareto optimal front have convex and connected in nature.

$$f1 = Max\ \eta, f2 = Min\ M_{tot}$$

$$, G(x) = \begin{matrix} T_{cond} \leq 120°C, T_{iron} \leq 100°C \\ \dfrac{\Delta V_2}{V_{20}} \leq 0.1, \dfrac{I_{10}}{I_1} \leq 0.1 \\ f_2 \leq 1, f_1 \leq 1 \\ residue < 10^{-6} \end{matrix}$$

$$3mm \leq a \leq 30mm, 14mm \leq b \leq 95mm, 6mm \leq c \leq 40mm, 10mm \leq d \leq 80mm$$
$$200 \leq n_1 \leq 1200, 0.15mm^2 \leq S_1 \leq 19mm^2, 0.15mm^2 \leq S_2 \leq 19mm^2$$

E.3 Helical Spring design problem (RWMOP3)

The Helical Spring design mixed variables optimization problem first proposed by Kannan and Kramer a well-known problem in the mechanical engineering field [43], in which stress ($f1$) and volume ($f2$) of a Helical Spring problem should be minimized. As can be seen in the following equations, there are three design variables (*x1-x3*), spring coils (*x1: an integer variable*), wire diameter (*x2: a*

discrete variable), and spring diameter (*x3*) as well as eight constraints. The pareto optimal front have convex and multi-connected in nature.

$$Pmax = 1000; C = \frac{x3}{x2}; K = \left(\frac{4*C-1}{4*C-4}\right) + \left(0.615 * \frac{x2}{x3}\right); \ G = 11500000; Vmax = 30; dmin = 0.2; S = 189000; dpm = 6; lmax = 14; dw$$

$$= 1.25; Pmax = 1000; Dmax = 3; P = 300; C = \frac{x3}{x2}; k = \frac{G*x2^4}{8*x1*(x3^3)};$$

$$dp = \frac{P}{k}; K = \left(\frac{4*C-1}{4*C-4}\right) + \left(0.615 * \frac{x2}{x3}\right);$$

$$f1 = \left(0.25 * pi^2 * (x2^2) * x3 * (x1+2)\right); f2 = \left(8 * K * Pmax * \frac{x3}{pi*(x2^3)}\right);$$

$$g(1) = -\left(lmax - \left(\frac{Pmax}{k}\right) - (1.05 * x2 * (x1+2))\right); g(2) = -(x2 - dmin);$$

$$g(3) = -(Dmax - (x2+x3)); g(4) = -(C-3); g(5) = -(dpm - dp); g(6) = -\left(\left(\frac{Pmax-P}{k}\right) - dw\right); g(7) = -\left(S - \left(8 * K\right.\right.$$

$$\left.\left. * Pmax * \frac{x3}{pi*(x2^3)}\right)\right); g(8) = -\left(Vmax - \left(0.25 * pi^2 * (x2^2) * x3 * (x1+2)\right)\right);$$

$$x2 = [0.009 \ 0.0095 \ 0.0104 \ 0.0118 \ 0.0128 \ 0.0132 \ 0.014 \ 0.015 \ 0.0162 \ldots$$

$$0.0173 \ 0.018 \ \ 0.020 \ \ 0.023 \ \ 0.025 \ 0.028 \ 0.032 \ \ 0.035 \ \ 0.041 \ \ 0.047 \ \ 0.054\ldots$$

$$0.063 \ 0.072 \ 0.080 \ \ 0.092 \ \ 0.105 \ \ 0.120 \ \ 0.135 \ \ 0.148 \ 0.162 \ 0.177 \ \ 0.192 \ \ 0.207\ldots$$

$$0.225 \ \ 0.244 \ \ 0.263 \ 0.283 \ 0.307 \ \ 0.331 \ \ 0.362 \ \ 0.394 \ \ 0.4375 \ 0.500];$$

$$x1 = [1 \ 2 \ 3 \ldots 30 \ 31 \ 32]; 0 < x3 \le 3; dim = 3;$$

E.4 Two bar truss design problem (RWMOP4)

