

A hybrid BFO-GA optimization framework for real-time power management in smart energy systems

Kuldeep Vaydande^{1,*},
Rahul Mirajkar², Amolkumar Jadhav³,
Viomesh Kumar Singh¹,
Praveenkumar Patel², Preeti Bailke¹,
Sonali Bhoite¹, Umar Mulani⁴ and
Mahavir A. Devmane⁵

¹Department of Information Technology, Vishwakarma Institute of Technology, Pune, India

²Department of Computer Science and Engineering, Bharati Vidyapeeth's College of Engineering, Kolhapur, India

³Department of Computer Science and Engineering, D.Y.Patil College of Engineering and Technology, Kolhapur, India

⁴Department of Computer Science and Engineering, MIT Art, Design and Technology University, Pune, India

⁵Department of Computer Science and Engineering(AI & ML) , VPPCOE & VA, Mumbai, India

*E-mail: kuldeep.vayadande1@vit.edu

Received for publication
August 23, 2025.

Abstract

Current energy management faces its most significant power optimization struggle because rising demand meets with environmental pressures and requires effective resource utilization. The research develops an advanced implementation of bitterling fish optimization (BFO) combined with the genetic algorithm (GA) to perform power consumption optimization in intricate systems. The proposed framework connects BFO swarm intelligence to GA evolutionary strategies through machine learning adaptation, which simultaneously minimizes power losses and improves system efficiency throughout multiple applications. The hybrid approach produces quicker convergence through improved optimization performance and delivers flexibility toward dynamic energy requirements. The model requires consideration of three vital elements: load distribution management and renewable integration, and power loss reduction, along with interactive visual controls to monitor power distribution and optimization progress, and system output. Additionally, cloud and edge computing technologies enable large-scale data processing and ensure low-latency optimization for real-time applications. The framework also supports adaptive learning mechanisms, allowing it to continuously refine optimization strategies based on real-time energy data and consumption patterns. The proposed power optimization framework has wide-ranging applications in smart grids, industrial automation, and renewable energy systems, offering a scalable, high-accuracy solution for modern energy management. The integration of predictive analytics and real-time monitoring empowers decision-makers with proactive energy-saving strategies, ultimately contributing to a sustainable, efficient, and intelligent power infrastructure. This research provides a novel, data-driven approach to power optimization, addressing critical challenges in energy management while enhancing reliability and operational efficiency in future energy infrastructures.

Keywords

power optimization, bitterling fish optimization, genetic algorithm, load balancing, energy efficiency, renewable energy, smart grids, edge-computing, predictive analytics

I. Introduction

Optimization plays a crucial role in various fields, particularly in power systems, where efficiency and sustainability are paramount. Traditional optimization methods often struggle to address complex, high-dimensional problems, necessitating the use of advanced computational techniques. Metaheuristic algorithms, inspired by natural and biological processes, have emerged as powerful tools for solving such challenges. Among these, bio-inspired algorithms have gained significant attention due to their adaptability and effectiveness in exploring vast solution spaces.

The bitterling fish optimization (BFO) algorithm uses the reproductive and survival methods of bitterling fish (*Rhodeus sericeus*) to develop its bio-inspired optimization model. Bitterling fish need to select proper reproduction grounds among their limited freshwater space and develop predator protection strategies to guarantee the survival of their young. The mathematical processing model of BFO creates various solution alternatives that work effectively with power system applications. Natural selection together with evolution drives genetic algorithm (GA) to develop into a powerful system for power optimization.

A defined evolutionary approach allows GA to provide both high operational efficiency and powerful outcomes for the resolution of complex optimization tasks. Relational optimization occurs through BFO and GA hybrid modeling, where each approach strengthens the resulting model for managing contemporary power systems.

a. Biological inspiration

The conceptual framework of BFO uses its approach on survival and reproduction methods implemented by *Rhodeus sericeus* bitterling fish. The mating process of female bitterling fish employs freshwater mussels as reproductive hosts to reproduce patterns that match the natural fish-mussel ecological interactions.

b. Background and importance of power optimization

Power systems of modernity demand energy optimization for achieving maximum efficiency, combined with sustainability alongside reduced costs. The rising need for energy in industrial and residential sectors has made power optimization a leading topic in research. These optimization methods face difficulties

when working with difficult-to-solve high-dimensional problems. So, researchers develop nature-inspired metaheuristic algorithms as a solution.

