



Liver Cancer Algorithm: A novel bio-inspired optimizer

Essam H. Houssein ^{a,*}, Diego Oliva ^b, Nagwan Abdel Samee ^c, Noha F. Mahmoud ^d, Marwa M. Emam ^a

^a Faculty of Computers and Information, Minia University, Minia, Egypt

^b Depto. Innovación Basada en la Información y el Conocimiento, Universidad de Guadalajara, CUCEI, Guadalajara, Jal, Mexico

^c Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia

^d Rehabilitation Sciences Department, Health and Rehabilitation Sciences College, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia

ARTICLE INFO

Keywords:

Liver Cancer Algorithm (LCA)
Metaheuristic algorithms (MAs)
Feature selection (FS)
Optimization
Bio-inspired
Random opposition-based learning (ROBL)

ABSTRACT

This paper introduces a new bio-inspired optimization algorithm named the Liver Cancer Algorithm (LCA), which mimics the liver tumor growth and takeover process. It uses an evolutionary search approach that simulates the behavior of liver tumors when taking over the liver organ. The tumor's ability to replicate and spread to other organs inspires the algorithm. LCA algorithm is developed using genetic operators and a Random Opposition-Based Learning (ROBL) strategy to efficiently balance local and global searches and explore the search space. The algorithm's efficiency is tested on the IEEE Congress of Evolutionary Computation in 2020 (CEC'2020) benchmark functions and compared to seven widely used metaheuristic algorithms, including Genetic Algorithm (GA), particle swarm optimization (PSO), Differential Evolution (DE), Adaptive Guided Differential Evolution Algorithm (AGDE), Improved Multi-Operator Differential Evolution (IMODE), Harris Hawks Optimization (HHO), Runge–Kutta Optimization Algorithm (RUN), weighted mean of vectors (INFO), and Coronavirus Herd Immunity Optimizer (CHIO). The statistical results of the convergence curve, boxplot, parameter space, and qualitative metrics show that the LCA algorithm performs competitively compared to well-known algorithms. Moreover, the versatility of the LCA algorithm extends beyond mathematical benchmark problems. It was also successfully applied to tackle the feature selection problem and optimize the support vector machine for various biomedical data classifications, resulting in the creation of the LCA-SVM model. The LCA-SVM model was evaluated in a total of twelve datasets, among which the Monoamine Oxidase (MAO) dataset stood out, showing the highest performance compared to the other datasets. In particular, the LCA-SVM model achieved an impressive accuracy of 98.704% on the MAO dataset. This outstanding result demonstrates the efficacy and potential of the LCA-SVM approach in handling complex datasets and producing highly accurate predictions. The experimental results indicate that the LCA algorithm surpasses other methods to solve mathematical benchmark problems and feature selection.

1. Introduction

An optimization problem involves finding the optimal solution among multiple solutions. It comprises three primary elements: decision variables, constraints, and an objective function. As science and technology advance, optimization problems have become more complex, bringing about new challenges that demand appropriate optimization tools [1]. Optimization methods can be classified into deterministic and stochastic approaches. Deterministic approaches, further separated into gradient-based and non-gradient-based techniques, perform well with linear, convex, and straightforward optimization

issues. However, they face limitations when dealing with complex, non-differentiable functions, nonlinear search areas, non-convex issues, and NP-hard problems [2]. To overcome these challenges and inefficiencies, stochastic methods have gained prominence. These approaches take advantage of random operators, random searches, and error-prone procedures to effectively address optimization problems. Optimization algorithms play a crucial role in numerous fields, including medical research, where they have been used to address complex and critical problems, such as disease diagnosis, drug discovery, and treatment optimization [3]. Real-world optimization problems often exhibit complex features that defy the capabilities of deterministic approaches,

* Corresponding author.

E-mail addresses: essam.halim@mu.edu.eg (E.H. Houssein), diego.oliva@cucei.udg.mx (D. Oliva), nmabdelSamee@pnu.edu.sa (N.A. Samee), Nfmahmoud@pnu.edu.sa (N.F. Mahmoud), marwa.khalef@mu.edu.eg (M.M. Emam).

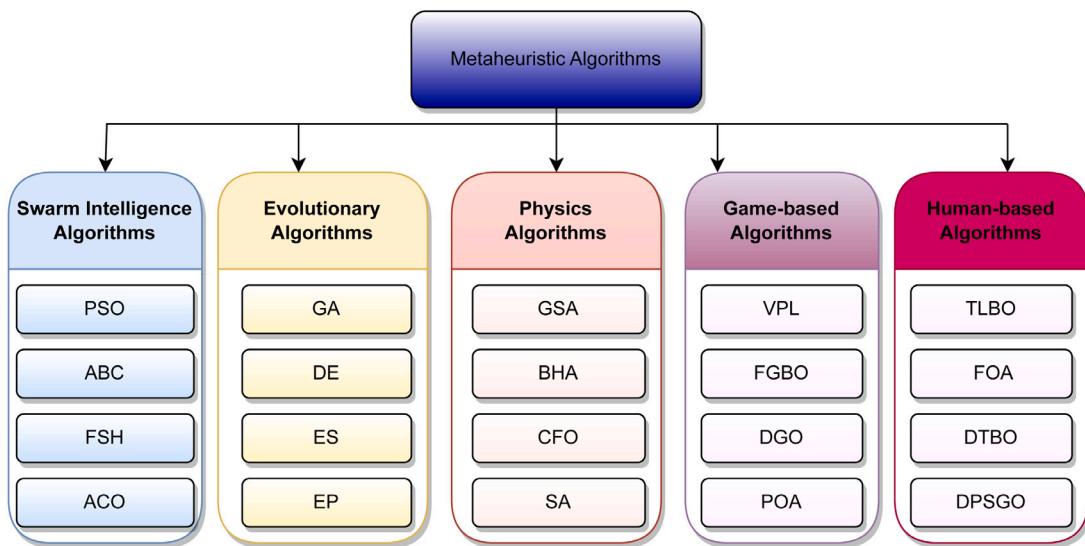


Fig. 1. Optimization Algorithm classification.

making stochastic methods, particularly metaheuristic algorithms, a viable solution to tackle such problems [4].

Metaheuristic algorithms (MAs) have gained popularity due to their ability to deliver effective solutions to optimization problems. Their benefits, such as simple concepts, easy implementation, and efficiency in high-dimensional and nonlinear environments, have contributed to their widespread use. These stochastic approaches initiate the optimization process with randomly feasible solutions in the problem-solving space [5–7]. Candidate solutions are updated and modified according to the algorithm's guidance during repetition-based iterations. Eventually, the best candidate solution among them is the solution of the problem. However, it is essential to note that solutions obtained from MAs might not always guarantee the optimal global solution. The inherent nature of random search in these approaches results in quasi-optimal solutions, which are generally very close to the optimal ones. Despite their simplicity, MAs' performance can vary significantly depending on their search and updating processes for candidate solutions. Researchers strive to achieve better solutions and tailor these algorithms to specific optimization problems. The simplicity of MAs, derived from fundamental theories and mathematical models inspired by nature, enables easy implementation and adaptation to real-world issues. They can be viewed as versatile “black boxes”, capable of producing specific results for different inputs and problems. Researchers can generate suitable solutions for various optimization challenges by modifying their structures and parameters [8,9]. Moreover, the randomness of these algorithms allows them to explore the entire search space, avoiding getting stuck in local regions and facilitating the discovery of desirable solutions. Due to their flexibility and adaptability, MAs excel in addressing a wide range of optimization problems, including complex, nonlinear, and non-differentiable numerical challenges. Their successful application spans various domains, making them valuable tools for researchers and practitioners alike [10,11]. Fig. 1 lists the most popular MAs.

Although there are differences among the various MAs, they have two common phases in the search process: exploration and exploitation [12]. The exploration phase includes a broad and random search of the solution space, whereas the exploitation phase focuses on searching more precisely in the region identified by the exploration phase. The algorithm's precision rises, while its randomness reduces as its exploitation skill improves. If the algorithm has a strong exploration ability, it can quickly converge to diverse solution sets by searching more randomly. Conversely, if the algorithm has a robust exploitative ability, it can boost the quality and precision of the solutions by

searching more locally. However, improving the exploration ability can weaken the exploitative capability and vice versa. Moreover, finding the optimal balance between the two phases can be challenging, as it varies depending on the optimization problem.

1.1. Motivations and contributions

With the growing magnitude of problems and the increasing need for rapid responses, classical methods are no longer sufficient for solving them. MAs have emerged as powerful tools in various fields, including machine learning, image processing, and biomedicine, significantly increasing their utilization. Optimization is a prominent application of MAs in the context of biomedical diseases. The No Free Lunch (NFL) [13] theorem underscores the need for specialized algorithms tailored to specific problem domains since no single algorithm can outperform all others on all optimization issues. To address this challenge, developing new bio-inspired optimization algorithms becomes imperative. These algorithms draw inspiration from natural phenomena, offering robust and efficient optimization solutions. They exhibit vital characteristics such as adaptability, parallelism, and global exploration, making them well-suited for tackling real-world optimization challenges. As complexity increases in various domains, innovative optimization algorithms are required to tackle complex problems, improve decision-making processes, and achieve superior performance. By advancing the optimization field through the implementation of novel bio-inspired algorithms, the limitations of traditional methods can be overcome, leading to improved optimization performance. These advances contribute to progress in diverse domains and have the potential to revolutionize optimization practices. Ultimately, these algorithms enable more effective and efficient solutions to complex problems.

Liver cancer, specifically hepatocellular carcinoma, presents a significant global health challenge due to its high prevalence and aggression. Hepatocellular carcinoma, a prevalent and aggressive form of primary liver cancer, arises from hepatocytes, the liver's main functional cells. The complexities of liver tumor growth, involving interactions with the microenvironment and signaling pathways, inspire algorithm design. Factors such as angiogenesis, immune evasion, and genetic changes contribute to tumor progression. Studying these mechanisms provides insights to develop optimization algorithms that mimic adaptive and robust biological processes. Developing effective treatment strategies for liver cancer is paramount to improving patient outcomes and reduce mortality rates. Optimization algorithms, inspired by

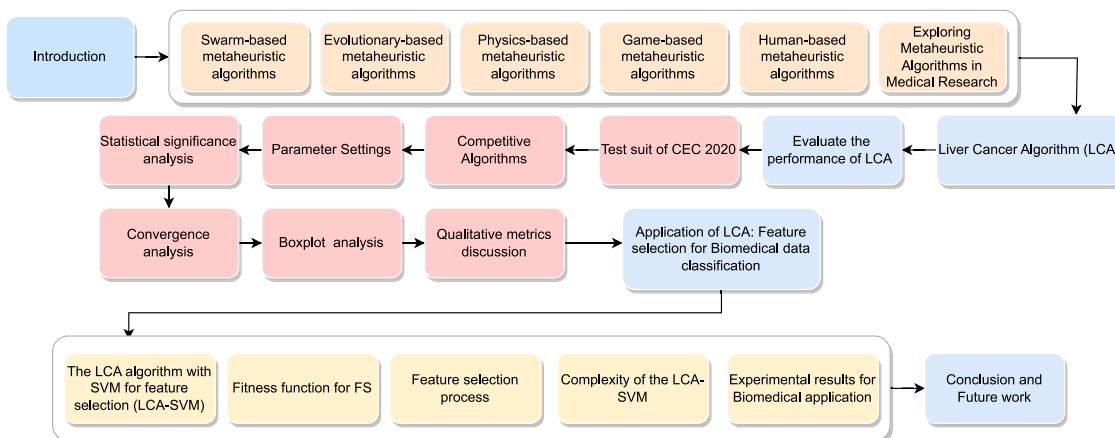


Fig. 2. Outline of the paper.

biological processes and natural phenomena, have shown immense potential in addressing complex medical problems, including liver tumor growth modeling and treatment optimization.

The novelty of this paper is the development of a new bio-inspired optimization algorithm called the Liver Cancer Algorithm (LCA), which is a promising algorithm for solving optimization problems. The inspiration for the LCA algorithm is derived from the behavior of liver tumors in the human body. Liver cancer, specifically hepatocellular carcinoma, is a severe and complex disease that poses significant challenges for diagnosis and treatment. The growth and spread of liver tumors depend on various characteristics, including the location, size, and aggressiveness of the tumor. The behavior of liver tumors can be modeled in terms of optimization concepts. In the context of optimization, the LCA algorithm mimics the process of spreading and growth of liver tumors in the liver. The algorithm's design is inspired by the tumor's adaptive nature, where it tries to find the most favorable environment for growth within the liver. It uses an evolutionary search approach that simulates the behavior of liver tumors when they take over the liver organ. The design of the LCA algorithm incorporates genetic operators and a Random Opposition-Based Learning (ROBL) strategy. Genetic operators in the LCA enable it to maintain diversity among candidate solutions and prevent premature convergence, increasing the likelihood of finding high-quality solutions. Moreover, the integration of a random opposition-based learning strategy in the LCA provides an innovative approach to initialize the population. The algorithm achieves a more balanced exploration and exploitation trade-off by generating opposite solutions to existing ones and incorporating them into the population. This strategic initialization helps the LCA avoid being trapped in a local optimum and increases the probability of discovering optimal or near-optimal solutions across the search space. This algorithm offers a good method for solving a range of real-world problems, further contributing to the optimization field. The efficacy of the LCA algorithm has been thoroughly evaluated on multiple fronts. Firstly, it has been tested on the CEC'2020 functions [14]. Furthermore, we leveraged the LCA's capabilities to address the challenging feature selection (FS) problem. We developed new wrapper FS algorithms based on the LCA's structure, enabling selection of relevant features for various biomedical datasets. Additionally, we integrated the LCA algorithm with the SVM classifier (LCA-SVM), facilitating accurate and rapid classification rates. To establish a comprehensive comparison, this paper has compared the LCA algorithm with eight other metaheuristic algorithms: GA, PSO, DE, AGDE, IMODE, HHO, RUN, INFO, and CHIO. The results of extensive experimentation have consistently shown that the LCA-SVM method outperforms many algorithms. The LCA algorithm can remarkably strike a balance between exploration and exploitation. This characteristic improves its convergence rate and enables it to perform more effective global searches.

These advantages contribute to its superior performance in various optimization scenarios. The following is a summary of this paper's main contributions:

- A new algorithm called LCA based on liver cancer is proposed.
- The proposed LCA is built based on genetic operators and ROBL.
- LCA has the main advantage of appropriately balancing exploration and exploitation. ROBL can provide opposite solutions that improve convergence and avoid getting stuck on a local optimum.
- LCA employs a mutation rate adjusted based on the number of iterations. This higher mutation rate facilitates the exploration of promising new locations and protects the algorithm from being a trap at a local minimum.
- Ten global optimization problems from the CEC'2020 test suite have been employed to evaluate the proposed LCA against several optimization algorithms.
- The LCA was applied to solve the FS problems. Several biomedical datasets were also used to evaluate LCA.

The proposed LCA algorithm offers several advantages when applied to global optimization problems. First, the LCA algorithm demonstrates high efficiency and effectiveness in addressing optimization problems in various scientific disciplines. It excels at handling complex high-dimensional problems, where its capabilities are well-suited to provide optimal solutions. A noteworthy advantage of the LCA algorithm is its remarkable ability to balance exploration and exploitation during the search process. This ability allows for rapid convergence, enabling the algorithm to converge efficiently toward suitable values for decision variables in optimization problems. This feature is precious when dealing with complex problems that require careful navigation through complex solution spaces. Another key advantage of the LCA algorithm is its exceptional performance in real-world optimization applications. The LCA algorithm leverages its strengths to enhance its problem solving capabilities by employing genetic algorithm operators such as crossover and mutation. The crossover operator facilitates the exchange of genetic information, allowing the algorithm to explore diverse solution spaces. On the other hand, the mutation operator introduces randomness, promoting exploration, and preventing premature convergence.

The remainder of this paper is structured as follows. Section Literature review presents the Literature review. The proposed LCA algorithm is presented in Section 3. The evaluation results and the analysis of the efficiency of the LCA algorithm are discussed in Section 4. Section 5 discusses the application of the LCA algorithm to feature selection problems, including SVM and hybrid LCA with SVM. Furthermore, Section 5.2 presents the simulation analysis of the LCA algorithm to solve FS problems, together with series of experimental data for biomedical applications. Finally, Section 6 concludes the article and

provides suggestions for future directions. The outline of the paper is given in Fig. 2

2. Literature review

Metaheuristic algorithms derive their inspiration from various natural phenomena, including the actions of animals, insects, birds, and other living organisms, physical laws, human activities, game rules and various evolution-based processes. These algorithms can be broadly classified into five groups based on the primary sources of inspiration that inform their design: swarm-based algorithms, evolutionary-based algorithms, physics-based algorithms, game-based algorithms, and human-based algorithms. Each group captures the unique aspects of natural or human phenomena and uses them to devise innovative optimization strategies. Using the principles of nature and human behavior, metaheuristic algorithms offer powerful and versatile optimization techniques that can effectively address complex real-world problems.