The 2-bar truss design problem is a well-known problem in the structural optimization field, in which structural volume (*f1*) and maximum stresses (*f2*) exerted on bars are minimized without elastic failure due to 100 kN load. Single-objective optimization problem of 2-bar truss is founded in Kirsch (1981). Later reformulated by Deb as a multi-objective optimization problem [43].As can be seen in the following equations, there are two design variables (*x1-x2*) related to cross section area of truss bar members as well as single constraint. The pareto optimal front have convex and connected in nature.

$$sigma_{AC} = \frac{20 * sqrt(16 + x3^2)}{x1 * x3}; sigma_{BC} = \frac{80 * sqrt(1 + x3^2)}{x2 * x3};$$

$$f1 = \left(x1 * sqrt(16 + x3^2)\right) + \left(x2 * sqrt(1 + x3^2)\right); f2 = \max(sigma_{AC}, sigma_{BC});$$

$$g(1) = \max(sigma_{AC}, sigma_{BC}) - 100000; dim = 3; 0 < x1, x2 \le 0.01; \quad 1 < x3 \le 3$$

E.5 Welded beam design problem (RWMOP5)

The welded beam design problem first proposed by Deb [43]. The fabrication cost (*f1*) and deflection of the beam (*f2*) of a welded beam should be minimized in this problem. There are four design variables: the thickness of the weld (*x1*), the length of the clamped bar (*x2*), the height of the bar (*x3*) and the thickness of the bar (*x4*) as well as four constraints. The pareto optimal front have convex and connected in nature.

$$f_1(x) = 1.10471 * x(1)\hat{\ }2 * x(2) + 0.04811 * x(3) * x(4) * (14.0 + x(2))$$

$$f_2(x) = 65856000 / (30 * \widehat{106} * x(4) * x(3)\hat{\ }3)$$

$$g_1(x) = tau - 13600, g_2(x) = sigma - 30000, g_3(x) = x(1) - x(4), g_4(x) = 6000 - P$$

$0.125 \leq x_1 \leq 5, 0.1 \leq x_2 \leq 10, 0.1 \leq x_3 \leq 10, 0.125 \leq x_4 \leq 5$ Where $Q = 6000 * \left(14 + \frac{x(2)}{2}\right)$;

$$D = sqrt\left(\frac{x(2)^2}{4} + \frac{(x(1) + x(3))^2}{4}\right), J = 2 * \left(x(1) * x(2) * sqrt(2) * \left(\frac{x(2)^2}{12} + \frac{(x(1) + x(3))^2}{4}\right)\right)$$

$$alpha = \frac{6000}{sqrt(2) * x(1) * x(2)}, beta = Q * \frac{D}{J}, P = tmpf * sqrt\left(x(3)^2 * \frac{x(4)^6}{36}\right) * \left(1 - x(3) * \frac{sqrt\left(\frac{30}{48}\right)}{28}\right)$$

$$tau = sqrt\left(alpha^2 + 2 * alpha * beta * \frac{x(2)}{2 * D} + beta^2\right), sigma = \frac{504000}{x(4) * x(3)^2}, tmpf = 4.013 * \frac{30 * 10^6}{196}$$

References

[1] J.H. Holland, J.S. Reitman, Cognitive systems based on adaptive algorithms, ACM SIGART Bull. (63) (1977), https://doi.org/10.1145/1045343.1045373, 49–49.

[2] C.-F. Tsai, W. Eberle, C.-Y. Chu, Genetic algorithms in feature and instance selection, Knowl. Base Syst. 39 (2013) 240–247, https://doi.org/10.1016/j.knosys.2012.11.005.

[3] M.-H. Lin, J.-F. Tsai, C.-S. Yu, A review of deterministic optimization methods in engineering and management, Math. Probl Eng. 2012 (2012) 1–15, https://doi.org/10.1155/2012/756023.

[4] D.B. Shmoys, C. Swamy, Stochastic optimization is (almost) as easy as deterministic optimization, in: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science, 2004, pp. 228–237, https://doi.org/10.1109/FOCS.2004.62, 2004.

[5] M. Dorigo, M. Birattari, T. Stutzle, Ant colony optimization, IEEE Comput. Intell. Mag. 1 (2006) 28–39.

[6] J. Kennedy, Particle swarm optimization, in: Encyclopedia of Machine Learning, EDN. Springer, 2011, pp. 760–766.

[7] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (4) (1997) 341–359, https://doi.org/10.1023/A:1008202821328.