c. Introduction to metaheuristic algorithms

Metaheuristic algorithms serve as advanced optimization tools, which prove invaluable for resolving complex problems across engineering fields besides economics and artificial intelligence domains. Because these algorithms draw their inspiration from natural phenomena along with biological evolution along with swarm intelligence and physical processes, they become very effective at navigating through extensive search areas. Metaheuristic algorithms achieve robust solutions because they excel at finding equilibrium points between exploration of new solutions and exploiting promising candidates. This makes them superior to traditional optimization techniques, whose weaknesses include local optima difficulties and inefficient computing resources. Metaheuristic algorithms operate as either single-solution-based or population-based entities in general classification terms. The single candidate optimization techniques simulated annealing and tabu search, work through successive improvements on their sole prospective solution. The population-based algorithms composed of GA, particle swarm optimization (PSO), ant colony optimization (ACO), and BFO function from diverse potential solutions while pursuing optimal results. One of the primary advantages of metaheuristic algorithms is their adaptability to various problem domains. They do not require gradient information and can handle discrete, continuous, and combinatorial optimization problems effectively. Their stochastic nature allows them to escape local optima, ensuring a higher probability of finding globally optimal solutions.

d. BFO algorithm

The BFO algorithm behaves as a new bio-inspired metaheuristic technique that uses bitterling fish reproductive and survival behaviors to operate. The Bitterling fish demonstrate exceptional intelligence because they practice smart decision-making during spawning and predator evasion, and the creation of successful offspring. Bitterling fish natural behavior serves as an optimization approach that directs candidate solutions (individual fish) to explore the solution area based on fitness evaluation.

BFO succeeds through three main operational mechanisms, which include exploration and

exploitation together with adaptation. At the beginning of its execution, the algorithm generates an initial set of candidate solutions followed by fitness evaluation before implementing iterative movements that use fish-inspired techniques. These strategies include:

The Fish Exploration Period shows fish being directionless as they search for better spawning areas, which matches how optimization explores various solution areas.

During exploitation, fish that discover ideal locations affect the behavior of other fish so search becomes focused on rewarding areas.

The population actively adjusts, as environmental changes occur to maintain robustness when optimizing dynamic problems.

BFO shows reliable performance when dealing with hard optimization problems through its adaptive search capabilities, which lead to superior results. The algorithm achieves successful implementation in power system optimization and machine learning and industrial process control.

e. GA in power optimization

The GA stands as a common evolutionary algorithm because it uses natural selection principles along with genetic evolution concepts. Researchers first identified GA as a method under the supervision of John Holland during the 1970s, and it continues to solve optimization problems throughout different fields extensively today. The algorithm follows several basic operational steps, which include:

Initialization: Randomly initiated procedures create the first set of candidate solutions, which form the population.

Fitness Evaluation: Every chromosome receives its value from an objective function evaluation process.

Selection: The process of selection identifies superior performing chromosomes through roulette wheel selection and other alternatives such as tournament selection and rank selection.

Crossover: The selected chromosomes reproduce through recombination in order to generate new offspring that enhance genetic diversity.

Mutation: Offspring undergo rounding of tiny random values to achieve both genetic diversity maintenance and avoid premature solutions.

The algorithm operates across multiple generations before termination when either a maximum

iteration limit is reached or the convergence threshold becomes satisfied.

The effectiveness of GA lies in its ability to resolve optimization problems that present non-linearity, multiple modalities, and high dimensions. GA has achieved widespread application across energy optimization, scheduling, machine learning, and engineering design fields.

f. Hybridization of BFO and GA for enhanced power optimization

The integration of BFO and GA combines the strengths of both algorithms, leveraging GA's structured evolutionary process with BFO's adaptive exploration capabilities. This hybrid approach enhances convergence speed, solution accuracy, and robustness in complex optimization scenarios. Applications of the BFO-GA hybrid include power optimization in smart grids, renewable energy management, and industrial automation. By incorporating the intelligence of natural selection from GA and the adaptive behavior of BFO, this hybrid model outperforms traditional optimization techniques, making it a promising solution for next generation optimization challenges.

g. Applications in energy systems

It is vital that the GA and BFO are integrated, enhancing energy systems by enhancing their efficiency, sustainability, and flexibility. With the incorporation of renewable energy. This model is powered by the optimum solar and wind power resources available resources, solving variability issues and importing grid Stability. It reduces fossil fuel consumption through control of power supply, which produces cleaner energy infrastructure.