2.1. Swarm-based metaheuristic algorithms

Swarm-based MAs draw motivation from the collective behaviors of animals, insects, and birds in nature, where individuals coordinate and interact to achieve common objectives. The design of swarm-based algorithms often revolves around animals' natural behaviors, such as searching for food, hunting, and navigating their environment. Some of the well-established algorithms in this class include PSO [15], Artificial Bee Colony (ABC) [16], and Ant Colony Optimization (ACO) [17]. For instance, PSO efficiently explores the search space for ideal solutions by modeling the flocking behavior of birds. Similarly, ABC is motivated by the foraging behavior of bee colonies, while ACO mimics the ants' ability to discover the shortest path between their nest and food sources. Furthermore, the Gray Wolf Optimization (GWO) [18] algorithm takes cues from the social life and hunting strategies of gray wolves, the Whale Optimization Algorithm (WOA) [19] derives inspiration from humpback whales' bubble-net hunting strategy, and the Marine Predators Algorithm (MPA) [20] emulate the search behaviors of marine predators, including Brownian and Lévy mechanisms during prey hunting. Also, there exist other highly used algorithms such as HHO [21], Slime Mould Algorithm (SMA) [22], and Tunicate Search Algorithm (TSA) [23]. Notable examples include the White Shark Optimizer (WSO) [24], Horse herd Optimization Algorithm (HOA) [25], the Reptile Search Algorithm (RSA) [26], the Hunger Games Search (HGS) [27], and Colony Predation Algorithm (CPA) [28]. Moreover, the Komodo Mlipir Algorithm (KMA) [29], which emulates the behavior of Komodo dragons, the RUN algorithm [30], the Golden Jackal Optimization (GJO) algorithm [31], the Giant trevally optimizer (GTO) [32], and the Artificial Hummingbird Algorithm (AHA) [33], which emulates the flight abilities and foraging techniques of hummingbirds. These algorithms offer versatile optimization strategies, harnessing the power of collective intelligence and cooperation observed in nature's swarming phenomena.

2.2. Evolutionary-based metaheuristic algorithms

Evolutionary-based MAs draw stimulation from biological sciences, genetics, natural selection, and random operators. These algorithms model concepts from biology, such as the reproductive process and natural selection, and utilize random operators such as selection, crossover, and mutation. Among the well-known evolutionary algorithms are the Genetic Algorithm (GA) [34] and Differential Evolution (DE) [35], which are based on mathematical modeling of biological processes. Evolutionary-based metaheuristics have gained widespread popularity for their effectiveness in solving complex optimization issues.

2.3. Physics-based metaheuristic algorithms

Physics-based MAs are optimization techniques simulated by different physical phenomena, concepts, and laws. Algorithms simulate real-world processes to guide the search for optimal solutions. Examples include Simulated Annealing (SA) [36], which mimics metal annealing, and Gravitational Search Algorithm (GSA) [37], inspired by Newtonian gravity. These algorithms use physics-based principles to update search agents and find optimal points. Other physics-based approaches, such as the Spring Search Algorithm (SSA) [38], and the Multi-Verso Optimizer (MVO) [39], draw inspiration from spring forces, momentum, and cosmological concepts, respectively. Additionally, the Equilibrium Optimizer (EO) [40], Archimedes Optimization Algorithm (AOA) [41], Henry Gas Solubility Optimization (HGSO) [42], and INFO [43] utilize various physical laws to govern search agent interactions. These algorithms offer innovative solutions to optimization problems and have found applications in different engineering disciplines. These algorithms combine insights from physics with optimization, providing practical and versatile tools to solve complex optimization challenges.

2.4. Game-based metaheuristic algorithms

Game-based MAs draw inspiration from various games, players' behaviors, and coaches' strategies. For example, the volleyball Premier League (VPL) method models interactions and decision-making in volleyball matches [44]. The Football Game-Based Optimizer (FGBO) [45] is influenced by players' behavior and club managers' decisions in football leagues. These algorithms are further enriched by incorporating players' efforts to improve their performance, leading to the development of innovative approaches such as the Darts Game Based Optimizer (DGO) [46]. Furthermore, the puzzle optimization Algorithm (POA) [47] mirrors the cognitive processes used in puzzle-solving. These game-based metaheuristics show how games and sports activities provide valuable insights for designing efficient optimization techniques.

2.5. Human-based metaheuristic algorithms

Human-based MAs are optimization techniques that draw inspiration from various aspects of human activities and dealings in both particular and social life. These algorithms simulate human processes through mathematical models to solve optimization problems effectively. One prominent example of a human-based algorithm is Teaching-Learning-Based Optimization (TLBO) [48]. TLBO is designed based on the dynamics of a classroom educational setting, where a teacher interacts with students to enhance their knowledge and understanding. Another noteworthy approach is the Following Optimization Algorithm (FOA) [49], inspired by a community leader's impact on individuals' progress. By simulating this leadership influence, FOA guides the optimization process toward better solutions. The realm of human-based metaheuristics expands further with algorithms like Driving Training-Based Optimization (DTBO) [50], Poor and Rich Optimization (PRO) [51], Dual-Population Social Group Optimization (DPSGO) [52], and Human Mental Search (HMS) [53]. To devise effective optimization strategies, these algorithms model various human activities, such as driving training, the economic activities of different social groups, brainstorming processes, and the way human eye vision works.

2.6. Exploring metaheuristic algorithms in medical research

MAs have gained significant attention in medical applications due to their familiar use in disease detection and classification effectiveness. For example, Thawkar et al. [54] suggested a hybrid approach called BOAALO for selecting features and predicting breast cancer. Similarly,

Sayed et al. [55] developed a hybrid convolutional neural network combined with Bald Eagle Search (BES) optimization to detect skin cancer. Xing et al. [56] introduced the enhanced Whale Optimizer with Quasi-Oppositional Learning and Gaussian Barebone for FS and COVID-19 Image Segmentation. Furthermore, Piri et al. [57] utilized the HHO algorithm for FS in medical data analysis. These studies highlight metaheuristic algorithms' diverse applications and effectiveness in medical research. Moreover, the authors of [58] develop an efficient image segmentation method for COVID-19 CT images using a cooperative swarm intelligence-based approach. The study aims to harness the advantages of parallel meta-heuristics and machine learning techniques (MLT) to enhance the accuracy and effectiveness of the segmentation process. The proposed method, CPGH (Cooperative PSO, GWO, and HHO), combines three practical algorithms in a cooperative model. Algorithms are executed concurrently and potential solutions are migrated across their populations after a set number of generations. The CPGH model is evaluated using three objective functions (cross-entropy, Otsu's, and Tsallis) on COVID-19 CT images obtained from open-sourced datasets. Researchers have recently developed several biological algorithms to tackle FS challenges, as evidenced by Khalid et al. [59]. One such method is the Coronavirus Disease Optimization Algorithm (COVIDOA), which simulates the reproduction conduct of the coronavirus during human cell infection. The study's authors have introduced a new binary optimization algorithm called the Binary Coronavirus Disease Optimization Algorithm (BCVIDOA) for FS. Moreover, many researchers have concentrated on this domain. For example, Houssein et al. [60] introduced a hybrid approach that combines the Search and Rescue optimization algorithm with the k-nearest neighbors (k-NN) and a wrapper FS technique. This approach aims to improve classification accuracy, identify the best set of features, and reduce search space size. Similarly, Abd Elaziz et al. [61] combined a boosted HHO algorithm based on heavy-tailed distributions and Henry gas solubility optimization (HGSO) with a growing search space to address FS issues for various chemical datasets to avoid the drawbacks of traditional FS methods. In the drug development, evaluating the potential risks of unknown biotransformed medications is crucial, as noted in [62]. Another approach to predict the drug product is a rough set of FS approaches [63,64].

Based on an extensive literature review, no existing metaheuristic algorithm has been specifically developed to model the behavior of liver tumors in the human body. The intricate growth and spread of liver tumors and their adaptive nature represent intelligent actions that can be the basis for an innovative optimizer design. Therefore, to fill this research gap and take advantage of insights from liver tumor behavior, we have developed a novel optimization algorithm, the Liver Cancer Algorithm (LCA), which will be presented in the subsequent section.

3. Liver Cancer Algorithm (LCA)

This section describes the LCA algorithm and presents a detailed mathematical formulation of its various steps. The behavior of liver tumors inspires LCA and integrates biological principles into its optimization process, making it a novel and practical approach for feature selection.

3.1. Mathematical model of liver cancer algorithm (LCA)

The LCA algorithm is designed to mimic the growth and spreading behavior of liver tumors, malignant growths in the liver that can severely impact the body's functionality. To replicate the tumor's growth and behavior, the LCA algorithm consists of several stages, each involving distinct mathematical formulations.

3.1.1. Tumor size calculation

At the LCA algorithm's core is the tumor's size calculation, which is essential for subsequent stages. To determine the size of the tumor, we employ a mathematical model based on the assumption that tumors have a hemi-ellipsoid shape, as observed in various types of cancers, including liver cancer [65]. The tumor is represented by three diameters: width, length, and height. However, the height dimension is challenging to measure directly, so we approximate it using a mathematical model.

The initial tumor size (position) is calculated using Eq. (1), which incorporates random opposition-based learning (ROBL) to establish the starting population with diverse search exploration capabilities.

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

where $i = 1, 2, \dots, N$, $j = 1, 2, \dots, D$. lb and ub indicate the upper and lower boundaries of the decision variables, D is the dimension of the search area, r_d is a random number between $[0, 1]$ based on the best objective of the current Position_i , and the opposite Position_i^* , so the start population is established. The size of the tumor position is calculated based on the length of tumor diameters (width, height, and length) length; width and height are random numbers between $[0, 1]$. The increase in the size of the liver tumor position is mathematically described in this study to create the LCA algorithm and perform optimization. The increase in tumor position size (position) is calculated by Eq. (2).

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

where $f = 1$ is a constant that describes a particular type of tumor [66].

3.1.2. Tumor replication

This stage of the tumor indicates the dangerous stage of disease infection after the size of the tumor has developed. Tumors replicate themselves in another place in the same liver organ. According to [67] exponential growth equation for breast cancer and liver tumor (hepatocellular carcinoma) is good for calculating the growth of the tumor position as shown in Eq. (3). The size of the tumor position V is satisfied because the cells divide constantly regardless of the size of the tumor.

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

where PG is the growth of the tumor position, r indicates the radius as the ellipse shape of the tumor defined in $[0, 1]$, the size and location of the tumor are calculated from the previous equation, T is the maximum iteration, and N is the number of the search agent. The spread of the tumor increases to obtain the optimal location to control the disease in the liver organ. The L'evy flight function represents this mechanism [68,69] as presented in Eqs. (4) and (5).

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

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

In this stage of the LCA algorithm, which was motivated by actual tumor-spreading processes, we assumed that they could gradually choose the best dive into the liver portions to capture them in competitive situations. Consequently, we considered that the tumor could assess (decide) its next action depending on the following rule to complete this phase. This procedure is described in Eqs. (6), (7), and (8).

$$y = \text{Position} + PG \quad (6)$$

$$Z = Y + S \times LF(D) \quad (7)$$

$$\text{Position}_{t+1}^i = \begin{cases} y & \text{if } \text{fit}(y) < \text{fit}(\text{Position}_t^i) \\ z & \text{if } \text{fit}(z) < \text{fit}(\text{Position}_t^i) \end{cases} \quad (8)$$

while D refers to the dimension space, S contains D components developed randomly in that space [0,1]. Finally, $\text{fit}(\cdot)$ is the fitness function that would be optimized.

3.1.3. Tumor spreading using crossover and mutation

The final stage of the LCA algorithm represents the progression of a tumor to metastatic liver cancer, an advanced form of the disease that originates in the liver but spreads to other parts of the body [70]. LCA employs genetic operators, including mutation and crossover, to introduce additional variation to achieve this.

Mutation: The mutation operation is implemented in LCA structural tasks as a solution objective. Each component is assigned a random number in [0,1]. When the value approaches the mutation rate (ζ), the tumor position of the target agent element is considered. A component from the y or z vectors is utilized to update the old vector if the value is less than the mutation rate (ζ). The mutation operator is built using the equations from Eq. (9) to Eq. (12) to indicate tumor development using the mutation operator: where j th dimension is defined by lb^j and ub^j , S have D elements derived from random numbers in [0, 1].

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

$$z_{\text{Mut}} = \begin{cases} \text{Position} & \text{if } \text{rand}_2 \geq \zeta \\ z & \text{else} \end{cases}$$

$$\text{Where: } \begin{cases} \zeta = \frac{1}{T}; \\ y = |\text{Position} - \text{Position}_i^j|; \\ z = y - S \end{cases} \quad (10)$$

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

$$z_{\text{Mut}} = \begin{cases} \text{Position} & \text{if } \rho_2 \geq \zeta \\ z & \text{else} \end{cases}$$

$$\text{Where: } \begin{cases} \zeta = \frac{t}{T}; \\ y = |\text{Position} - \text{Position}_i^j|; \\ z = y - S \end{cases} \quad (12)$$

Crossover: Crossover is used to combine two individuals to produce new offspring. In LCA, we use a linear combination of random numbers τ and τ' to create the new offspring w_{Cross} , as defined in Eq. (13).

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

Selection: The application of the greedy selection strategy in LCA is based on differential evolution. Children are generated when the evaluation functions (mutation and crossover) are acquired. The performance of parents and children is then evaluated in parallel and parents can remain in the population if they perform well. Using Eq. (14) establishes the criteria for greedy selection.

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

The pseudocode of the LCA algorithm is discussed in Algorithm 1, while Fig. 3 illustrates the corresponding flowchart.

Algorithm 1 Pseudocode of LCA algorithm.

```

1: Initialization: Initialize liver tumor position size population by Eq. (1).
2: while (The termination requirement has not been satisfied) do
3:   Estimate the tumor position fitness value. ▷ Exploitation phase
4:   if ( Position ==< 0.8) or( position > 0.8) then
5:     Update the tumor position with Eq. (2)
6:   end if
7:   else if ( Position > 0.8) and ( Position < 1.96)
8:     Update the tumor position by Eq. (3)
9:   else if (Position > 1.96) ▷ Exploration phase
10:    Using Eq.(8), modify the position replication of the current search agent.
11:    else Apply genetic operator Eqs. (9) – (13) to calculate Position tumor spreading .
12:    Using Eq.(14), update the current search agent's tumor position replication.
13:    Compute each tumor position fitness.
14:     $t = t + 1$ 
15: end while
16: Return bestFitness, Position

```

3.2. Time complexity of the LCA algorithm

The time complexity of the LCA algorithm mainly arises from three main procedures: initialization, fitness assessment, and update of tumor position. The initialization step has a complexity of $O(N)$, where N represents the number of tumors in the algorithm population. Subsequently, during each iteration, the algorithm assesses the fitness of each tumor, resulting in a complexity of $O(T \times N)$, where T denotes the number of iterations. Furthermore, tumor position update involves finding the best position and updating the position vector for every tumor, resulting in a complexity of $O(T \times N \times D)$, where D represents the dimension of the optimization problem. Combining the complexities of the three procedures, the overall computational complexity of the LCA algorithm can be defined as $O(N \times (T + T \times N + D + 1))$.

4. Experiments to evaluate the performance of LCA

The proposed LCA algorithm is tested using the CEC'2020 test suite, which consists of ten benchmark functions. Hybrid, multimodal, unimodal, and composite optimization problems are included in the CEC'2020 test set [14]. Fig. 4 illustrates a 2D representation of the CEC'2020 functions to aid comprehend each problem's core. Note that for experimental purposes, the dimensionality of the problems is set to 10 and 20. Several metrics are used, including qualitative metrics such as historical search values, average fitness history, and optimization history. In addition, they used quantitative metrics such as the mean and standard deviation (STD) to find the best solutions. A non-parametric Friedman test was involved in verifying the efficiency of the compared algorithms.

4.1. The competitive algorithms

The LCA algorithm is evaluated and compared with several widely recognized metaheuristic algorithms, which are listed as follows:

1. The Genetic Algorithm (GA) [34]: GA is an evolutionary optimization algorithm motivated by natural selection and genetics. It begins with an initial individual of possible solutions represented as chromosomes. New offspring are created and evaluated based on a fitness function through selection, crossover, and mutation. The adaptability and ability of GA to explore various solutions make it effective for optimization in various domains.

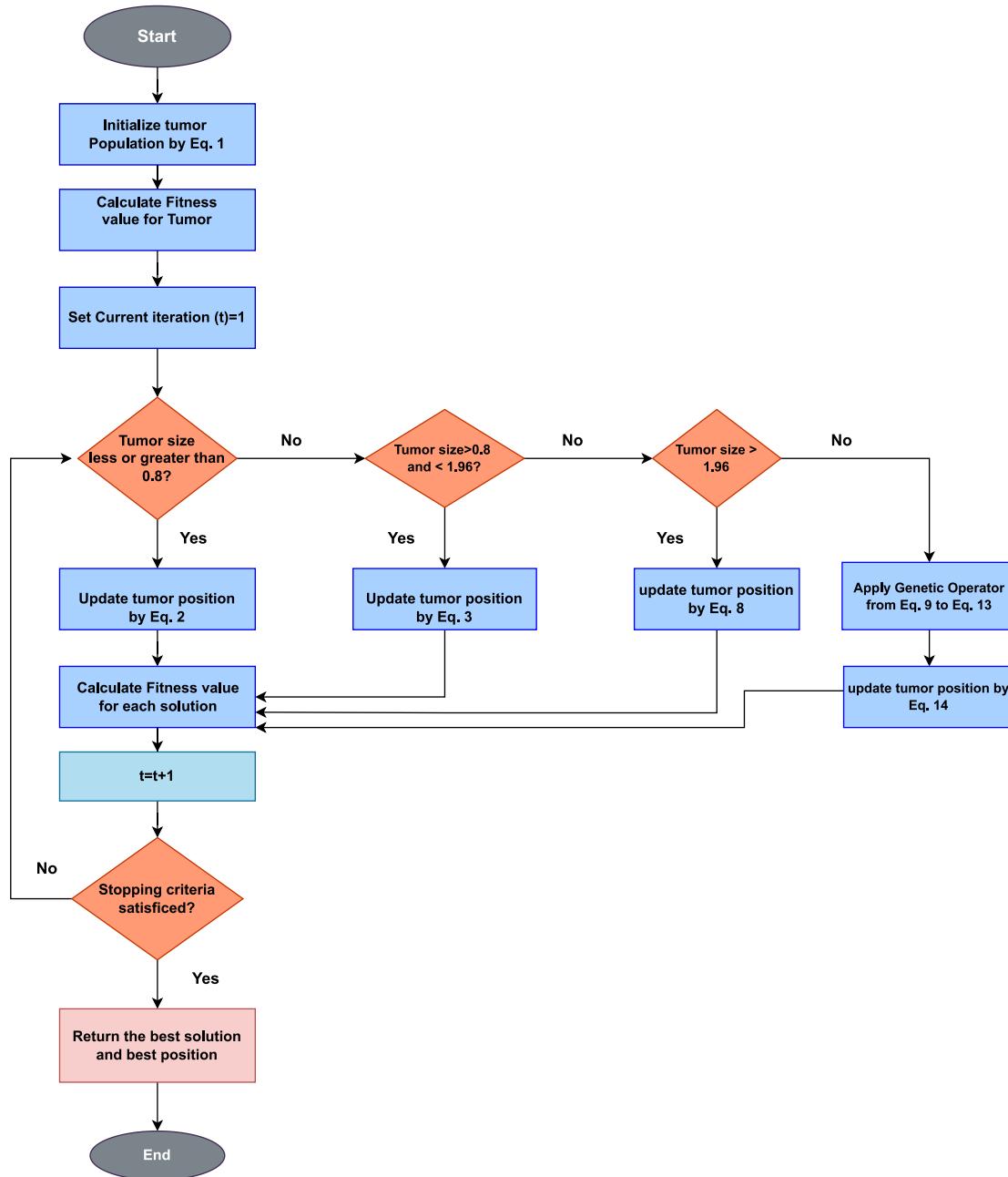


Fig. 3. LCA Flowchart.