[8] C. Zhang, L. Zhou, Y. Li, Pareto optimal reconfiguration planning and distributed parallel motion control of mobile modular robots, IEEE Trans. Ind. Electron. 1 (2023), https://doi.org/10.1109/TIE.2023.3321997. –10.

[9] B. Zhu, Y. Sun, J. Zhao, J. Han, P. Zhang, T. Fan, A critical scenario search method for intelligent vehicle testing based on the social cognitive optimization algorithm, IEEE Trans. Intell. Transport. Syst. 24 (8) (2023) 7974–7986, https://doi.org/10.1109/TITS.2023.3268324.

[10] B. Cao, J. Zhao, P. Yang, Y. Gu, K. Muhammad, J.J.P.C. Rodrigues, V.H.C. de Albuquerque, Multiobjective 3-D topology optimization of next-generation wireless data center network, IEEE Trans. Ind. Inf. 16 (5) (2020) 3597–3605, https://doi.org/10.1109/TII.2019.2952565.

[11] L. Zhang, C. Sun, G. Cai, L.H. Koh, Charging and discharging optimization strategy for electric vehicles considering elasticity demand response, eTransportation 18 (2023) 100262, https://doi.org/10.1016/j.etran.2023.100262.

[12] B. Cao, Z. Li, X. Liu, Z. Lv, H. He, Mobility-aware multiobjective task offloading for vehicular edge computing in digital twin environment, IEEE J. Sel. Area. Commun. 41 (10) (2023) 3046–3055, https://doi.org/10.1109/JSAC.2023.3310100.

[13] B. Cao, X. Wang, W. Zhang, H. Song, Z. Lv, A many-objective optimization model of industrial Internet of things based on private blockchain, IEEE Network 34 (5) (2020) 78–83, https://doi.org/10.1109/MNET.011.1900536.

[14] I. BoussäiD, J. Lepagnot, P. Siarry, A survey on optimization metaheuristics, Inf. Sci. 237 (2013) 82–117, https://doi.org/10.1016/j.ins.2013.02.041.

[15] M. Helbig, A.P. Engelbrecht, Performance measures for dynamic multi-objective optimisation algorithms, Inf. Sci. 250 (2013) 61–81, https://doi.org/10.1016/j.ins.2013.06.051.

[16] H.-G. Beyer, B. Sendhoff, Robust optimization–a comprehensive survey, Comput. Methods Appl. Mech. Eng. 196 (33–34) (2007) 3190–3218, https://doi.org/10.1016/j.cma.2007.03.003.

[17] C.A.C. Coello Coello, A comprehensive survey of evolutionarybased multiobjective optimization techniques, Knowl. Inf. Syst. 1 (3) (1999) 269–308, https://doi.org/10.1007/BF03325101.

[18] K. Deb, Multi-objective Optimization Using Evolutionary Algorithms, vol. 16, Wiley, 2001.

[19] C. von Lucken, Bar, B. an, C. Brizuela, A survey on multi-" objective evolutionary algorithms for many-objective problems, Comput. Optim. Appl. 58 (2014) 707–756.

[20] T.T. Nguyen, S. Yang, J. Branke, Evolutionary dynamic optimization: a survey of the state of the art, Swarm Evol. Comput. 6 (2012) 1–24, https://doi.org/10.1016/j.swevo.2012.05.001.

[21] N. Padhye, P. Mittal, K. Deb, Feasibility preserving constraint-handling strategies for real parameter evolutionary optimization, Comput. Optim. Appl. 62 (3) (2015) 851–890, https://doi.org/10.1007/s10589-015-9752-6.

[22] W. Kaidi, M. Khishe, M. Mohammadi, Dynamic levy flight chimp optimization, Knowl. Base Syst. 235 (2022) 107625, https://doi.org/10.1016/j.knosys.2021.107625.

[23] M. Khishe, M. Nezhadshahbodaghi, M.R. Mosavi, D. Martín, A weighted chimp optimization algorithm, IEEE Access 9 (2021) 158508–158539, https://doi.org/10.1109/ACCESS.2021.3130933.

[24] M. Khishe, N. Orouji, M.R. Mosavi, Multi-objective chimp optimizer: an innovative algorithm for multi-objective problems, Expert Syst. Appl. 211 (2023) 118734, https://doi.org/10.1016/j.eswa.2022.118734.