In smart grids, the BFO-GA hybrid facilitates real-time load demand balancing and management. It is flexible fault-tolerant control systems to enhance performance and efficiency. Energy moves along the grid. By minimizing transmission losses and makes the grid more robust, the plan facilitates designing energy systems that can sustain themselves.

The industrial domain benefits a lot from this strategy of optimization as it minimizes energy wastage in production systems. The BFO-GA approach enhances the scheduling and resource utilization investing in production factories, that is, saving and improving working efficiency. It is also employed in electric vehicle (EV) charging optimization, with effective energy use with lower grid stress at peak.

Besides that, the model has a significant role to play in microgrids, when distributed energy systems must be precisely optimization for stability and efficiency maintenance. With real-time data analysis integrated, the hybrid BFO-GA improves decision-making procedures, thereby minimizing operations costs and enhancing energy security.

Overall, the combination of GA and BFO provides a strong solution to the present energy problems, making it the best, prudent use of resources, carbon footprint reduction, and promoting sustainable energy development.

The current study introduces a new hybrid algorithm (HBFO-GA) that combines the full ability of BFO to explore the global regions and the ability of the GA to exploit the local resources using an adaptive switching mechanism. The suggested framework develops a power-loss reduction model of real-time optimization subjected to practical operational limitations and is executed in Python 3.12 using the IEEE 30-bus benchmark system. A comparison to other metaheuristics algorithm such as PSO, differential evolution (DE), and Firefly Algorithm show that the proposed HBFO-GA performs better in terms of convergence performance and optimization accuracy. Also, the framework confirms that it can be used in real-time, is computationally efficient, and can be scaled to smart-grid settings that include integration of renewable energy sources.

The IEEE 30-bus benchmark power network underlies the studied system of implementing and evaluating the proposed Hybrid BFO-GA framework. This test system is a typical smart energy network that consists of a variety of interrelated components that attempt to simulate conditions in the real-world grid. The main components include:

- **Power Generation Units:** six thermal generating units that are non-linearly characterized in their cost.
- **Load Buses:** There are 21 load buses, which depict different industrial and residential power loads.
- **Transmission Lines and Transformers:** There are 41 transmission lines and 6 tap-changing transformers that enable the use of controlled voltage and power flow.
- **Renewable Energy Sources:** Simulated units of solar photovoltaic (PV) and wind power are combined to evaluate the impact of variability of renewable.

- **Control and Monitoring System:** This is based on a real time decision control layer developed in Python 3.12 that supports 24/7 monitoring of power losses, voltage profiles and loads distribution.
- **Optimization Engine:** The optimization engine software is placed on the top of the system, ensuring the total active power losses are minimized whilst keeping the voltage constant and ensuring efficient generation dispatch.

The proposed system arrangement offers an effective platform that is realistic to test the efficiency of the hybrid algorithm in dynamic, constrained, and renewable-integrated power systems.

II. Literature Review

a. The genesis and algorithmic paradigm of BFO

The foundational work by Zareian et al. (2024) formally presents the BFO algorithm to the scientific community. The socio-ecological inspirations, mainly the intriguing symbiotic method of breeding of bitterling fish with freshwater mussels, are expected to be carefully described in this paper and converted into a logical, mathematical, and algorithmic form. The authors describe how particular behaviors are abstracted as operators to be explored and exploited through a solution search space, e.g., mate selection, territorial defense, parasitic egg-laying strategies, and host mussel searching. Much of this introductory work would probably be devoted to the establishment of the efficacy of BFO. This is usually done by subjecting it to a test battery of standard benchmark optimization problems, including unimodal, multimodal, separable, and non-separable, as a test of its rate of convergence, its sensitivity to local optima, and its overall solution quality. It would also be important to make comparisons with the already existing metaheuristics in order to place BFO in the current range of optimization tools. The main contribution is therefore the creation of the new nature-inspired computational paradigm, which may provide a special set of search dynamics that may be useful in special classes of optimization problems. The future lines of research developments out of this introduction would probably be the sensitivity of the parameters, the theoretical basis of its convergence, and the investigation of variants of BFO with adaptive mechanisms or other models of behavior.

b. Domain-specific applications and hybridizations of BFO

After its conceptualization, BFO algorithm is starting to be extended and implemented in numerous demanding real-world issues, sometimes with new applications of the algorithm, or with hybrid frameworks that attempt to leverage its advantages with other computational methods.