2. Particle Swarm Optimization (PSO) [15]: PSO algorithm is a population-based optimization technique motivated by the collective behavior of bird flocking. It utilizes a particle swarm that moves through the search space to discover the optimal solution. Each particle maintains a position and velocity, updated iteratively based on its own experience and the experience of the best-performing particle in the swarm. This algorithm has been used successfully to solve several optimization issues and has proven to be effective in doing so.
3. Differential Evolution (DE) [35]: DE is a population-based optimization algorithm that utilizes mutation, crossover, and selection operations. It starts with an initial population and generates trial solutions by perturbing individuals. The algorithm selects the best solution between the target and trial vectors based on their fitness values.
4. Adaptive Guided Differential Evolution (AGDE) Algorithm [71]: AGDE algorithm is a metaheuristic optimization algorithm that

combines differential evolution with guided search. It adapts its control parameters depending on the fitness function of the current population and operates a guided search mechanism to improve exploration and exploitation.

5. Improved Multi-Operator Differential Evolution (IMODE) algorithm [72]: IMODE is a metaheuristic optimization algorithm that extends the traditional differential evolution approach by introducing multiple mutation and crossover operators. Integrates a diverse set of operators to enhance the exploration and exploitation capabilities of the algorithm. This algorithm effectively solves various optimization problems and exhibits a good balance between exploration and exploitation.
6. Harris Hawks Optimization (HHO) [21]: HHO algorithm is a bio-inspired optimization algorithm that mimics the cooperative hunting behavior of Harris's hawks. It uses a social hierarchy and individual collaboration to find optimal solutions. The HHO

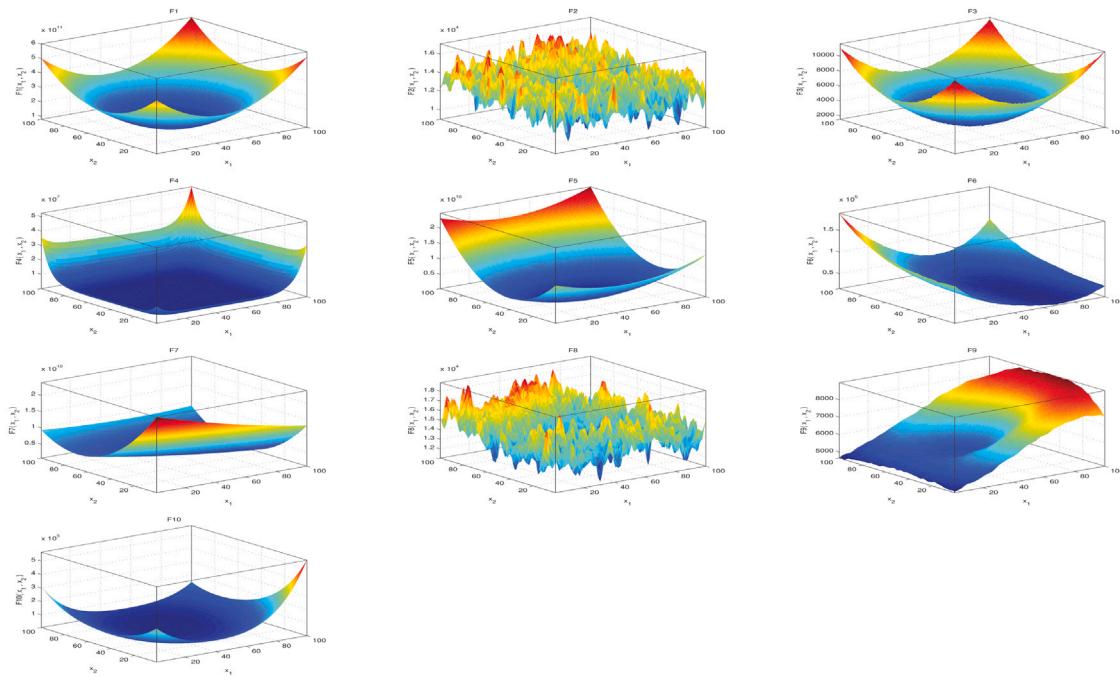


Fig. 4. The 2D visualization of the CEC'2020 benchmarks.

algorithm efficiently explores the search space and converges toward ideal solutions by striking a balance between exploration and exploitation and using a variety of search operators.

7. Runge–Kutta Optimization Algorithm (RUN) [30]: RUN algorithm is an optimization algorithm inspired by the Runge–Kutta numerical method. It employs population dynamics and differential equations to perform optimization.
8. weighted meaN OF vectOrs (INFO) [43]: INFO algorithm computes the weighted mean of a group of vectors, where the fitness values of the vectors determine the weights. With this strategy, the algorithm can navigate the search space efficiently while balancing exploration and exploitation. The INFO algorithm aims to converge toward optimal solutions by leveraging the weighted mean computation.
9. Coronavirus Herd Immunity Optimizer (CHIO) [73]: CHIO algorithm is a bio-inspired optimization algorithm that mimics achieving herd immunity during the COVID-19 pandemic. Utilizes vaccination and infection operators to explore and optimize solutions in the problem space. The CHIO algorithm searches for optimal or near-optimal solutions by balancing exploration and exploitation.

4.2. Parameter settings

The parameter configurations for the compared algorithms are presented in [Table 1](#). To ensure a fair and impartial comparison, default values are adopted based on the findings of a study by Arcuri and Briand [74]. This study demonstrated that default parameterizations offer a suitable basis for algorithmic comparisons, minimizing any potential bias from algorithm-specific tuning. Using default values, we aim to provide an objective evaluation of the performance of the algorithm. Furthermore, to ensure the reliability and robustness of the benchmarking comparison, we independently executed the simulations 30 times. This approach accounts for potential variability from the algorithms' random initialization or stochastic elements. Conducting multiple runs enables us to assess their efficiency and effectiveness with more accuracy.

Table 1
Control parameter values for compared algorithms.

Algorithm	Parameter	Value
GA	Selection rate	0.8
PSO	Population size and Velocity	50 , and 65
DE	Num Gen	500
AGDE	p	0.1
IMODE	D	2
HHO	Beta	1.5
RUN	a,b	20,12
INFO	C,D	2,4
CHIO	A,S	0,1,0,1,2

4.3. Statistical significance analysis on CEC'2020 benchmark functions

The exploratory and exploitative capabilities of the LCA algorithm compared to other well-known algorithms are shown in in-depth results and comparisons in this section. [Table 2](#) provides the mean fitness and STD for LCA, along with other algorithms, over 10 CEC'2020 benchmark functions of Dim = 10. Similarly, [Table 3](#) presents the mean fitness and STD results on 10 CEC'2020 benchmarks of Dim = 20. The best values are highlighted in bold to emphasize how well the proposed LCA algorithm performs compared to existing optimization techniques across various solution dimensions.

As demonstrated in [Table 2](#), LCA achieved the best overall performance among all algorithms, as confirmed by the Friedman mean rank sum test. It outperformed other algorithms, including IMODE, CHIO, AGDE, INFO, PSO, HHO, GA, DE, and RUN, making it the most promising algorithm for solving the CEC'2020 functions with a solution dimension of Dim = 10. Analyzing the individual benchmark functions, we observe that for function F1, the LCA algorithm obtained a mean value of $1.00E + 02$ with a remarkably low standard deviation. These results outperformed all the compared algorithms. For function F2, the GA algorithm performs quite well, having the lowest mean fitness value ($1.20E+03$) among all algorithms, indicating its effectiveness in this function. On the contrary, the LCA outperforms the INFO, CHIO, HHO, IMODE, and PSO algorithms. In the case of function F3, LCA outperformed GA, DE, PSO, AGDE, HHO, and IMODE in both mean and

Table 2

Mean and STD of fitness values for 30 runs of competitor algorithms on the CEC'20 test suite with dimension 10.

Algorithms	Criteria	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	Friedman	Rank
GA	Mean	1.66E+08	1.20E+03	7.34E+02	1.90E+03	5.89E+03	1.61E+03	4.93E+03	2.31E+03	2.74E+03	2.94E+03	5.3	2
	STD	2.87E+08	8.34E+01	4.20E+00	4.48E-01	3.53E+03	1.36E+01	2.00E+03	9.86E+00	1.63E+01	1.04E-01		
PSO	Mean	4.04E+05	2.03E+03	7.60E+02	1.90E+03	5.71E+03	1.60E+03	1.53E+04	2.71E+03	2.77E+03	2.94E+03	3.7	2
	STD	5.97E+05	3.44E+02	3.65E+01	8.98E-01	2.39E+03	2.75E-01	6.23E+03	6.86E+02	1.40E+01	1.83E+01		
DE	Mean	3.42E+04	1.41E+03	7.34E+02	1.90E+03	9.48E+03	1.60E+03	9.38E+03	2.31E+03	2.75E+03	2.93E+03	7.1	7
	STD	2.84E+04	1.15E+02	2.26E+00	9.31E-01	1.28E+03	2.52E-01	4.19E+03	3.97E+00	2.03E+01	2.75E+01		
AGDE	Mean	4.43E+05	1.74E+03	8.02E+02	1.91E+03	4.48E+04	1.61E+03	9.48E+03	2.31E+03	2.86E+03	2.96E+03	6.1	5
	STD	7.17E+04	4.12E+02	1.24E+01	1.33E+00	3.77E+04	9.91E+00	9.72E+03	1.00E+01	6.61E+01	6.37E+01		
IMODE	Mean	1.00E+02	1.66E+03	7.30E+02	1.90E+03	2.28E+03	1.60E+03	2.29E+03	2.30E+03	2.76E+03	2.95E+03	7.1	8
	STD	1.73E-13	6.83E+01	7.51E+00	1.67E-01	1.50E+02	2.53E-01	1.79E+02	7.34E-01	1.12E+01	3.37E-01		
HHO	Mean	3.35E+02	2.49E+03	7.16E+02	1.90E+03	2.65E+05	1.63E+03	9.18E+03	2.30E+03	2.59E+03	2.94E+03	2.5	2
	STD	2.27E+02	1.89E+02	1.16E+00	2.53E-01	1.39E+05	2.15E+01	1.95E+03	1.68E-11	1.51E+02	2.08E-02		
RUN	Mean	1.03E+04	1.57E+03	7.20E+02	1.90E+03	1.40E+04	1.60E+03	2.20E+03	2.30E+03	2.76E+03	2.92E+03	4.3	3
	STD	4.03E+03	6.16E+01	3.30E+00	3.75E-01	4.18E+03	2.41E-04	1.36E+02	6.79E-01	3.92E+00	4.13E+01		
INFO	Mean	9.42E+02	2.30E+03	7.14E+02	1.90E+03	3.24E+05	1.62E+03	1.09E+04	2.30E+03	2.61E+03	2.94E+03	6.2	6
	STD	9.87E+02	3.59E+02	4.83E-01	2.00E-01	1.48E+05	2.39E-01	3.55E+03	3.44E-11	1.18E+02	3.73E-02		
CHIO	Mean	1.37E+03	2.57E+03	7.14E+02	1.90E+03	2.43E+05	1.64E+03	1.03E+04	2.30E+03	2.67E+03	2.94E+03	8.1	7
	STD	1.18E+03	2.26E+02	3.55E+00	1.29E-01	7.14E+04	1.29E+01	4.70E+03	3.40E-11	1.48E+02	2.03E-02		
LCA	Mean	1.00E+02	1.77E+03	7.26E+02	1.90E+03	2.17E+03	1.60E+03	2.64E+03	2.30E+03	2.67E+03	2.93E+03	1.6	1
	STD	3.24E-11	3.20E+02	5.38E+00	3.20E-01	1.15E+02	2.53E-01	2.57E+02	4.78E-01	1.51E+02	2.91E+01		

Table 3

Mean and STD of fitness values for 30 runs of competitor algorithms on the CEC'20 test suite with dimension 20.

Algorithms	Criteria	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	Friedman	Rank
GA	Mean	6.64E+08	2.36E+03	7.84E+02	2.44E+03	1.79E+06	1.60E+03	1.16E+05	2.82E+03	2.84E+03	2.98E+03	7.1	5
	STD	6.04E+08	3.85E+02	2.35E+01	9.32E+02	1.14E+06	2.78E-13	1.03E+05	7.90E+02	2.50E+01	3.62E+01		
PSO	Mean	4.10E+06	4.06E+03	9.26E+02	1.93E+03	1.15E+06	1.60E+03	1.25E+06	3.42E+03	3.03E+03	3.00E+03	5.2	2
	STD	1.24E+06	7.53E+02	1.90E+01	5.52E+00	7.05E+05	2.78E-13	8.35E+05	1.91E+03	1.42E+01	3.26E+01		
DE	Mean	3.84E+08	2.24E+03	7.66E+02	1.90E+03	4.77E+05	1.60E+03	1.57E+05	2.39E+03	2.87E+03	2.97E+03	8.1	7
	STD	6.66E+08	2.86E+02	1.27E+01	5.00E-01	3.81E+05	2.78E-13	1.22E+05	4.07E+01	5.17E+01	4.41E+01		
AGDE	Mean	3.77E+06	3.62E+03	8.91E+02	1.93E+03	2.92E+05	1.60E+03	1.83E+05	4.84E+03	3.17E+03	2.97E+03	7.2	2
	STD	9.14E+05	7.93E+02	1.02E+01	2.23E+00	1.65E+05	2.78E-13	7.66E+04	2.23E+03	1.97E+01	2.84E+01		
IMODE	Mean	1.00E+02	3.06E+03	8.23E+02	1.91E+03	2.22E+04	1.60E+03	3.39E+03	4.23E+03	2.88E+03	2.95E+03	9.1	8
	STD	1.97E-09	6.20E+02	6.03E+01	1.98E+00	3.22E+04	2.78E-13	1.67E+02	1.87E+03	1.64E+01	3.85E+01		
HHO	Mean	5.76E+02	3.48E+03	7.40E+02	1.90E+03	2.91E+05	1.60E+03	1.40E+05	2.30E+03	3.02E+03	2.99E+03	9.3	9
	STD	2.99E+02	5.01E+02	5.14E+00	4.83E-01	1.37E+05	2.78E-13	1.51E+04	8.61E-10	5.15E+01	8.66E+00		
RUN	Mean	8.59E+03	1.91E+03	7.46E+02	1.90E+03	1.87E+05	1.60E+03	1.59E+05	3.34E+03	2.86E+03	2.91E+03	5.9	3
	STD	1.80E+03	3.00E+02	1.78E+01	9.11E-01	1.86E+05	2.78E-13	1.52E+05	9.23E+02	1.28E+01	1.27E-01		
INFO	Mean	1.91E+02	3.70E+03	7.36E+02	1.90E+03	2.61E+05	1.60E+03	1.09E+05	2.30E+03	2.89E+03	2.98E+03	8.2	6
	STD	4.79E+01	7.11E+02	4.64E+00	1.14E-01	1.16E+05	2.78E-13	6.24E+04	1.03E-09	2.61E+02	1.45E+01		
CHIO	Mean	4.01E+02	3.58E+03	7.31E+02	1.90E+03	4.01E+05	1.60E+03	1.31E+05	2.30E+03	3.00E+03	3.00E+03	10.1	10
	STD	3.12E+02	5.42E+02	3.80E+00	4.07E-01	8.73E+04	2.78E-13	5.61E+04	8.82E-10	3.19E+01	7.83E+00		
LCA	Mean	1.00E+02	2.39E+03	8.16E+02	1.90E+03	3.95E+03	1.60E+03	3.04E+03	2.29E+03	2.92E+03	2.91E+03	5.3	1
	STD	7.95E-08	3.36E+02	5.17E+00	3.22E+00	1.43E+03	2.78E-13	5.33E+02	1.14E+03	2.11E+01	3.59E+01		

standard deviation, while the CHIO and INFO algorithms obtained the best performance for this function. In the case of F4, LCA's performance was approximately similar to other algorithms. Moving on to functions F5, F6, F7, LCA, superior to all other algorithms. In particular, for the function F8, the LCA obtains the best performance. Additionally, for F9, INFO outperformed all other algorithms and LCA outperformed some other algorithms, including RUN, IMODE, AGDE, DE, PSO, and GA. On F10, the LCA outperformed all the compared algorithms except the

RUN algorithm. In summary, the comprehensive analysis and comparison demonstrate that the proposed LCA algorithm performs better than other optimization algorithms in most benchmark functions. Its robustness, efficiency, and consistency in finding optimal solutions make it an auspicious approach to tackle complex optimization problems.