[25] M. Khishe, M.R. Mosavi, Chimp optimization algorithm, Expert Syst. Appl. 149 (2020) 113338, https://doi.org/10.1016/j.eswa.2020.113338.

[26] Q. Bo, W. Cheng, M. Khishe, Evolving chimp optimization algorithm by weighted opposition-based technique and greedy search for multimodal engineering problems, Appl. Soft Comput. 132 (2023) 109869, https://doi.org/10.1016/j.asoc.2022.109869.

[27] M. Khishe, Greedy opposition-based learning for chimp optimization algorithm, Artif. Intell. Rev. 56 (8) (2023) 7633–7663, https://doi.org/10.1007/s10462-022-10343-w.

[28] A. Asrari, S. Lotfifard, M.S. Payam, Pareto dominancebased multiobjective optimization method for distribution network reconfiguration, IEEE Trans. Smart Grid 7 (3) (2016) 1401–1410, https://doi.org/10.1109/TSG.2015.2468683.

[29] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P.N. Suganthan, Q. Zhang, Multiobjective evolutionary algorithms: a survey of the state of the art, Swarm Evol. Comput. 1 (1) (2011) 32–49, https://doi.org/10.1016/j.swevo.2011.03.001, 3//.

[30] K. Deb, Multi-objective optimization, in: Search Methodologies, 449, Springer, 2014, p. 403. *EDN*.
[31] N. Padhye, K. Deb, Evolutionary multi-objective optimization and decision making for selective laser sintering, in: Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, 2010, pp. 1259–1266, https://doi.org/10.1145/1830483.1830709.
[32] F.S. Gharehchopogh, B. Abdollahzadeh, S. Barshandeh, B. Arasteh, A multi-objective mutation-based dynamic Harris Hawks optimization for botnet detection in IoT, Int. Things 24 (2023) 100952, https://doi.org/10.1016/j.iot.2023.100952.
[33] F.S. Gharehchopogh, T. Ibrikci, An improved African vultures optimization algorithm using different fitness functions for multi-level thresholding image segmentation, Multimed. Tool. Appl. (2023) 1–47, https://doi.org/10.1007/s11042-023-16300-1.
[34] F.S. Gharehchopogh, An improved Harris Hawks optimization algorithm with multi-strategy for community detection in social networks, JBE 20 (3) (2023) 1175–1197, https://doi.org/10.1007/s42235-022-00303-z.
[35] F.S. Gharehchopogh, A. Ucan, T. Ibrikci, B. Arasteh, G. Isik, Slime mould algorithm: a comprehensive survey of its variants and applications, Arch. Comput. Methods Eng.: State of the Art Reviews 30 (4) (2023) 2683–2723, https://doi.org/10.1007/s11831-023-09883-3.
[36] J. Piri, P. Mohapatra, B. Acharya, F.S. Gharehchopogh, V.C. Gerogiannis, A. Kanavos, S. Manika, Feature selection using artificial gorilla troop optimization for biomedical data: a case analysis with COVID-19 data, Mathematics 10 (15) (2022) 2742, https://doi.org/10.3390/math10152742.
[37] Z. Xiao, J. Shu, H. Jiang, J.C.S. Lui, G. Min, J. Liu, S. Dustdar, Multi-objective parallel task offloading and content caching in D2D-aided MEC networks, IEEE Trans. Mobile Comput. (2022) 1–16, https://doi.org/10.1109/TMC.2022.3199876.
[38] B. Cao, J. Zhao, Y. Gu, Y. Ling, X. Ma, Applying graph-based differential grouping for multiobjective large-scale optimization, Swarm Evol. Comput. 53 (2020) 100626, https://doi.org/10.1016/j.swevo.2019.100626.
[39] Y. Duan, Y. Zhao, J. Hu, An initialization-free distributed algorithm for dynamic economic dispatch problems in microgrid: modeling, optimization and analysis, Sustain. Energy, Grids and Networks 34 (2023) 101004, https://doi.org/10.1016/j.segan.2023.101004.
[40] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197, https://doi.org/10.1109/4235.996017.