Faisal et al., showcase one of the first major uses of BFO in the biomedical domain with their work of 2024, which is dedicated to the problem of automated recognition of optic disk in human retinal images. This is essential in the detection and management of glaucoma and other retinopathies at its initial stages, yet proves challenging due to anatomical variation, pathological changes, and quality of the images. The application of BFO in the research probably is to optimize important parameters in an image-processing pipeline, e.g., tuning threshold values to segmentation or optimal features to an identification model to detect the optic disk, or a direct search strategy for the locations of the disk boundaries. The essence of the contribution is to show the ability of BFO to explore the high-dimensional search space of the medical image analysis and potentially improve accuracy, robustness, or computational efficiency of the approach over existing optic disk localization systems. This paper sets the stage toward the wider use of BFO in other difficult biomedical signal and image processing problems where parameter fine-tuning or feature optimization is the most important issue of concern.

Balasubramanian et al. (2024) developed a complex hybrid optimization approach to improve the performance of the cognitive sensor networks in the field of wireless communication. They use the combination of BFO and the Mexican axolotl optimization algorithm to come up with a new spectrum sensing and multi-hop clustering routing protocol. Cognitive sensor networks are used in dynamic radio conditions, and it needs intelligent schemes to efficiently use its spectrum and to use data routing that is energy conscious to make the most out of the network life and to implement reliable communication. It is probable that this study capitalizes on the different search properties of both BFO and the Mexican axolotl algorithm, which are then potentially allocated to dissimilar sub-problems within the overall network optimization problem (e.g., BFO to find optimal cluster head selection and MAO (Metaheuristic Algorithm Optimization / Multi-Agent Optimization) to find optimal routing

paths). The contribution is a new bio-inspired hybrid solution, which is specific to the multi-faceted problems of resource management in such networks and may be more effective than single-algorithm solutions in resource management measures such as network throughput, latency, or energy usage.

Reddy and Vijayaraj (2024) has investigated the use of BFO to renewable energy systems and describes a hybrid algorithm to use with deep learning (DL) with BFO and secretary bird optimization (SBO). The goal behind this intricate architecture is to optimize the performance of high voltage gain interleaved boost converters that are very important elements in the system that incorporate PV and wind energy. The impossibility of controlling the power flow and conversion in such hybrid renewable systems is hard because the sources are intermittent, and the power electronic converters are not linear. The predictive control or system modeling may be done by the DL component, and tuning of hyperparameters of the DL model may be done by the hybrid BBO-SBO optimizer, which may correct the switching parameters of the converter in real-time or may solve the complex power dispatch problems. The main contribution of the paper is a complex, multi-layered optimization model, which is intended to optimize the performance and stability of integrated renewable energy systems, and demonstrates the capabilities of BFO in the complex, dynamic control problems of current power engineering.

Continuing on its utilization in electromechanical systems, Balamurugan and Sri Revathi (2025) examine the use of BFO to enhance the smoothness of torques in Brushless DC (BLDC) motors, which have an inbuilt DC–DC converter. BLDC motors also have torque ripple, which can cause unwanted vibrations, acoustic noise, and lower precision, especially in high-performance systems. Their study is probably to use BFO [2] to carefully balance the control parameters in the motor drive system or the related DC–DC converter, including proportional-integral-derivative (PID) controller gains, pulse width modulation (PWM) plans, or current shaping approaches. It aims to reduce torque variation in the operating range of the motor. The innovation of the study is the application of an innovative optimization algorithm designed for a particular problem in power electronics control, namely, to obtain high-quality operational properties (e.g., the absence of cogging torques, a strong reduction of commutation effects) in BLDC motors, which are widely used in robotics, electric cars, and industrial robotization.

c. GAs: Enduring principles and contemporary refinements

One of the earliest and most impactful of the evolutionary computation classes, GAs still remain a cornerstone of optimization, and current research efforts are directed at improving their operations, as well as adaptation to particular problems domains.