Additionally, Table 3 reports the results for CEC'2020 with dimension 20. The Liver Cancer Algorithm (LCA) consistently outperforms all other algorithms in functions, achieving the lowest mean fitness values

with remarkable stability (indicated by low standard deviations). Its superiority is evident as it consistently achieves the lowest mean fitness values for most functions, showcasing its exceptional performance. In particular, LCA competes in the functions F1, F2, F4, F5, F6, F7, F8, F9, and F10, which achieve the lowest mean fitness values among all algorithms. The consistently low or negligible standard deviations of LCA imply that it consistently produces highly stable and reliable results. The CHIO algorithm demonstrates competent and consistent performance across most functions, with relatively low mean fitness values. It stands out on F3, achieving the lowest mean fitness value among all algorithms for this particular function. Furthermore, it exhibits a mean value similar to that of LCA in F5. The INFO algorithm displays consistent and competitive performance in the F3, F4 and F8 functions. On the other hand, the RUN algorithm performs admirably on F2, F6, and F10, attaining competitive mean fitness values for these functions. The HHO algorithm also shows competitiveness on the F6 function. Interestingly, the IMODE algorithm performs comparable to LCA on F1 but shows relatively poor performance on F2 and F3, with higher mean fitness values than other algorithms. The DE algorithm demonstrates more robust performance on the F3 and F6 functions. However, the GA algorithm exhibits poor performance on F1, as indicated by its high mean fitness value, suggesting that it requires improvement in finding the optimal solution for F1. On the other hand, GA performs well in F9 and F10.

Overall, the experimental results underscore the exceptional optimization capabilities of the proposed LCA algorithm on the CEC'20 benchmark functions with dimension 20. Across various test functions, it consistently outperforms other state-of-the-art algorithms (GA, PSO, DE, AGDE, IMODE, HHO, RUN, INFO, and CHIO). The innovative approach of the LCA algorithm, inspired by the growth of liver tumors, enables it to effectively balance local and global search, leading to efficient exploration and exploitation of the search space. The comprehensive comparison highlights LCA's potential as a valuable addition to the field of bio-inspired optimization algorithms. Its robust performance in various test functions showcases its versatility and applicability to real-world problem solving, particularly in the critical medical imaging, finance, and engineering domains. However, the results also provide insights into areas for potential improvement in the other compared algorithms. Future research efforts could enhance or hybridize these algorithms with the LCA approach to achieve even more promising results. In conclusion, the Liver Cancer Algorithm is a novel and promising contribution to bio-inspired optimization. It offers efficient and effective solutions to complex optimization problems, exemplified by its outstanding performance on the CEC'20 benchmark functions.

4.4. Convergence analysis

The performance of the LCA algorithm and other comparative MAs for the CEC'2020 test functions for dimensions 10 and 20 are shown in [Figs. 5](#) and [6](#), respectively. [Fig. 5](#) illustrates the performance of the LCA algorithm and nine other MAs for the CEC'2020 test functions using convergence curves and their counterparts. A stable point was reached for all functions tested in the proposed algorithm, showing that it quickly converges to the best solution. The LCA algorithm outperformed the other algorithms for F1, F4, F5, F6, F7, and F8. However, for F2, GA was found to be the best, while the CHIO and INFO algorithms were the best for F3. For F9 and F10, the best performance was achieved by the RUN and INFO algorithms, respectively. These experimental results, supported by the convergence curves, demonstrate that the LCA algorithm significantly enhances search capabilities and has advantages over the other algorithms tested.

Furthermore, [Fig. 6](#) presents the convergence curves of the CEC'2020 test functions for dimension 20. In [Fig. 6\(a\)](#), the convergence curves for the F1 function, which represents a uni-modal space, clearly demonstrate the superior performance of the LCA algorithm. Moving on to the multimodal functions (F2 to F4), depicted in [Figs. 6\(b-d\)](#), we

observe that the RUN algorithm performs better for F2. In contrast, the CHIO algorithm outperforms other algorithms on the F3 function. In particular, LCA exhibits better performance in the F4 function. Furthermore, the convergence curves for hybrid functions (F5, F6, and F7), presented in [Figs. 6\(e-g\)](#), show that LCA achieves remarkable results. It delivers significant performance in F5 and F7, surpassing other algorithms. However, for the F6 function, all algorithms exhibit similar performance levels. Moving to the composition functions (F8, F9, and F10) in [Figs. 6\(h-j\)](#), LCA shows relative efficiency in solving problems in complex spaces. It demonstrates competitive performance in these composition functions. The convergence curves validate the superiority of the LCA algorithm across various test functions. It outperforms other algorithms consistently and effectively handles uni-modal, multimodal, hybrid, and composition functions, making it a versatile and robust optimization method.

4.5. Boxplot analysis

To visually represent the characteristics of the data distribution, we utilized boxplot analysis. In combination with the results presented in [Table 2](#) and [Table 3](#), we used boxplots to gain further insight into the data distributions. Boxplots are effective graphical tools to display data distributions in quartiles. The edges of the boxplots represent the minimum and maximum data points, serving as the lower and upper whiskers. The middle of the rectangle within the boxplot denotes the interquartile range, which separates the lower and upper quartiles. A narrow boxplot signifies a high level of data agreement and consistency. For clarity, [Fig. 7](#) illustrates the boxplots for functions F1 to F10 with a solution dimension of $Dim = 10$. [Fig. 8](#) presents the boxplots for functions F1 to F10 with a solution dimension of $Dim = 20$. These boxplots provide a visual representation of the spread and distribution of the data, allowing for a comprehensive understanding of the algorithm's performance across the benchmark functions.

In all the functions tested, the boxplots of the proposed LCA algorithm exhibit optimal and lower values. Boxplots are considerably narrow compared to the distributions of other algorithms. This indicates that the proposed LCA algorithm performs exceptionally well. In fact, the results reveal that the proposed LCA algorithm outperforms other algorithms in most of the test functions. These findings suggest that the proposed LCA algorithm is a promising approach for solving the optimization problems. Further research can explore the potential of the LCA algorithm in other domains and real-world applications.

4.6. Qualitative metrics discussion

Monitoring the actions of particles or search agents can yield important information in the algorithm convergence and optimization search process. [Fig. 9](#) presents the qualitative analysis of the LCA algorithm, showing the behavior of agents in a two-dimensional function space. The figure includes convergence curves, search history, and average fitness history, which comprehensively illustrate the search process. By analyzing these behaviors, we can gain a deeper understanding of the performance and effectiveness of the LCA algorithm.

The following points provide an explanation of the qualitative analysis:

- *Regards the domain's topology:*

[Fig. 9](#) presents the functions viewed in the two-dimensional area in the first column. These functions exhibit different typologies that are important to identify the optimal performance shape of the function for the algorithm. By analyzing the typology of the function, the algorithm can determine the most efficient approach to optimize the function. Therefore, it is crucial to understand the typology of the function in developing practical optimization algorithms.

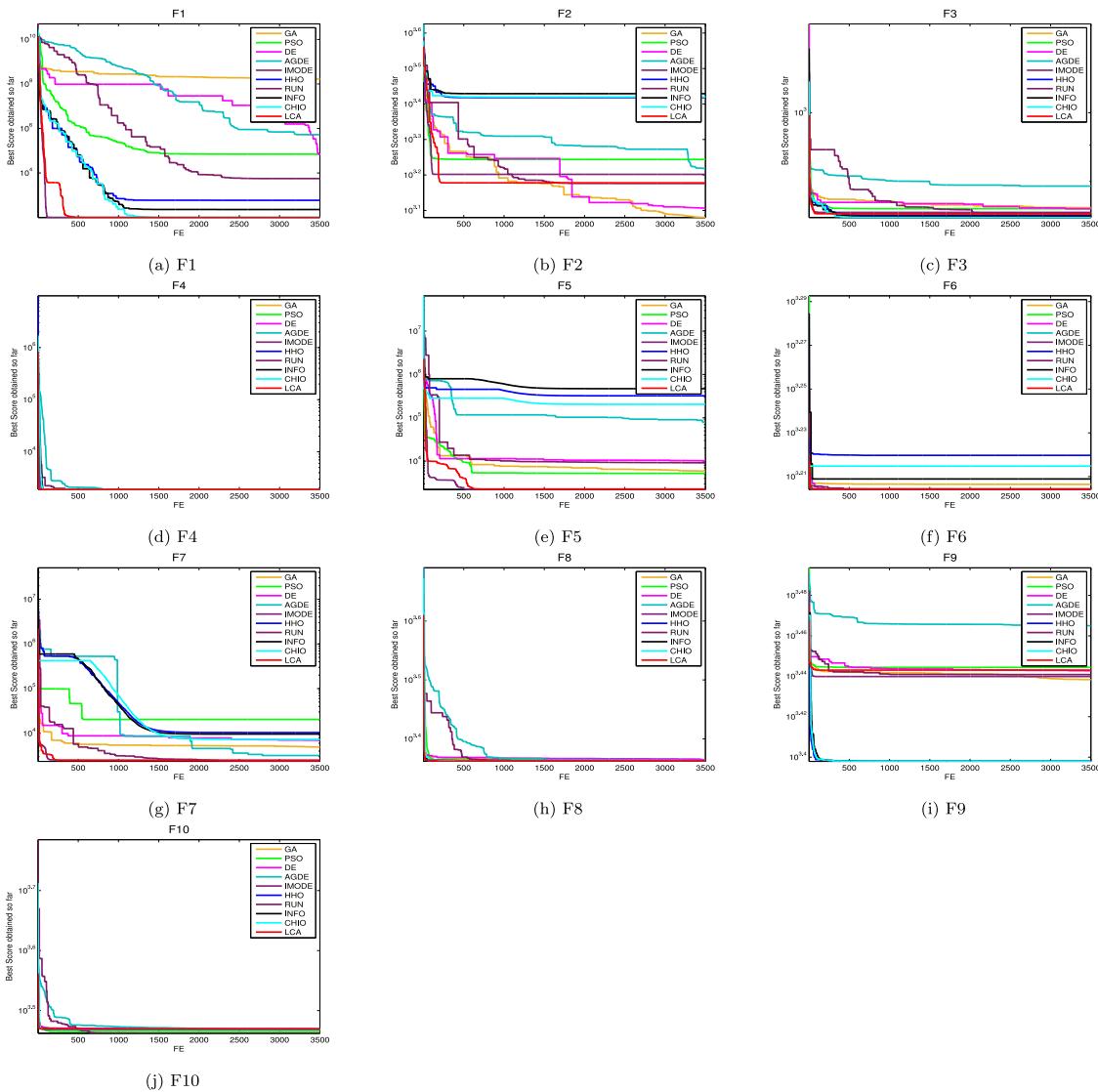


Fig. 5. The convergence graphs for the LCA algorithm and compared algorithms on CEC'2020 test suite with $Dim = 10$.

- *Regards the search history:* The second column of Fig. 9 displays the search history of the agents, spanning from the initial to the final iteration. Counter lines are used to represent the search space, and a color gradient from blue to red indicates better fitness levels. The search history shows that the LCA algorithm can successfully identify areas with the lowest fitness values for specific functions. This implies that the LCA algorithm can search the solution space efficiently and identify optimal solutions. Therefore, the visualization of search history provides valuable insight into the performance and effectiveness of the LCA algorithm.
- *Regards the average fitness history:* Fig. 9 shows the average fitness history, the average fitness value over time, represented in the third column. This average provides valuable information about the performance of the agents and their contribution to the optimization process. Each curve in the graph shows a steady decline over time, showing that the population is growing with each iteration. This improvement is evidence of cooperative-seeking behavior among agents, and it supports the use of particle law updates. In other words, as the fitness values of the population continue to decrease over time, agents are working together effectively to find better solutions.
- *Regarding the optimization history:* The fourth column of Fig. 9 shows the optimization improvement, which tracks the fitness

results of 100 iterations for each experiment to demonstrate how fitness improves over time. The curves in this graph show a steady decrease, indicating that the LCA algorithm (the optimization algorithm that is being used) works well with the agents to determine the best possible solution. This suggests that the agents are effectively collaborating throughout the optimization process.

- *Regarding the diversity metric:* The final column of Fig. 9 displays the diversity plot, which indicates the average distance traveled by the agents through the optimization process. This graph provides information about the agents' exploration during the search for the best solution. By measuring the average distance covered, we can get an idea of how diverse the solutions explored by the agents are.

5. Application of LCA: Feature selection for biomedical data classification

Feature selection (FS) is an essential step in model development, filtering, and selecting relevant features from a dataset to improve model training, prevent overfitting, and improve generalization. Medical datasets often have high dimensions, and FS has shown positive results in various medical domains. ML employs three common FS techniques: filter, wrapper, and embedding methods. Dealing with

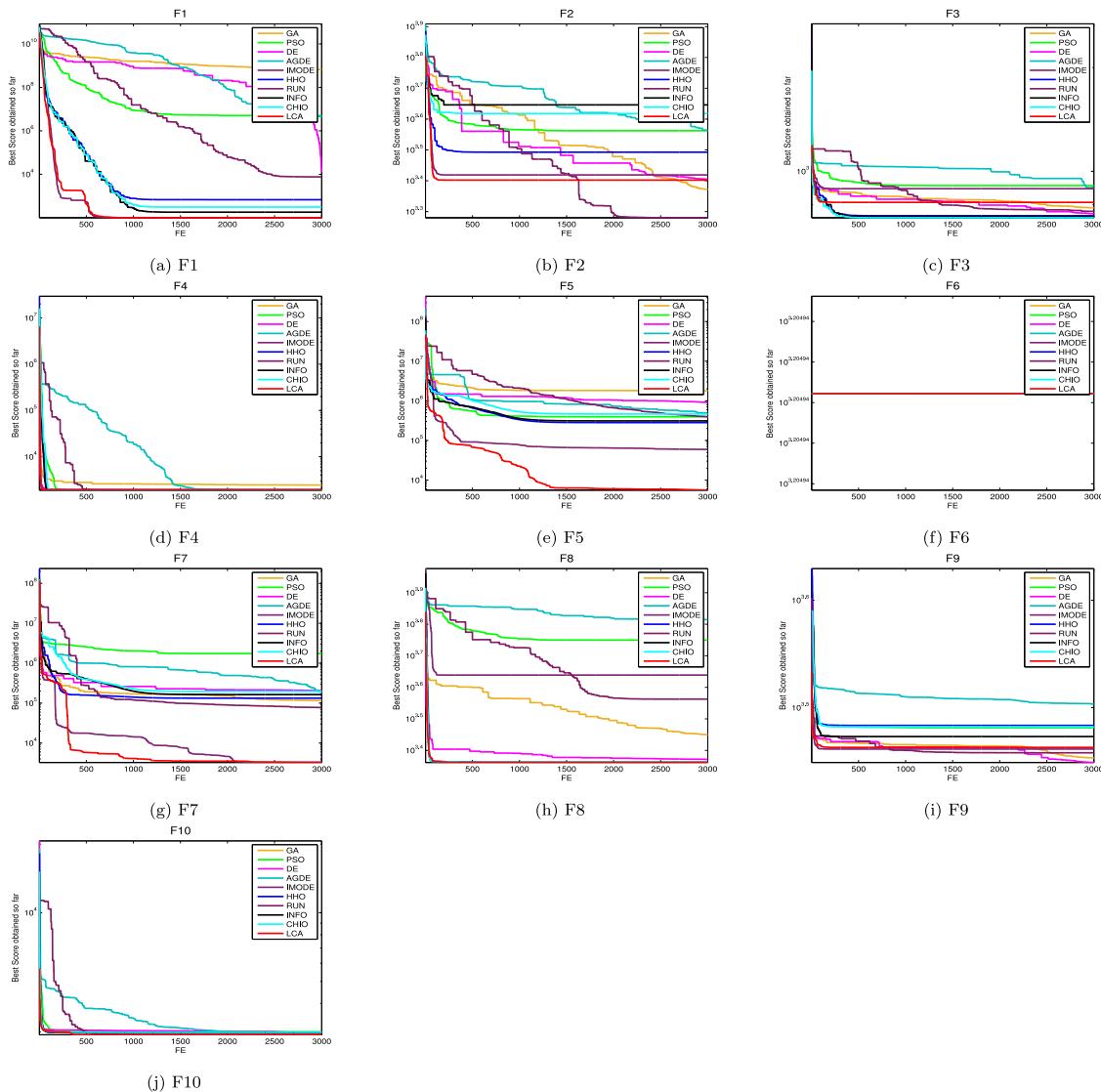


Fig. 6. The convergence graphs for the LCA algorithm and compared algorithms on CEC'2020 test suite with $Dim = 20$.

large feature spaces presents challenges, but binary variants of MAs offer solutions, efficiently exploring search spaces to identify optimal feature subsets. Incorporating MAs in FS enhances the accuracy and performance of the model. In FS, various approaches involve selecting appropriate features, evaluating subsets through different measures, identifying various subsets, and validating features [63,75]. Wrapper approaches yield more precise results, but take longer to complete than filter methods, while hybrid approaches combine the strengths of both. MAs' fitness function assesses the quality of each feature collection, which is crucial for FS and machine learning algorithms. Researchers have focused on enhancing these systems, employing integrated MAs to improve classification accuracy and reduce the size of the search space [76].