[41] C.A. Coello Coello, M.S. Lechuga, MOPSO: a proposal for multiple objective particle swarm optimization, 2002, in: Proceedings of the 2002 Congress on Evolutionary Computation, 2002, pp. 1051–1056, https://doi.org/10.1109/CEC.2002.1004388. CEC.
[42] I. Alaya, C. Solnon, K. Ghedira, Ant colony optimization for multi-objective optimization problems, ICTAI 1 (2007) 450–457, https://doi.org/10.1109/ICTAI.2007.108.
[43] F. Xue, A.C. Sanderson, R.J. Graves, Pareto-based multiobjective differential evolution, 2003, in: The 2003 Congress on Evolutionary Computation, 2003, pp. 862–869. CEC'03.
[44] J.D. Knowles, D.W. Corne, Approximating the nondominated front using the Pareto archived evolution strategy, Evol. Comput. 8 (2) (2000) 149–172, https://doi.org/10.1162/106365600568167.
[45] D. Wolpert, No free lunch theorem for optimization, IEEE Trans. Evol. Comput. (1997) 467–482.
[46] Y. Jin, M. Olhofer, B. Sendhoff, Dynamic weighted aggregation for evolutionary multi-objective optimization, Why does it work and how? (2001).
[47] J. Branke, K. Deb, H. Dierolf, M. Osswald, Finding knees in multi-objective optimization, in: International Conference on Parallel Problem Solving from Nature, 2004, pp. 722–731.
[48] J.B. Kollat, P. Reed, A framework for visually interactive decision-making and design using evolutionary multiobjective optimization [Video], Environ. Model. Software 22 (12) (2007) 1691–1704, https://doi.org/10.1016/j.envsoft.2007.02.001.
[49] S. Mirjalili, P. Jangir, S. Saremi, Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems, Appl. Intell. 46 (1) (2017) 79–95, https://doi.org/10.1007/s10489-016-0825-8.
[50] M. Premkumar, P. Jangir, R. Sowmya, H.H. Alhelou, S. Mirjalili, B.S. Kumar, Multi-objective equilibrium optimizer: framework and development for solving multi-objective optimization problems, J. Comput. Design and Eng. 9 (1) (2021) 24–50, https://doi.org/10.1093/jcde/qwab065.
[51] M. Premkumar, P. Jangir, R. Sowmya, H.H. Alhelou, A.A. Heidari, H. Chen, MOSMA: multi-objective slime mould algorithm based on elitist non-dominated sorting, IEEE Access 9 (2020) 3229–3248, https://doi.org/10.1109/ACCESS.2020.3047936.
[52] M. Premkumar, P. Jangir, B. Santhosh Kumar, R. Sowmya, H. Haes Alhelou, L. Abualigah, A.R. Yildiz, S. Mirjalili, A new arithmetic optimization algorithm for solving real-world multiobjective CEC-2021 constrained optimiza- tion problems: diversity analysis and validations, IEEE Access 9 (2021) 84263–84295.
[53] H. Buch, I.N. Trivedi, A new non-dominated sorting ions motion algorithm: development and applications, Deci- SionSci, Letture 9 (1) (2020) 59–76.
[54] Q. Zhang, H. Li, MOEA/D: a multi-objective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (6) (2007) 712–731, https://doi.org/10.1109/TEVC.2007.892759.
[55] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197, https://doi.org/10.1109/4235.996017.
[56] S. Mirjalili, P. Jangir, S.Z. Mirjalili, S. Saremi, I.N. Trivedi, Optimization of problems with multiple objectives using the multi-verse optimization algorithm, Knowl. Base Syst. 134 (2017) 50–71, https://doi.org/10.1016/j.knosys.2017.07.018.
[57] P. Jangir, N. Jangir, 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, Eng. Appl. Artif. Intell. 72 (2018) 449–467, https://doi.org/10.1016/j.engappai.2018.04.018.
[58] M. Premkumar, P. Jangir, R. Sowmya, MOGBO: a new Multiobjective Gradient-Based Optimizer for real-world structural optimization problems, Knowl. Base Syst. 218 (2021) 106856, https://doi.org/10.1016/j.knosys.2021.106856.