While MAs are commonly used, their limitations can have an impact on accuracy and overall system performance. Striking a balance between exploration and exploitation is crucial to optimizing MAs and obtaining optimal solutions. To address these challenges, we used the Liver Cancer Algorithm (LCA) for FS and optimized the support vector machine (SVM) parameters to improve the classification accuracy. This section delves into the development of the LCA algorithm in conjunction with the SVM to address challenges in the classification of biomedical data and FS. Using the strengths of MA and SVM, the proposed approach LCA-SVM aims to improve the effectiveness and

efficiency of feature selection for medical research and diagnosis. This section is organized as follows: Section 5.1 presents the hybrid of LCA with SVM for FS and classification. Section 5.1.1 provides the fitness function for FS. Section 5.1.2 presents the FS process. Section 5.1.3 provides the computational complexity of the proposed LCA-SVM classification technique. Section 5.2 presents the results of the proposed LCA-SVM classification technique.

5.1. Integrated LCA algorithm with SVM (LCA-SVM) approach for feature selection

The SVM class of ML algorithms utilizes kernel functions to transform data into an optimal solution. It can perform both regression and classification tasks and is essentially a linear model [77]. The SVM algorithm draws a line to separate the data into different classes, with a certain distance from the nearest point. The purpose is to determine the optimal line, and the algorithm's results are affected by parameters set during development, which can be adjusted for better performance. The parameter C controls the balance between correctly identifying the training points and a smooth decision boundary, while Gamma determines the influence range of each training session. SVM is widely used in bioinformatics and can be applied to predict various toxicities and diseases, such as mutagenic toxicity [78].

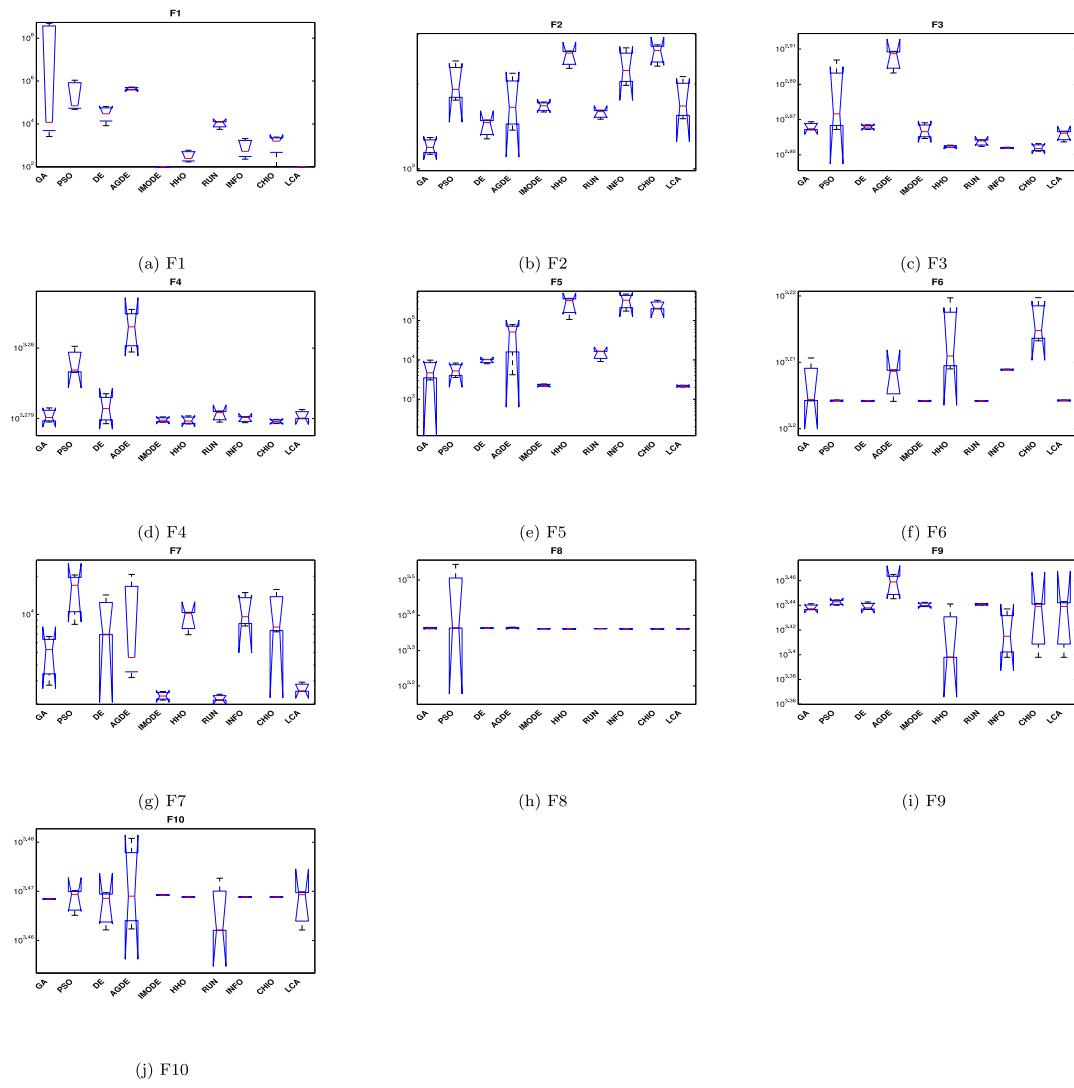


Fig. 7. The boxplot of the compared algorithms on CEC'2020 test suite with $Dim = 10$.

The SVM and LCA algorithm are utilized together for the purposes of FS, parameter optimization, and classification. The LCA algorithm serves as a wrapper methodology for FS, with the resulting set of features being utilized by the SVM. Through the use of cross-validation folds, the position of a tumor within the liver can be identified and classified accordingly.

The flowchart of the proposed LCA-SVM method is shown in Fig. 10 including three phases: (1) preprocessing, (2) FS and optimization, and (3) classification and validation [79]. The position vector is modified using Eq. (15).

$$Position(t+1) = \begin{cases} Y & \text{if } LF(fit(Y)) < LF(fit(Position(t))) \\ Z & \text{if } LF(fit(Z)) < LF(fit(Position(t))) \end{cases} \quad (15)$$

5.1.1. Fitness function for FS

To guarantee their efficacy, the solutions produced by the LCA algorithm require continuous evaluation during the iterative process. The LCA-SVM classifier achieves this by utilizing the fitness function (fit), defined by Eqs. (16), (17), and (18):

$$fit = \varpi + \beta \frac{|R|}{|C|} - G \quad (16)$$

$$\beta = \omega \quad (17)$$

$$fit > T \quad (18)$$

According to Eq. (16), the variables R , C , ϖ , and β have specific meanings. R represents the classification error rate, while C represents the total number of features in the dataset. ϖ and β are two factors used to calculate the relative weights of classification quality and subset length, respectively, as determined by the classifiers. The value of α ranges from 0 to 1. The fitness of each algorithm is assessed by comparing it to the fitness function, represented by the group column G and the condition T . To maximize the solution, the value of fit must be greater than T .

5.1.2. Feature selection process

A preprocessing step known as FS is frequently used before ML algorithms to choose a subset of features that is clear of duplicates. Various ML approaches can increase prediction accuracy and better understand the data by choosing highly associated with other features. This indicates that only one feature is required to characterize the data when two features are adequately associated. Since there are $2N$ possible feature combinations for a feature vector of size N , this large space is difficult to fully investigate. Consequently, the number of features makes it difficult to directly evaluate an NP problem [80, 81]. Selecting the best features to use is made difficult by the broad search space. So many feature selection problems are solved with metaheuristic algorithms. As shown in Fig. 10, the LCA algorithm is modified in this section to choose the important features and perform

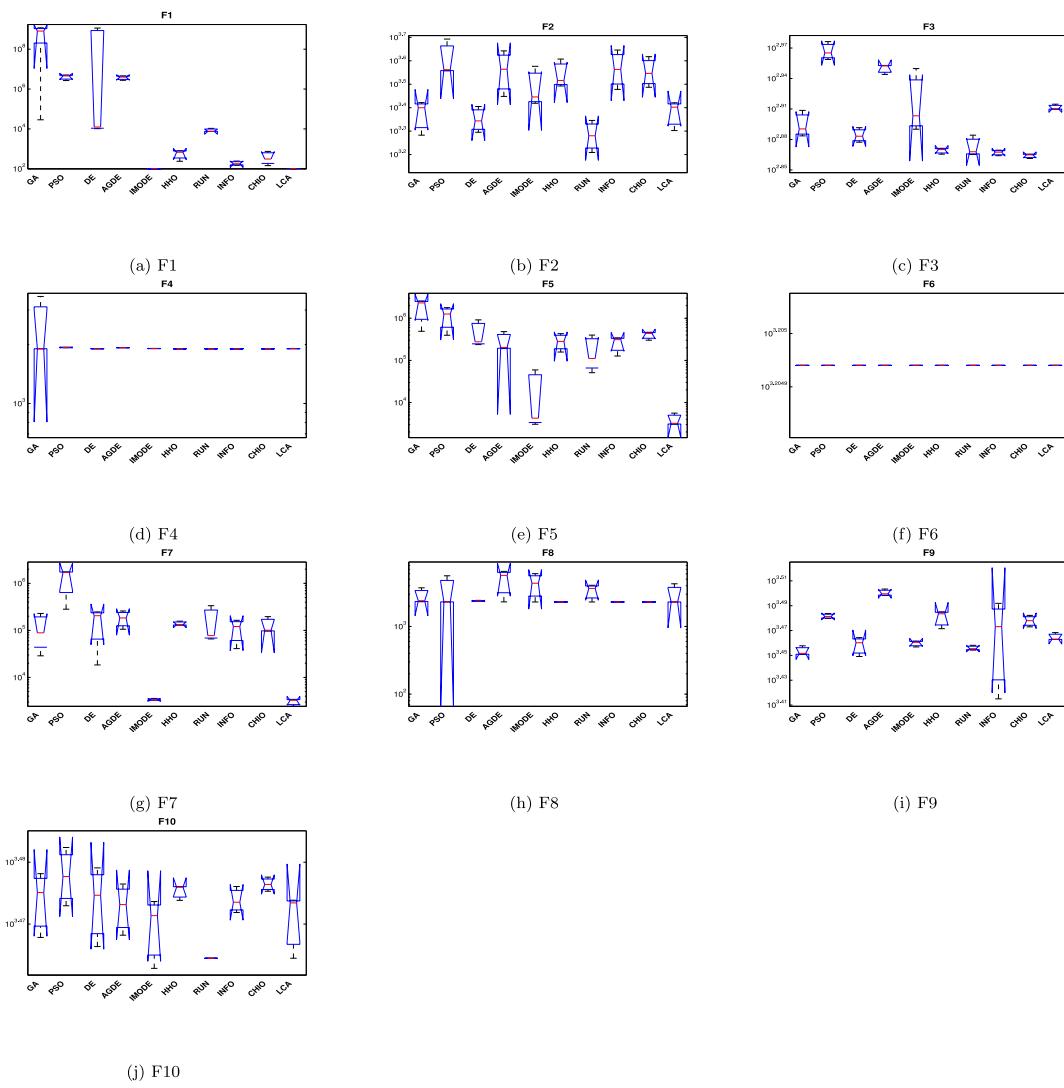


Fig. 8. The boxplot of the compared algorithms on CEC'2020 test suite with $\text{Dim} = 20$.

an adaptive search in the feature space to choose the optimum feature subset. The optimal solution should contain the fewest possible selected features, the highest classification accuracy possible, and the lowest classification error rate possible.

5.1.3. Complexity of the LCA-SVM

The SVM classifier used in the LCA-SVM approach has a complexity of $O(k^3)$, where k represents the amount of data [82]. When combined with the LCA algorithm, the overall time complexity of the LCA-SVM approach can be expressed as $(N \times (T + TD + 1)) \times O(k^3)$. In the LCA-SVM approach, N represents the number of tumors, T is the number of iterations in the LCA algorithm, and D is the dimension of the problem. As the LCA-SVM approach involves multiple iterations and updating the positions of tumors in each iteration, the computational complexity can be affected by the number of tumors, iterations, and the dimension of the problem.

5.2. Performance of LCA algorithm on biomedical application

In this experiment, we evaluated the performance of our proposed LCS-SVM model using various datasets. These datasets were sourced from the UCI repository¹ and the MOA dataset was obtained from

cheminformatic.org.² By applying the LCA-SVM model to these diverse datasets, our objective was to assess its effectiveness in different scenarios and validate its applicability across various domains. The utilization of publicly available datasets ensures transparency and allows fair comparisons with other approaches to feature selection. The description of the used datasets is reported in Table 4.

5.2.1. Statistical results analysis

This section presents a comprehensive analysis of the proposed liver cancer algorithm (LCA) to address the FS and classification problem for biomedical data. The LCA algorithm is compared with several other algorithms, including PSO, IMODE, AGDE, HHO, INFO, RUN, and CHIO, to evaluate its performance in various datasets. The results shown in Tables 5, 6, 7, and 8 demonstrate the superiority of the LCA algorithm in solving the FS problem. Across different datasets, the LCA algorithm consistently outperforms its competitors in terms of mean, best, and STD values, indicating its effectiveness in achieving higher classification accuracy and robustness.

In Table 5, which includes the results for the MAO, QSAR biodegradation, and drug review datasets, we observe the effectiveness of the LCA algorithm in achieving the highest mean and best accuracy

¹ <https://archive.ics.uci.edu/ml/index.php>

² <https://brunl01.users.greyc.fr/CHEMISTRY/>

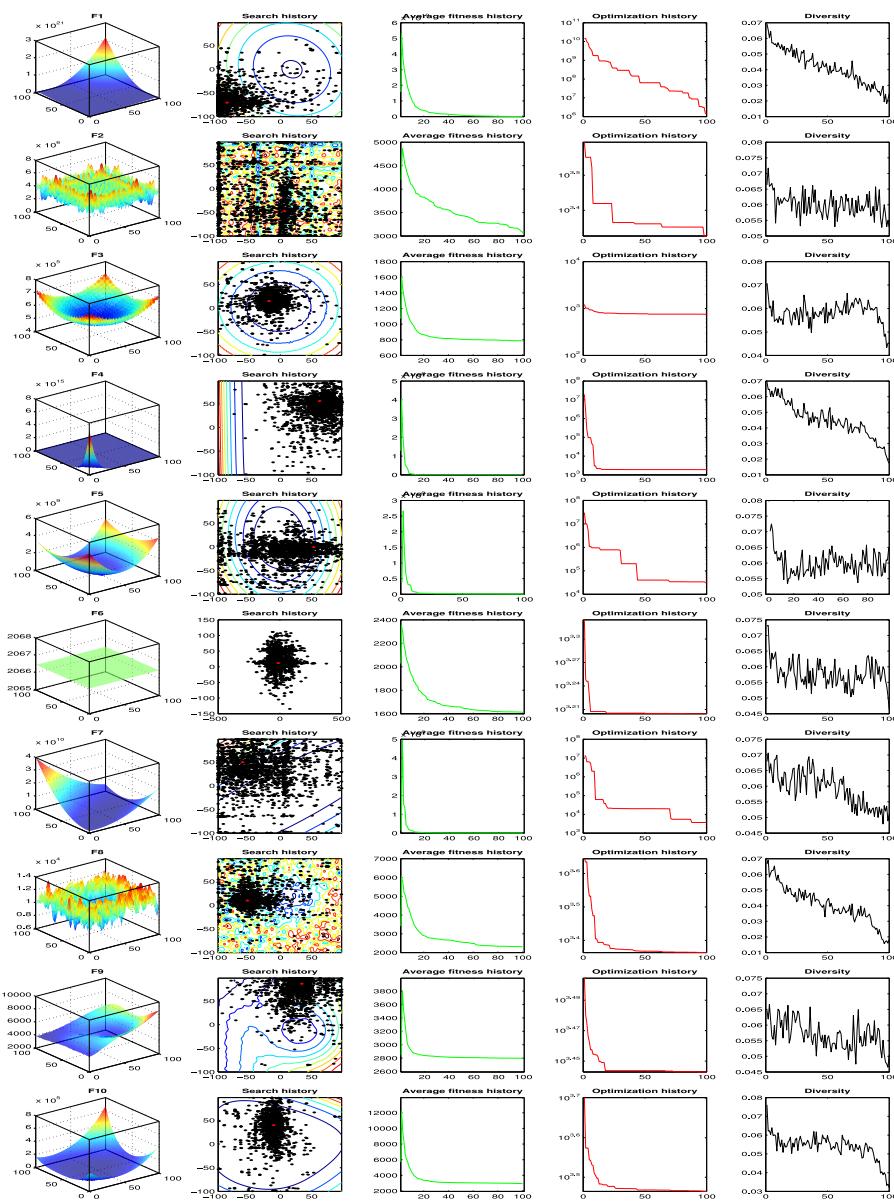


Fig. 9. Qualitative Metrics on CEC'20 Test Suite.

Table 4
Definition for datasets.

Dataset	Features number	Instance number
Monoamine Oxidase (MAO)	1665	68
Quantitative Structure–Activity Relationship (QSAR) Biodegradation	1055	41
Drug Review	6	2,15,063
Immunotherapy	8	90
QSAR androgen receptor	1024	1687
Gene Expression Cancer RNA-Seq	20,531	801
Anticancer Petitude lung cancer	3	902
Drug consumption	32	1885
Primary Tumor	17	339
Hepatitis C Virus (HCV) for Egyptian patients	29	1385
Liver Disorders	7	345
Hepatocellular Carcinoma dataset (HCC)	49	165

values in the three datasets, making it a competitive choice for feature selection and classification tasks in biomedical data. The second-best performing algorithm, HHO, shows competitive results in the MAO and Drug Review datasets but lags behind in the QSAR biodegradation dataset. On the contrary, the IMODE algorithm ranks the lowest in terms of mean and best values, indicating its limited performance in all three datasets. The low STD values observed across all algorithms suggest that they produce consistent results. Additionally, CPU time values vary among algorithms, with INFO having the lowest CPU time on all three datasets and PSO having the highest CPU time on the QSAR Biodegradation dataset. The results suggest that the LCA algorithm performs well on all three datasets, with HHO being the second-best performer. The algorithms exhibit consistent results with low STD values, but their computational efficiency differs, as indicated by their CPU time values.

Moving to Table 6, which presents the mean values for the drug consumption, QSAR androgen receptor, and immunotherapy datasets, we observe that the LCA algorithm consistently achieves the highest accuracy. The best and worst accuracy values for each dataset provide insight into the range of accuracies achieved by each algorithm. In

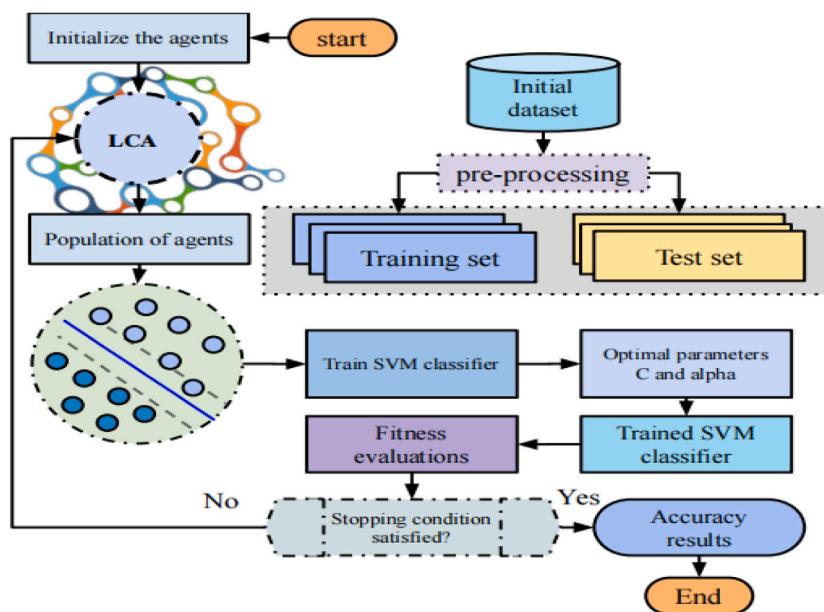


Fig. 10. Hybrid LCA-SVM Flowchart.

Table 5

Mean, STD, Best, Worst, and CPU time achieved results using SVM with compared algorithms depended on FE on MonoAmine Oxidase (MAO), QSAR Biodegradation and Drug Review datasets.

Algorithm	Mean	STD	Best	Worst	CPU time
Dataset 1: MonoAmine Oxidase (MAO)					
LCA	9.86E+01	1.50E-02	98.704	94.500	0.366
CHIO	9.14E+01	1.21E+00	93.885	88.523	0.443
PSO	9.11E+01	1.29E+00	88.065	86.014	0.781
AGDE	9.10E+01	2.87E-01	90.180	90.100	0.819
HHO	9.74E+01	8.71E-02	96.597	94.464	0.470
INFO	9.62E+01	2.99E-01	94.190	92.600	0.872
RUN	9.02E+01	4.17E-01	92.150	88.276	0.559
IMODE	8.88E+01	4.56-E00	87.854	86.886	0.891
Dataset 2: QSAR Biodegradation					
LCA	9.01E+01	2.07E-04	91.101	90.105	0.420
CHIO	8.60E+01	1.08E-01	85.130	84.190	0.620
PSO	8.22E-01	6.45E-01	78.578	74.178	0.893
AGDE	8.90E+01	1.07E-00	88.790	87.490	0.707
HHO	8.64E-01	9.23E-04	84.512	83.200	0.576
INFO	9.00E+01	4.86E-02	90.190	89.700	0.761
RUN	8.80E+01	4.20E-00	88.005	87.030	0.800
IMODE	8.10E+01	3.30-E00	86.920	85.180	0.970
Dataset 3: Drug Review					
LCA	9.10E+01	1.05E-02	92.909	91.110	0.120
CHIO	8.66E+00	1.08E-01	87.010	86.071	0.600
PSO	8.67E+00	1.00E-00	84.100	84.110	0.910
AGDE	8.71E+01	1.16E-00	86.180	85.190	0.377
HHO	8.80E+00	1.10E-02	88.110	86.081	0.410
INFO	9.00E+01	1.87E-02	90.080	89.712	0.370
RUN	8.88E+01	1.18E-00	88.052	87.100	0.860
IMODE	8.35E+01	1.00-E00	84.021	84.171	0.961

particular, the LCA algorithm demonstrates high accuracy, even with challenging datasets, as evidenced by achieving the best accuracy of 95.14% and the worst accuracy of 91.01% on dataset No. 6. On the other hand, the CHIO algorithm's best accuracy is 92.12%, and its

worst accuracy is 88.17% on dataset No. 4, indicating its relatively less consistent performance. Regarding CPU time, the HHO algorithm is the fastest, with an average runtime of 0.4 s per dataset, while the PSO algorithm is the slowest, with an average runtime of 0.9 s. However, it is essential to consider the context in which the algorithms are used, as these differences in CPU time may not be significant.

Table 7 presents the results for three datasets: HCV Egypt, Gene Expression Cancer RNA-Seq, and Primary Tumor. For the HCV Egypt dataset, the LCA algorithm achieved the highest mean accuracy of 85.5%, followed by AGDE with 86.0%. PSO algorithm had the lowest mean accuracy of 80.0%. The CPU time for all algorithms ranged from 0.1 to 0.5 s. For the Gene Expression Cancer RNA-Seq dataset, the LCA algorithm obtained the highest mean accuracy of 99.0%, followed by CHIO with 97.0%. PSO algorithm had the lowest mean accuracy of 90.7%. The CPU time for all algorithms ranged from 0.3 to 0.9 s. For the Primary Tumor dataset, the LCA algorithm achieved the highest mean accuracy of 98.5%, followed by INFO with 98.0%. PSO algorithm had the lowest mean accuracy of 9.65%. The CPU time for all algorithms ranged from 0.1 to 0.3 s. Based on the table provided, we can see that the performance of the selected algorithms varies depending on the dataset being used. For example, on dataset No. 7 (HCV Egypt), LCA and IMODE performed the best with mean fitness values of 8.55E+01 and 8.80E+01, respectively. On the other hand, for Dataset 8 (Gene Expression Cancer RNA-Seq), LCA and CHIO performed the best with mean fitness values of 9.90E+01 and 9.70E+01, respectively. Finally, on dataset No. 9 (Primary Tumor), LCA and INFO performed the best with mean fitness values of 9.85E+01 and 9.80E+01, respectively. The results demonstrate that the performance of the LCA algorithm varies depending on the dataset being used, and it excels on two out of three datasets while still performing competently on the third dataset. Taking into account the overall performance across all datasets, the LCA algorithm emerges as the best among the selected algorithms, showcasing its versatility and competitive performance.

Moving to Table 8, we examine the performance on the Alzheimer, HCC survival and liver datasets. For dataset No. 10, the Alzheimer dataset, LCA obtained the highest mean accuracy value of 98.0% with the lowest STD value of 0.0017. The PSO algorithm obtained the lowest mean accuracy value of 94.0% with an STD value of 0.15. LCA obtained the best accuracy value of 92.220%, and the PSO algorithm obtained the worst accuracy value of 84.120%. Regarding CPU time, HHO obtained the lowest CPU time value of 0.1 s, while the PSO

Table 6

Mean, STD, Best, Worst, and CPU time achieved results using SVM with compared algorithms depend on FE on Drug consumption, QSAR androgen receptor and Immunotherapy datasets.

Algorithm	Mean	STD	Best	Worst	CPU time
Dataset 4: Drug consumption					
LCA	9.70E+01	1.99E-02	93.100	91.011	0.300
CHIO	8.75E+00	1.06E-01	88.111	87.160	0.600
PSO	8.74E+00	1.00E-00	87.112	87.100	0.811
AGDE	8.60E+01	1.17E-00	86.010	86.010	0.478
HHO	9.40E+00	1.60E-02	90.021	88.180	0.400
INFO	9.50E+01	1.88E-02	91.181	90.110	0.300
RUN	8.81E+01	1.11E-00	89.150	86.120	0.600
IMODE	8.50E+01	1.00-E00	86.120	85.080	0.900
Dataset 5: QSAR androgen receptor					
LCA	9.40E+01	1.00E-02	94.160	92.010	0.220
CHIO	9.04E+01	1.28E-01	90.100	89.100	0.620
PSO	9.00E+00	1.00E-01	88.100	88.109	0.901
AGDE	8.89E+01	1.76E-01	87.019	87.210	0.328
HHO	9.10E+00	1.70E-02	91.220	90.100	0.410
INFO	9.10E+01	1.99E-02	92.190	91.100	0.300
RUN	8.81E+01	1.11E-00	87.150	87.120	0.820
IMODE	8.50E+01	1.20-E00	87.120	86.080	0.901
Dataset 6: Immunotherapy					
LCA	9.50E+01	1.20E-02	95.140	92.102	0.120
CHIO	9.55E+01	1.80E-01	91.221	90.121	0.600
PSO	9.40E+00	1.91E-01	90.121	89.120	0.800
AGDE	8.95E+01	1.79E-01	89.110	86.011	0.223
HHO	9.350E+00	1.89E-02	93.100	91.121	0.400
INFO	9.40E+01	2.19E-02	94.091	92.101	0.201
RUN	8.90E+01	1.91E-01	88.150	87.120	0.721
IMODE	8.69E+01	1.88-E01	86.011	85.172	0.711

algorithm obtained the highest CPU time value of 0.24 s. For data set 11, the HCC survival dataset, LCA also obtained the highest mean accuracy value of 98.0% with an STD value of 0.0108. In comparison, the PSO algorithm obtained the lowest mean accuracy value of 96.0% with an STD value of 0.16. LCA obtained the best accuracy value of 93.500%, and the PSO algorithm obtained the worst accuracy value of 85.100%. Regarding CPU time, HHO obtained the lowest CPU time value of 0.15 s, while the PSO algorithm obtained the highest CPU time value of 0.30 s. For dataset No. 12, the Indian Liver Patient dataset, LCA again obtained the highest mean accuracy value of 88.0% with an STD value of 0.17. The PSO algorithm obtained the lowest mean accuracy value of 86.5% with an STD value of 1.60. LCA obtained the best accuracy value of 83.20%, and the worst accuracy value of 80.10% was obtained by the CHIO and PSO algorithms with respect to CPU time, the LCA, HHO, and CHIO algorithms obtained the lowest CPU time value of 0.10 s, while the PSO algorithm obtained the highest CPU time value of 0.14 s. Overall, the LCA algorithm obtained the highest mean accuracy value for all three datasets and the best accuracy value for two of the three datasets. Furthermore, the LCA algorithm obtained the lowest STD value for two of the three datasets, indicating that its performance is consistent between different runs. The PSO algorithm generally got the lowest accuracy values and the highest STD and CPU time values for all three datasets. The HHO algorithm obtained the lowest CPU time value for the three datasets, indicating that it is computationally efficient.

In summary, the proposed LCA algorithm proves superior in resolving the feature selection and classification issue for biological data in the end. Across numerous datasets, it consistently outperforms other algorithms' accuracy and consistency, demonstrating its competitive

Table 7

Mean, STD, Best, Worst, and CPU time achieved results using SVM with compared algorithms depend on FE on HCV Egypt, Gene Expression Cancer RNA-Seq and Primary Tumor datasets.

Algorithm	Mean	STD	Best	Worst	CPU time
Dataset 7: HCV Egypt					
LCA	8.55E+01	1.40E-02	88.160	86.310	0.100
CHIO	8.35E+01	1.80E-02	85.120	84.100	0.300
PSO	8.00E+00	1.81E-01	82.100	81.100	0.500
AGDE	8.60E+01	1.72E-01	84.210	83.100	0.220
HHO	8.40E+00	1.99E-02	85.100	84.102	0.200
INFO	8.30E+01	2.29E-01	84.120	82.120	0.250
RUN	8.20E+01	1.98E-01	84.150	83.100	0.220
IMODE	8.70E+01	1.90-E00	82.002	81.190	0.400
Dataset 8: Gene Expression Cancer RNA-Seq					
LCA	9.90E+01	2.00E-02	98.160	97.240	0.400
CHIO	9.70E+01	1.99E-01	94.022	92.120	0.700
PSO	9.70E+00	1.86E-01	92.100	91.120	0.900
AGDE	8.90E+01	1.70E-01	89.210	88.102	0.600
HHO	9.80E+00	2.18E-02	96.100	95.100	0.300
INFO	9.30E+01	2.33E-02	97.112	96.330	0.500
RUN	8.40E+01	1.77E-01	89.020	88.130	0.800
IMODE	8.76E+01	1.89-E00	85.100	86.200	0.700
Dataset 9: Primary Tumor					
LCA	9.85E+01	1.79E-02	95.061	94.150	0.100
CHIO	9.65E+01	1.56E-01	92.020	90.220	0.200
PSO	9.64E+00	1.59E-02	91.430	90.520	0.150
AGDE	8.48E+01	1.56E-02	90.400	88.400	0.200
HHO	9.74E+00	1.75E-01	92.020	91.040	0.100
INFO	9.79E+01	1.44E-01	93.012	92.310	0.300
RUN	8.94E+01	1.89E-01	88.320	87.100	0.200
IMODE	8.70E+01	1.79-E00	87.110	87.800	0.300

performance. The LCA algorithm is a promising method for biomedical data analysis and classification tasks because of its robustness and scalability, evident in its ability to achieve high accuracy in various datasets. However, as with any algorithm, there may be certain limitations and scope for additional improvement, which might be addressed in future research.

To evaluate the effectiveness of the proposed LCA algorithm, we performed two important statistical tests: the Wilcoxon rank-sum test and the Average Selection Size of Features (ASS) test. These tests provide crucial insights into the algorithm's performance compared to other state-of-the-art feature selection algorithms.

- Wilcoxon rank-sum test: The Wilcoxon rank-sum test was employed to determine the statistical significance of the findings obtained by various algorithms, including the proposed LCA algorithm. The test's p-values, shown in Tables 9, 10, 11, 12, 13, and 14 based on accuracy measure, offer valuable information regarding the algorithm's performance. Interestingly, the LCA algorithm achieved p-values less than 1% for both datasets, indicating its significant advantage over the other feature selection algorithms. This statistical analysis solidifies the superiority of the LCA algorithm in comparison to the other optimizers studied in this investigation.
- Average Selection Size of Features (ASS): The ASS test was utilized to assess the number of features selected by each algorithm during iterations. The ASS value is calculated as the average ratio of the number of selected features to the total number of features in the initial dataset. The formula for ASS is given as:

$$ASS = \frac{1}{M} \sum_{i=1}^M \frac{\text{length } (Q_i)}{L} \quad (19)$$

Table 8

Mean, STD, Best, Worst, and CPU time achieved results using SVM with compared algorithms depend on FE on Alzheimer Features, HCC Survival and Indian Liver Patient datasets.

Algorithm	Mean	STD	Best	Worst	CPU time
Dataset 10: Alzheimer Features					
LCA	9.80E+01	1.70E-03	92.220	91.201	0.100
CHIO	9.60E+01	1.40E-02	88.120	86.020	0.200
PSO	9.40E+00	1.50E-01	86.132	84.120	0.240
AGDE	8.40E+01	1.30E-01	88.010	87.120	0.300
HHO	9.60E+00	1.34E-02	91.011	90.100	0.100
INFO	9.79E+01	1.49E-02	91.200	90.501	0.100
RUN	8.79E+01	1.50E-00	88.320	86.920	0.300
IMODE	8.74E+01	1.69-E00	86.110	85.950	0.300
Dataset 11: HCC Survival					
LCA	9.80E+01	1.08E-02	93.500	92.241	0.100
CHIO	9.70E+01	1.38E-02	89.220	88.201	0.200
PSO	9.60E+00	1.55E-01	86.100	85.100	0.300
AGDE	8.50E+01	1.80E-01	88.200	87.200	0.300
HHO	9.50E+00	1.49E-02	90.240	89.100	0.150
INFO	9.40E+01	1.30E-02	91.200	90.500	0.300
RUN	8.65E+01	1.80E-01	87.021	86.320	0.200
IMODE	8.70E+01	1.70-E00	86.100	85.150	0.300
Dataset 12: Indian Liver Patient					
LCA	8.80E+01	1.70E-01	83.200	82.040	0.100
CHIO	8.75E+01	1.40E-00	81.201	80.100	0.100
PSO	8.65E+00	1.60E-00	80.120	80.100	0.120
AGDE	8.10E+01	1.65E-00	80.000	79.100	0.140
HHO	7.80E+00	1.45E-00	79.121	78.100	0.100
INFO	7.85E+01	1.40E-00	79.000	77.010	0.120
RUN	7.80E+01	1.20E-00	78.220	77.221	0.120
IMODE	7.75E+01	1.79-E00	79.120	77.210	0.130

Table 9
Wilcoxon rank-sum test for MOA and QSAR.