[59] S. Kumar, P. Jangir, G.G. Tejani, M. Premkumar, MOPGO: a new physics-based multi-objective plasma generation optimizer for solving structural optimization problems, IEEE Access 9 (2021) 84982–85016, https://doi.org/10.1109/ACCESS.2021.3087739.
[60] P. Jangir, A.A. Heidari, H. Chen, Elitist non-dominated sorting Harris hawks optimization: framework and developments for multi-objective problems, Expert Syst. Appl. 186 (2021) 115747, https://doi.org/10.1016/j.eswa.2021.115747.
[61] S. Kumar, P. Jangir, G.G. Tejani, M. Premkumar, MOTEO: a novel physics-based multiobjective thermal exchange optimization algorithm to design truss structures, Knowl. Base Syst. 242 (2022) 108422, https://doi.org/10.1016/j.knosys.2022.108422.
[62] S. Kumar, P. Jangir, G.G. Tejani, M. Premkumar, A decomposition based multi-objective heat transfer search algorithm for structure optimization, Knowl. Base Syst. 253 (2022) 109591, https://doi.org/10.1016/j.knosys.2022.109591.
[63] N. Ganesh, R. Shankar, K. Kalita, P. Jangir, D. Oliva, M. Pérez-Cisneros, A novel decomposition-based multi-objective symbiotic organism search optimization algorithm, Mathematics 11 (8) (2023) 1898, https://doi.org/10.3390/math11081898.
[64] S.B. Pandya, J. Visumathi, M. Mahdal, T.K. Mahanta, P. Jangir, A novel MOGNDO algorithm for security-constrained optimal power flow problems, Electronics 11 (22) (2022) 3825, https://doi.org/10.3390/electronics11223825.
[65] P. Jangir, Non-dominated sorting moth flame optimizer: a novel multi-objective optimization algorithm for solving engineering design problems, Eng. Technol. open Access J. 2 (1) (2018) 17–31.
[66] P. Jangir, N. Jangir, Non-dominated sorting whale optimization algorithm, Global J. Res, Eng. 17 (4) (2017) 15–42.
[67] P. Jangir, 'MONSDA:-A novel multi-objective non-dominated sorting dragonfly algorithm, Global J. Res. Eng.: Front. Electr. Electron. Eng. 20 (2020) 28–52.
[68] T.T. Binh, U. Korn, MOBES: a multiobjective evolution strategy for constrained optimization problems, Third Int. Conf. Genetic Algorithms Mendel 97 (1997) 27.
[69] A. Osyczka, S. Kundu, A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm, Struct. Optim. 10 (2) (1995) 94–99, https://doi.org/10.1007/BF01743536.
[70] J. Zheng, Z. Zhang, J. Zou, S. Yang, J. Ou, Y. Hu, A dynamic multiobjective particle swarm optimization algorithm based on adversarial decomposition and neighborhood evolution, Swarm Evol. Comput. 69 (2022) 100987, https://doi.org/10.1016/j.swevo.2021.100987.
[71] A. Ben-Said, A. Moukrim, R.N. Guibadj, J. Verny, Using decompositionbased multi-objective algorithm to solve selective pickup and delivery problems with time windows, Comput. Oper. Res. 145 (2022) 105867, https://doi.org/10.1016/j.cor.2022.105867.

[72] D. Zouache, F.B. Abdelaziz, Guided manta ray foraging optimization using epsilon dominance for multi-objective optimization in engineering design, Expert Syst. Appl. 189 (2022) 116126, https://doi.org/10.1016/j.eswa.2021.116126.

[73] S. Yin, Q. Luo, Y. Zhou, IBMSMA: an indicator-based multi-swarm slime mould algorithm for multi-objective truss optimization problems, JBE (2022) 1–28.

[74] S. Li, H. Chen, Y. Chen, Y. Xiong, Z. Song, Hybrid method with parallel-factor theory, a support vector machine, and particle filter optimization for intelligent machinery failure identification, Machines 11 (8) (2023) 837, https://doi.org/10.3390/machines11080837.

[75] E.H. Houssein, D. Oliva, N.A. Samee, N.F. Mahmoud, M.M. Emam, Liver Cancer Algorithm: a novel bio-inspired optimizer, Comput. Biol. Med. 165 (2023) 107389, https://doi.org/10.1016/j.compbiomed.2023.107389. ISSN 0010-4825.