LCA vs.	QSAR	MAO
	p-value	p-value
IMODE	2.67E-11	3.09E-13
AGDE	3.18E-11	7.22E-06
INFO	2.58E-11	2.78E-12
RUN	2.45E-11	5.31E-06
CHIO	2.48E-11	1.52E-11
PSO	2.49E-11	1.66E-06
HHO	8.97E-08	7.15E-13

Table 10
Wilcoxon rank-sum test for drug review and drug consumption.

LCA vs.	Drug review	Drug consumption
	p-value	p-value
IMODE	1.07E-11	3.05E-11
AGDE	2.10E-11	7.02E-05
INFO	2.50E-10	1.70E-09
RUN	1.40E-05	3.21E-07
CHIO	1.40E-09	1.30E-09
PSO	3.09E-10	1.60E-04
HHO	6.97E-09	5.05E-11

Table 11

Wilcoxon rank-sum test for QSAR androgen receptor and Immunotherapy.

LCA vs.	QSAR androgen receptor	Immunotherapy
	p-value	p-value
IMODE	2.05E-11	3.01E-13
AGDE	3.22E-11	7.01E-06
INFO	1.50E-11	2.01E-12
RUN	1.20E-11	5.22E-06
CHIO	1.50E-11	1.30E-11
PSO	2.30E-11	1.66E-06
HHO	6.97E-08	5.15E-11

Table 12

Wilcoxon rank-sum test for HCV Egypt and Gene Expression Cancer RNA-Seq.

LCA vs.	HCV Egypt	Gene Expression Cancer RNA-Seq
	p-value	p-value
IMODE	2.10E-11	2.01E-12
AGDE	3.20E-10	7.20E-05
INFO	1.40E-10	2.11E-10
RUN	1.10E-10	5.20E-05
CHIO	1.40E-10	1.20E-10
PSO	2.30E-10	1.60E-05
HHO	4.90E-07	5.15E-11

Table 13

Wilcoxon rank-sum test for Primary Tumor and Alzheimer Features.

LCA vs.	Primary Tumor	Alzheimer Features
	p-value	p-value
IMODE	2.05E-11	2.00E-11
AGDE	3.10E-10	7.10E-05
INFO	1.30E-10	2.10E-10
RUN	1.07E-10	5.10E-05
CHIO	1.20E-10	1.10E-10
PSO	1.20E-10	1.40E-05
HHO	3.70E-07	5.10E-11

Table 14

Wilcoxon rank-sum test for HCC Survival Data and Indian Liver Patient.

LCA vs.	HCC Survival Data	Indian Liver Patient
	p-value	p-value
IMODE	1.05E-11	1.00E-11
AGDE	2.10E-05	5.10E-05
INFO	1.20E-05	2.02E-04
RUN	1.04E-05	5.03E-05
CHIO	1.05E-05	1.00E-05
PSO	1.10E-05	1.10E-03
HHO	3.50E-05	5.05E-11

where L represents the number of features in the initial dataset, Q is the highest score obtained at each iteration, and M is the total number of iterations.

Tables 15, 16, 17, 18, 19, and 20 present the results of the ASS test for the LCA algorithm. In particular, the LCA algorithm achieved the lowest number of selected features in all datasets, indicating its ability to identify a concise and informative subset of features for effective classification.

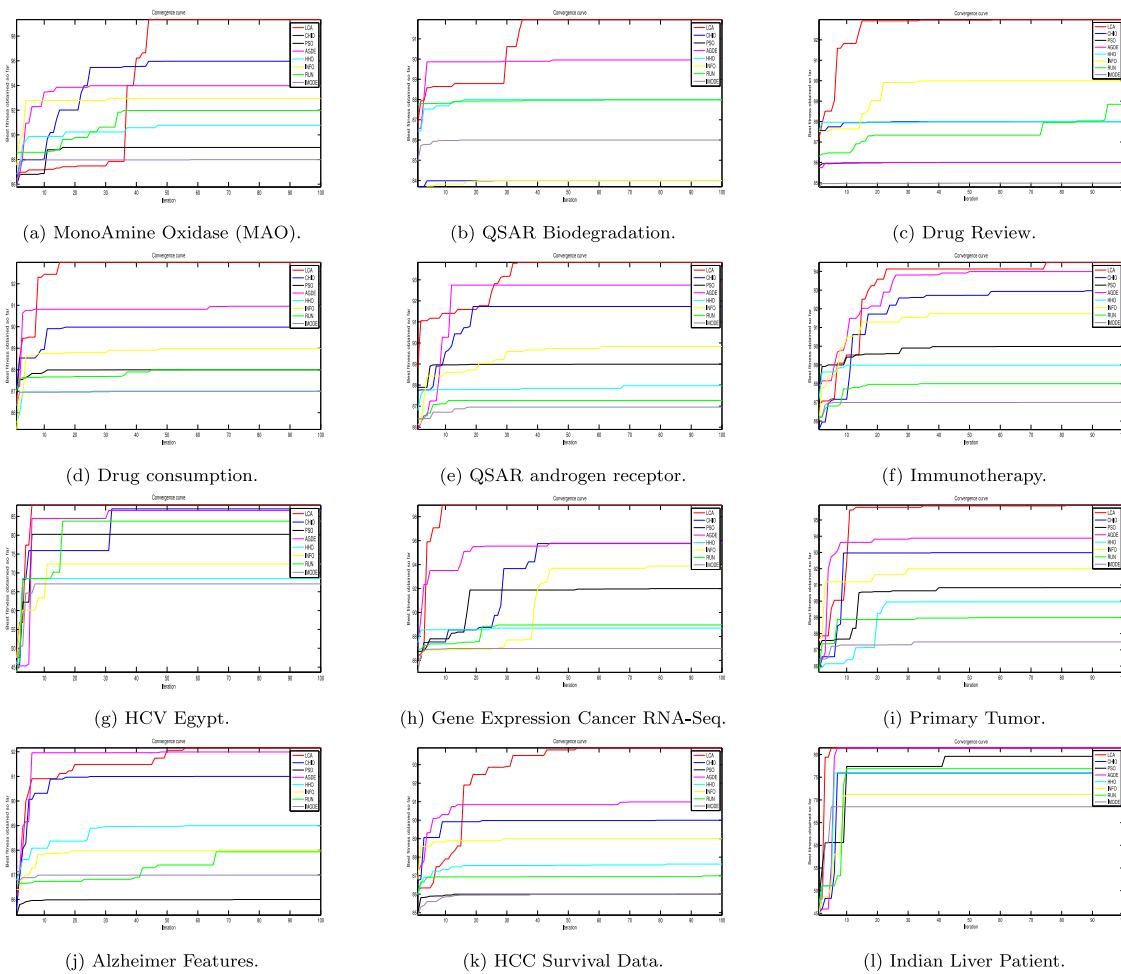


Fig. 11. The convergence graphs acquired from the LCA algorithm and the compared algorithms across twelve datasets.

Table 15
Average selection size of features (ASS) for MOA and QSAR.

	QSAR	MOA
	ASS	ASS
IMODE	0.1134	0.3334
AGDE	0.4401	0.6614
INFO	0.1571	0.7704
RUN	0.4109	0.4073
CHIO	0.3028	0.4109
PSO	0.4617	0.4666
HHO	0.4999	0.4733
LCA	0.1108	0.2019

Table 16
Average selection size of features (ASS) for Drug Review and Drug consumption.

	Drug Review	Drug consumption
	ASS	ASS
IMODE	0.2070	0.2088
AGDE	0.2089	0.3102
INFO	0.2086	0.2902
RUN	0.2082	0.3102
CHIO	0.4666	0.4011
PSO	0.4617	0.5600
HHO	0.4999	0.4733
LCA	0.1793	0.2009

Table 17
Average selection size of features (ASS) for QSAR androgen receptor and Immunotherapy.

	QSAR androgen receptor	Immunotherapy
	ASS	ASS
IMODE	0.2070	0.2088
AGDE	0.2089	0.3102
INFO	0.2086	0.2902
RUN	0.2082	0.3102
CHIO	0.4666	0.4011
PSO	0.4617	0.5600
HHO	0.4999	0.4733
LCA	0.1793	0.2009

5.2.2. Convergence behavior analysis

Fig. 11 compares the convergence graphs of the LCA algorithm and competitors for twelve datasets. Stable positions of all datasets served as evidence that the LCA algorithm converges successfully. Additionally, the LCA algorithm outperformed other algorithms in achieving the most significant mean of the best solutions and the most rapid convergence for most datasets. These results suggest that the LCA algorithm is a good optimization approach for addressing FS issues and performing high performance.

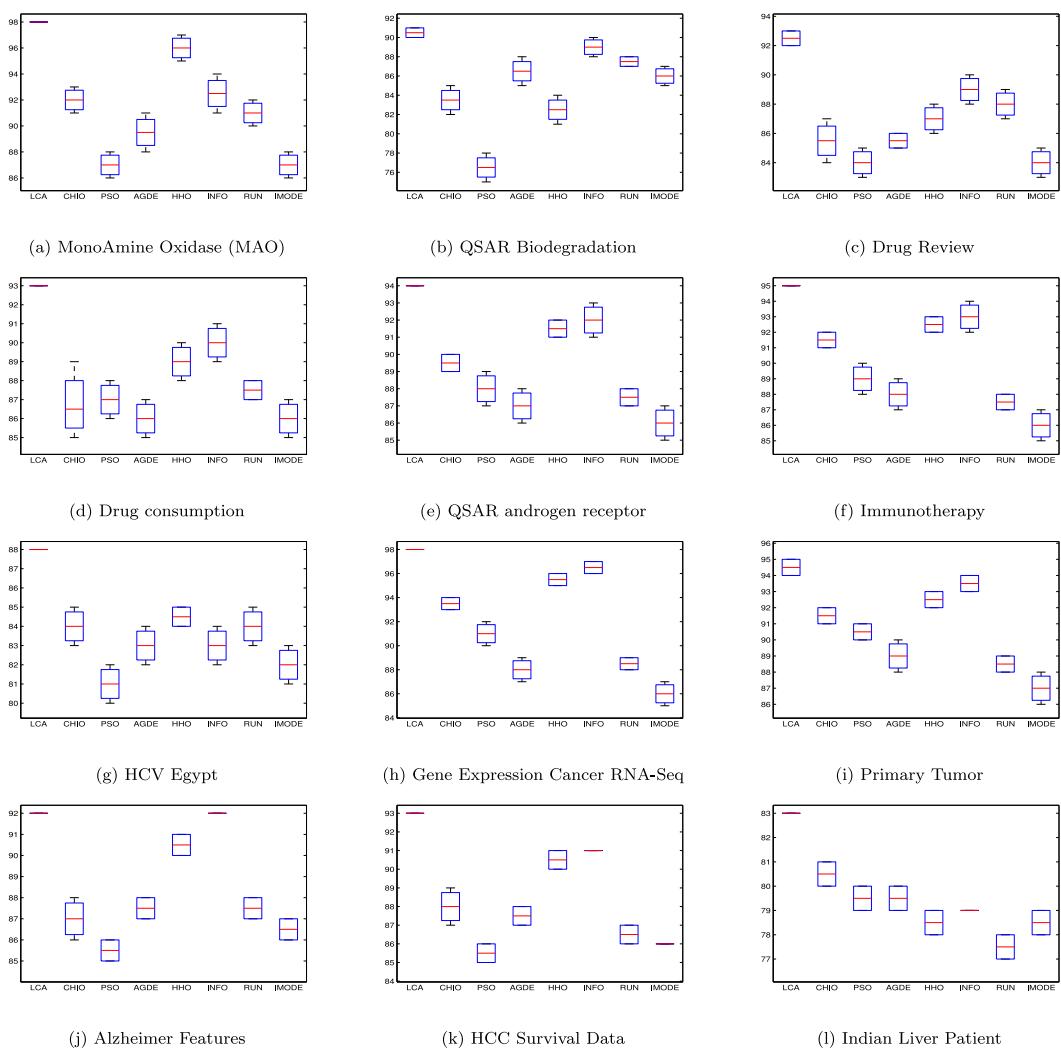


Fig. 12. The boxplot comparison obtained from the algorithms compared that employ SVM in twelve datasets.

Table 18
Average selection size of features (ASS) for HCV Egypt and Gene Expression Cancer RNA-Seq.

	HCV Egypt	Gene Expression Cancer RNA-Seq
ASS	ASS	ASS
IMODE	0.3020	0.2134
AGDE	0.2921	0.2014
INFO	0.3652	0.2635
RUN	0.4236	0.4556
CHIO	0.4218	0.4918
PSO	0.5900	0.5000
HHO	0.3223	0.3393
LCA	0.1104	0.1004

Table 19
Average selection size of features (ASS) for Primary Tumor and Alzheimer Features.

	Primary Tumor	Alzheimer Features
	ASS	ASS
IMODE	0.2012	0.2003
AGDE	0.2114	0.2481
INFO	0.3621	0.3429
RUN	0.5000	0.3333
CHIO	0.2481	0.3379
PSO	0.5673	0.5100
HHO	0.2367	0.2433
LCA	0.1393	0.1000

5.2.3. Boxplot analysis

The non-parametric method of boxplot is employed to assess the performance of various datasets. A boxplot is a descriptive statistics tool to represent several numerical data using their quartiles. It may include lines that extend vertically from the boxes to indicate variability outside the upper and lower quartiles, and individual points can be plotted as outliers. The highest or lowest data points obtained are the algorithm's maximum or minima. The dispersion and skewness of the data and the contours of the data are represented by the distances between the various areas of the box. Boxplots for the LCA-SVM method on the

twelve datasets are displayed in Fig. 12. In most cases, the boxplots of the suggested LCA algorithm are narrower than the distributions of other algorithms and have the largest values.

6. Conclusion and future work

This paper proposes a new bio-inspired optimization algorithm named the Liver Cancer Algorithm (LCA) as a promising approach to solve optimization problems. The inspiration for the LCA algorithm is derived from the behavior of liver tumors in the human body. Liver cancer, specifically hepatocellular carcinoma, is a severe and

Table 20
Average selection size of features (ASS) for HCC Survival Data and Indian Liver Patient.

	HCC Survival Data	Indian Liver Patient
ASS	ASS	ASS
IMODE	0.1429	0.1408
AGDE	0.2229	0.3368
INFO	0.7428	0.7554
RUN	0.5041	0.3017
CHIO	0.5030	0.3081
PSO	0.2881	0.2680
HHO	0.1229	0.1429
LCA	0.0039	0.1018

complex disease that poses significant challenges for diagnosis and treatment. The growth and spread of liver tumors depend on various factors, including the location, size, and aggressiveness of the tumor. The behavior of liver tumors can be modeled in terms of optimization concepts. In the context of optimization, the LCA algorithm mimics the process of spreading and growth of liver tumors in the liver. The algorithm design is inspired by the adaptive nature of the tumor, where it tries to find the most favorable environment for growth within the liver. It uses an evolutionary search approach that simulates the behavior of liver tumors when they take over the liver organ. The ability of the tumor to replicate and spread to other organs inspires the algorithm. The design of the LCA algorithm incorporates genetic operators and a Random Opposition-Based Learning (ROBL) strategy. These elements efficiently balance local and global searches, allowing for effective exploration of the search space. The fitness function of the candidate solutions allows the LCA algorithm to traverse multiple levels of liver tumors. If a candidate solution yields poor fitness, the algorithm allows another tumor to modify the position of the fittest tumor size. The results demonstrated that the LCA algorithm strikes a suitable balance between global exploration and local exploitation, leading to effective solutions to optimization problems. Against the compared algorithms, the LCA algorithm showed a competitive advantage in various optimization applications. The evaluation of the LCA algorithm involved a qualitative analysis using multiple metrics, including search history, trajectory, average fitness of the solutions, and the best solution in each iteration. This comprehensive assessment provided valuable information on the efficiency and effectiveness of the algorithm. Furthermore, we leveraged the LCA's capabilities to address the challenging feature selection problem. We developed new wrapper FS algorithms based on the LCA's structure, enabling selection of relevant features. Additionally, we integrated the LCA algorithm with the SVM classifier, facilitating accurate and rapid classification rates. We utilized medical datasets to evaluate the performance of the proposed LCA algorithm variants, showcasing the algorithm's proficiency in practical optimization problems. The successful implementation of the LCA algorithm for feature selection further emphasized its high capability in real-world scenarios. These results highlight the potential of the LCA algorithm as a powerful tool in optimization and its applicability in various domains, including medical data analysis.

In terms of future research, the LCA algorithm presents several promising avenues of exploration. First, the LCA algorithm can be extended to address large-scale problems across different domains, including multiobjective optimization and image processing. Furthermore, the application of the LCA algorithm in drug development holds significant potential. Furthermore, exploring the hybridization of the LCA algorithm with other metaheuristic algorithms offers an exciting direction for future studies. Specifically, the authors propose investigating the binary version of LCA, which would expand its capabilities. In addition, the LCA algorithm can be applied to tackle a wide range of challenging real-world problems and diverse applications, such as real-time systems, image segmentation problems, and improving deep learning architectures. Notably, developing a multi-objective variant

of the LCA algorithm to handle NP-hard problems like the Traveling Salesman Problem is worth considering. Lastly, future focus should also include the application of the LCA algorithm to complex engineering problems and exploring the hybridization of its hunting strategies with other evolutionary methods. These research directions advance the effectiveness, versatility, and applicability of LCA in various domains.

Funding

Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2023R206), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

CRediT authorship contribution statement

Essam H. Houssein: Supervising, Software, Methodology, Conceptualization, Formal analysis, Investigation, Visualization, Writing – review & editing. **Diego Oliva:** Methodology, Conceptualization, Formal analysis, Investigation, Visualization, Writing – review & editing. **Nagwan Abdel Samee:** Funding acquisition, Resources, Data curation, Validation, Writing – review & editing. **Noha F. Mahmoud:** Resources, Data curation, Validation, Writing – review & editing. **Marwa M. Emam:** Methodology, Conceptualization, Formal analysis, Investigation, Visualization, Writing – review & editing.

Declaration of competing interest

No conflict of interest exists. We confirm that the manuscript has been read and approved by all named authors.

Data availability

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Acknowledgments

The authors express their gratitude to Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2023R206), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

References

- [1] G. Dhiman, Ssc: A hybrid nature-inspired meta-heuristic optimization algorithm for engineering applications, *Knowl.-Based Syst.* 222 (2021) 106926.
- [2] X. Chen, C. Mei, B. Xu, K. Yu, X. Huang, Quadratic interpolation based teaching-learning-based optimization for chemical dynamic system optimization, *Knowl.-Based Syst.* 145 (2018) 250–263.
- [3] E.H. Houssein, D. Oliva, E. Çelik, M.M. Emam, R.M. Ghoniem, Boosted sooty tern optimization algorithm for global optimization and feature selection, *Expert Syst. Appl.* 213 (2023) 119015.
- [4] M.M. Emam, H.A. El-Sattar, E.H. Houssein, S. Kamel, Modified orca predation algorithm: Developments and perspectives on global optimization and hybrid energy systems, *Neural Comput. Appl.* (2023) 1–23.
- [5] E.H. Houssein, E. Çelik, M.A. Mahdy, R.M. Ghoniem, Self-adaptive equilibrium optimizer for solving global, combinatorial, engineering, and multi-objective problems, *Expert Syst. Appl.* 195 (2022) 116552.
- [6] N. Singh, E.H. Houssein, S. Singh, G. Dhiman, Hssahho: A novel hybrid salp swarm-Harris Hawks optimization algorithm for complex engineering problems, *J. Ambient. Intell. Humaniz. Comput.* (2022) 1–37.
- [7] E.H. Houssein, M.M. Emam, A.A. Ali, Improved manta ray foraging optimization for multi-level thresholding using covid-19 ct images, *Neural Comput. Appl.* 33 (24) (2021) 16899–16919.
- [8] M.M. Emam, E.H. Houssein, R.M. Ghoniem, A modified reptile search algorithm for global optimization and image segmentation: Case study brain mri images, *Comput. Biol. Med.* 152 (2023) 106404.
- [9] M.M. Emam, N.A. Samee, M.M. Jamjoom, E.H. Houssein, Optimized deep learning architecture for brain tumor classification using improved hunger games search algorithm, *Comput. Biol. Med.* (2023) 106966.

- [10] E.H. Houssein, D. Oliva, E. Çelik, M.M. Emam, R.M. Ghoniem, Boosted sooty tern optimization algorithm for global optimization and feature selection, *Expert Syst. Appl.* 213 (2023) 119015.
- [11] M. Mafarja, T. Thaher, J. Too, H. Chantar, H. Turabieh, E.H. Houssein, M.M. Emam, An efficient high-dimensional feature selection approach driven by enhanced multi-strategy grey wolf optimizer for biological data classification, *Neural Comput. Appl.* (2022) 1–27.
- [12] O. Olorunda, A.P. Engelbrecht, Measuring exploration/exploitation in particle swarms using swarm diversity, in: 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), IEEE, 2008, pp. 1128–1134.
- [13] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, *IEEE Trans. Evol. Comput.* 1 (1) (1997) 67–82.
- [14] E.H. Houssein, E. Çelik, M.A. Mahdy, R.M. Ghoniem, Self-adaptive equilibrium optimizer for solving global, combinatorial, engineering, and multi-objective problems, *Expert Syst. Appl.* 195 (2022) 116552.
- [15] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Sixth International Symposium on Micro Machine and Human Science, IEEE, 1995, pp. 39–43.
- [16] D. Karaboga, Artificial bee colony algorithm, *Scholarpedia* 5 (3) (2010) 6915.
- [17] M. Dorigo, T. Stützle, *Ant Colony Optimization: Overview and Recent Advances*, Springer, 2019.
- [18] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, *Adv. Eng. Software* 69 (2014) 46–61.
- [19] S. Mirjalili, A. Lewis, The whale optimization algorithm, *Adv. Eng. Software* 95 (2016) 51–67.
- [20] A. Faramarzi, M. Heidarinejad, S. Mirjalili, A.H. Gandomi, Marine predators algorithm: A nature-inspired metaheuristic, *Expert Syst. Appl.* (2020) 113377.
- [21] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarrah, M. Mafarja, H. Chen, Harris Hawks optimization: Algorithm and applications, *Future Gener. Comput. Syst.* 97 (2019) 849–872.
- [22] S. Li, H. Chen, M. Wang, A.A. Heidari, S. Mirjalili, Slime mould algorithm: A new method for stochastic optimization, *Future Gener. Comput. Syst.* 111 (2020) 300–323.
- [23] S. Kaur, L.K. Awasthi, A. Sangal, G. Dhiman, Tunicate swarm algorithm: A new bio-inspired based metaheuristic paradigm for global optimization, *Eng. Appl. Artif. Intell.* 90 (2020) 103541.
- [24] M. Braik, A. Hammouri, J. Atwan, M.A. Al-Betar, M.A. Awadallah, White shark optimizer: A novel bio-inspired meta-heuristic algorithm for global optimization problems, *Knowl.-Based Syst.* 243 (2022) 108457.
- [25] F. MiarNaeimi, G. Azizyan, M. Rashki, Horse herd optimization algorithm: A nature-inspired algorithm for high-dimensional optimization problems, *Knowl.-Based Syst.* 213 (2021) 106711.
- [26] L. Abualigah, M. Abd Elaziz, P. Sumari, Z.W. Geem, A.H. Gandomi, Reptile search algorithm (rsa): A nature-inspired meta-heuristic optimizer, *Expert Syst. Appl.* 191 (2022) 116158.
- [27] Y. Yang, H. Chen, A.A. Heidari, A.H. Gandomi, Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts, *Expert Syst. Appl.* 177 (2021) 114864.
- [28] J. Tu, H. Chen, M. Wang, A.H. Gandomi, The colony predation algorithm, *J. Bionic Eng.* 18 (2021) 674–710.
- [29] S. Suyanto, A.A. Ariyanto, A.F. Ariyanto, Komodo mlipir algorithm, *Appl. Soft Comput.* 114 (2022) 108043.
- [30] I. Ahmadianfar, A.A. Heidari, A.H. Gandomi, X. Chu, H. Chen, Run beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method, *Expert Syst. Appl.* 181 (2021) 115079.
- [31] N. Chopra, M.M. Ansari, Golden jackal optimization: A novel nature-inspired optimizer for engineering applications, *Expert Syst. Appl.* 198 (2022) 116924.
- [32] H.T. Sadeeq, A.M. Abdulazeez, Giant trevally optimizer (gto): A novel metaheuristic algorithm for global optimization and challenging engineering problems, *IEEE Access* 10 (2022) 121615–121640.
- [33] W. Zhao, L. Wang, S. Mirjalili, Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications, *Comput. Methods Appl. Mech. Engrg.* 388 (2022) 114194.
- [34] J.H. Holland, Genetic algorithms, *Sci. Am.* 267 (1) (1992) 66–73.
- [35] R. Storn, K. Price, Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces, *J. Glob. Optim.* 11 (4) (1997) 341–359.
- [36] X. Yao, A new simulated annealing algorithm, *Int. J. Comput. Math.* 56 (3–4) (1995) 161–168.
- [37] E. Raschedi, H. Nezamabadi-Pour, S. Saryazdi, Gsa: A gravitational search algorithm, *Inf. Sci.* 179 (13) (2009) 2232–2248.
- [38] M. Dehghani, Z. Montazeri, A. Dehghani, A. Seifi, Spring search algorithm: A new meta-heuristic optimization algorithm inspired by Hooke's law, in: 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation, KBEI, IEEE, 2017, pp. 0210–0214.
- [39] S. Mirjalili, S.M. Mirjalili, A. Hatamlou, Multi-verse optimizer: A nature-inspired algorithm for global optimization, *Neural Comput. Appl.* 27 (2016) 495–513.
- [40] A. Faramarzi, M. Heidarinejad, B. Stephens, S. Mirjalili, Equilibrium optimizer: A novel optimization algorithm, *Knowl.-Based Syst.* 191 (2020) 105190.
- [41] F.A. Hashim, K. Hussain, E.H. Houssein, M.S. Mabrouk, W. Al-Atabany, Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems, *Appl. Intell.* 51 (2021) 1531–1551.
- [42] F.A. Hashim, E.H. Houssein, M.S. Mabrouk, W. Al-Atabany, S. Mirjalili, Henry gas solubility optimization: A novel physics-based algorithm, *Future Gener. Comput. Syst.* 101 (2019) 646–667.
- [43] I. Ahmadianfar, A.A. Heidari, S. Noshadian, H. Chen, A.H. Gandomi, Info: An efficient optimization algorithm based on weighted mean of vectors, *Expert Syst. Appl.* 195 (2022) 116516.
- [44] R. Moghdani, K. Salimifard, Volleyball premier league algorithm, *Appl. Soft Comput.* 64 (2018) 161–185.
- [45] M. Dehghani, M. Mardaneh, J.M. Guerrero, O. Malik, V. Kumar, Football game based optimization: An application to solve energy commitment problem, *Int. J. Intell. Eng. Syst.* 13 (5) (2020) 514–523.
- [46] M. Dehghani, Z. Montazeri, H. Givi, J.M. Guerrero, G. Dhiman, Darts game optimizer: A new optimization technique based on darts game, *Int. J. Intell. Eng. Syst.* 13 (5) (2020) 286–294.
- [47] F.A. Zeidabadi, M. Dehghani, Poa: Puzzle optimization algorithm, *Int. J. Intell. Eng. Syst.* 15 (1).
- [48] R.V. Rao, V.J. Savsani, D. Vakharia, Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems, *Comput.-Aided Des.* 43 (3) (2011) 303–315.
- [49] M. Dehghani, M. Mardaneh, O. Malik, Foa: Following optimization algorithm for solving power engineering optimization problems, *J. Oper. Automat. Power Eng.* 8 (1) (2020) 57–64.
- [50] M. Dehghani, E. Trojovská, P. Trojovský, A new human-based metaheuristic algorithm for solving optimization problems on the base of simulation of driving training process, *Sci. Rep.* 12 (1) (2022) 9924.
- [51] S.H.S. Moosavi, V.K. Bardsiri, Poor and rich optimization algorithm: A new human-based and multi populations algorithm, *Eng. Appl. Artif. Intell.* 86 (2019) 165–181.
- [52] C. Wang, X. Zhang, Y. Niu, S. Gao, J. Jiang, Z. Zhang, P. Yu, H. Dong, Dual-population social group optimization algorithm based on human social group behavior law, *IEEE Trans. Comput. Soc. Syst.* 10 (1) (2022) 166–177.
- [53] S.J. Mousavirad, H. Ebrahimpour-Komleh, Human mental search: A new population-based metaheuristic optimization algorithm, *Appl. Intell.* 47 (2017) 850–887.
- [54] S. Thawkar, S. Sharma, M. Khanna, L. kumar Singh, Breast cancer prediction using a hybrid method based on butterfly optimization algorithm and ant lion optimizer, *Comput. Biol. Med.* 139 (2021) 104968.
- [55] G.I. Sayed, M.M. Soliman, A.E. Hassani, A novel melanoma prediction model for imbalanced data using optimized squeezeenet by bald eagle search optimization, *Comput. Biol. Med.* 136 (2021) 104712.
- [56] J. Xing, H. Zhao, H. Chen, R. Deng, L. Xiao, Boosting whale optimizer with quasi-oppositional learning and Gaussian barebone for feature selection and covid-19 image segmentation, *J. Bionic Eng.* 20 (2) (2023) 797–818.
- [57] J. Piri, P. Mohapatra, An analytical study of modified multi-objective Harris Hawk optimizer towards medical data feature selection, *Comput. Biol. Med.* 135 (2021) 104558.
- [58] M. Sabha, T. Thaher, M.M. Emam, Cooperative swarm intelligence algorithms for adaptive multilevel thresholding segmentation of covid-19 ct-scan images, *JUCS - J. Universal Comput. Sci.* 29 (7) (2023) 759–804, <http://dx.doi.org/10.3897/jucs.93498>.
- [59] A.M. Khalid, H.M. Hamza, S. Mirjalili, K.M. Hosny, Bcovidoa: A novel binary coronavirus disease optimization algorithm for feature selection, *Knowl.-Based Syst.* 248 (2022) 108789.
- [60] E.H. Houssein, E. Saber, A.A. Ali, Y.M. Wazery, Centroid mutation-based search and rescue optimization algorithm for feature selection and classification, *Expert Syst. Appl.* 191 (2022) 116235.
- [61] M. Abd Elaziz, D. Yousri, Automatic selection of heavy-tailed distributions-based synergy henry gas solubility and Harris Hawk optimizer for feature selection: case study drug design and discovery, *Artif. Intell. Rev.* (2021) 1–46.
- [62] M. Abd Elaziz, Y.S. Moemen, A.E. Hassani, S. Xiong, Toxicity risks evaluation of unknown fda biotransformed drugs based on a multi-objective feature selection approach, *Appl. Soft Comput.* 97 (2020) 105509.
- [63] E.H. Houssein, N. Neggaz, M.E. Hosney, W.M. Mohamed, M. Hassaballah, Enhanced Harris Hawks optimization with genetic operators for selection chemical descriptors and compounds activities, *Neural Comput. Appl.* 33 (20) (2021) 13601–13618.
- [64] E.H. Houssein, M.E. Hosney, M. Elhoseny, D. Oliva, W.M. Mohamed, M. Hassaballah, Hybrid Harris Hawks optimization with cuckoo search for drug design and discovery in chemoinformatics, *Sci. Rep.* 10 (1) (2020) 1–22.
- [65] J.P. Feldman, R. Goldwasser, S. Mark, J. Schwartz, I. Orion, A mathematical model for tumor volume evaluation using two-dimensions, *J. Appl. Quant. Methods* 4 (4) (2009) 455–462.
- [66] J. Sápi, L. Kovács, D.A. Drexler, P. Kocsis, D. Gajári, Z. Sápi, Tumor volume estimation and quasi-continuous administration for most effective bevacizumab therapy, *PLoS One* 10 (11) (2015) e0142190.
- [67] A. Talkington, R. Durrett, Estimating tumor growth rates in vivo, *Bull. Math. Biol.* 77 (10) (2015) 1934–1954.

- [68] E. Emary, H.M. Zawbaa, M. Sharawi, Impact of Lèvy flight on modern meta-heuristic optimizers, *Appl. Soft Comput.* 75 (2019) 775–789.
- [69] E.H. Houssein, M.M. Emam, A.A. Ali, An efficient multilevel thresholding segmentation method for thermography breast cancer imaging based on improved chimp optimization algorithm, *Expert Syst. Appl.* 185 (2021) 115651.
- [70] S. Vatandoust, T.J. Price, C.S. Karapetis, Colorectal cancer: Metastases to a single organ, *World J. Gastroenterol.* 21 (41) (2015) 11767.
- [71] A.W. Mohamed, A.K. Mohamed, Adaptive guided differential evolution algorithm with novel mutation for numerical optimization, *Int. J. Mach. Learn. Cybern.* 10 (2) (2019) 253–277.
- [72] K.M. Sallam, S.M. Elsayed, R.K. Chakrabortty, M.J. Ryan, Improved multi-operator differential evolution algorithm for solving unconstrained problems, in: 2020 IEEE Congress on Evolutionary Computation, CEC, IEEE, 2020, pp. 1–8.
- [73] M.A. Al-Betar, Z.A.A. Alyasseri, M.A. Awadallah, I. Abu Doush, Coronavirus herd immunity optimizer (chio), *Neural Comput. Appl.* 33 (10) (2021) 5011–5042.
- [74] A. Arcuri, G. Fraser, Parameter tuning or default values? An empirical investigation in search-based software engineering, *Empir. Softw. Eng.* 18 (3) (2013) 594–623.
- [75] E.H. Houssein, M.M. Emam, A.A. Ali, An optimized deep learning architecture for breast cancer diagnosis based on improved marine predators algorithm, *Neural Comput. Appl.* (2022) 1–19.
- [76] J. Cai, J. Luo, S. Wang, S. Yang, Feature selection in machine learning: A new perspective, *Neurocomputing* 300 (2018) 70–79.
- [77] E.H. Houssein, M.M. Emam, A.A. Ali, P.N. Suganthan, Deep and machine learning techniques for medical imaging-based breast cancer: A comprehensive review, *Expert Syst. Appl.* 167 (2021) 114161.
- [78] D. Devikanniga, A. Ramu, A. Haldorai, Efficient diagnosis of liver disease using support vector machine optimized with crows search algorithm, *EAI Endorsed Trans. Energy Web* 7 (29) (2020) e10.
- [79] E.H. Houssein, N. Neggaz, M.E. Hosney, W.M. Mohamed, M. Hassaballah, Enhanced Harris Hawks optimization with genetic operators for selection chemical descriptors and compounds activities, *Neural Comput. Appl.* 33 (20) (2021) 13601–13618.
- [80] A.G. Hussien, A.E. Hassanien, E.H. Houssein, Swarming behaviour of salps algorithm for predicting chemical compound activities, in: 2017 Eighth International Conference on Intelligent Computing and Information Systems, ICICIS, IEEE, 2017, pp. 315–320.
- [81] G. Chandrashekhar, F. Sahin, A survey on feature selection methods, *Comput. Electr. Eng.* 40 (1) (2014) 16–28.
- [82] A. Abdiansah, R. Wardoyo, Time complexity analysis of support vector machines (svm) in libsvm, *Int. J. Comput. Appl.* 128 (3) (2015) 28–34.