A tutorial by Whitley gives a detailed summary of the principles that GAs are based on. The book is priceless to both beginners and professionals, providing a comprehensive description of the essential elements of chromosomal representation of solutions, population start-up, fitness analysis and the canonical genetic operators of the process of selection (e.g., roulette wheel, tournament, rank-based), crossover (e.g., single-point, multi-point, uniform), and mutation. Other important issues such as parameter setting (population size, crossover/mutation rates), the constraints, and the balance between exploration and exploitation that GAs seek to maintain are likely addressed in the tutorial as well. Such a work has mainly a pedagogical contribution, guaranteeing a solid knowledge of GA mechanics, which is necessary to their successful use, critical analysis and more sophisticated or hybrid evolution methods. The foundational knowledge forms a basis of much of the work in evolutionary optimization.

Continuing on these principles, Zhu and Pan (2024) offer a piece of research dedicated to the improvement of GAs work specifically on the robot paths planning tasks in grid-like environments. The application of standard GAs to path planning may have problems, such as converging to suboptimal paths prematurely, it may not be able to efficiently navigate a complex arrangement of obstacles, or it may produce infeasible paths, which go against the constraints. Probably, the study will bring some particular changes to the GA scheme, including new encoding schemes to describe paths on a grid (e.g. sequence of moves, waypoints), special genetic operators to help maintain the path validity or exploration of a wider range of paths, and advanced fitness functions to trade off the path length, path smoothness, clearance of obstacles, and computational cost. Its main contribution is the creation of a high-quality GA variant that shows a better performance in terms of quality of the solution (e.g. shorter, safer paths) or computational efficiency compared to traditional GAs or other established and proven path planning algorithms for a similar robotic system.

d. Versatility of GAs in optimization and machine learning

The natural flexibility and resilience of GAs have seen them be used successfully in a wide range of optimization problems, including general-purpose system parameter optimization, as well as playing an inseparable role in more sophisticated machine learning processes.

This article by Hu (2024) is an example of the wide applicability of GAs in optimization of system parameters in many complex systems where more conventional analytical or gradient-based optimization methods might not be sufficient or convenient. This study probably gives us a general structure or case examples of how GAs would be effective to search through high-dimensional and non-linear parameter space to find configurations that optimize the performance, efficiency, robustness, or other desired goals in a system. The systems in question may be engineering design, control system tuning, calibration of simulation models or optimization of manufacturing processes. The addition is that it strengthens the use of GAs as a potent and general black-box optimization method, which can address issues that have noisy fitness functions or discontinuous search landscapes, thus enabling its use in improving a large variety of scientific and industrial problems.

Housing the important field of machine learning model development, the article by Houssein et al. (2024) is narrower in its focus on the feature selection problem, where the authors apply GAs. Choosing an optimal set of input features is not only important in developing parsimonious, generalizable, and interpretable machine learning models, but it is also important in minimizing the computational cost. Probably, this paper explains the fact that the combinatorial quality of feature selection (the ability to select the best subset of 2^N possibilities) fits the search functionality of GA particularly well. It would specify the standard GA encoding of this problem (i.e., a binary string and each bit represents whether or not a feature is used) and the fitness function, which typically measures the performance (e.g., accuracy, F1-score, AUC (Area Under Curve)) of a selected machine learning model trained on the subset of features selected by the GA, typically with a penalty term to favor smaller subsets.

Offering a broader picture of heuristic optimization in a critical infrastructure application, Niu et al. (2024) provide an in-depth review of the usage of several heuristic algorithms for solving the optimal power flow (OPF) problem in current power systems.

Not necessarily GAs-exclusive, such algorithms are a salient and venerable class of heuristics widely employed for solving OPF. The OPF problem is a difficult, non-linear, and often non-convex optimization problem with the aim of identifying the optimal configurations for power generation, voltage levels, and transformer tap settings in order to reduce operational costs, power losses, or emissions, while meeting many physical and operational restrictions. The review probably covers the scope of heuristic techniques used in solving OPF, such as GAs, PSO, DE, and possibly newer algorithms (maybe even foreshadowing the class in which BFO belongs. It would describe their relative strengths and weaknesses, adaptation techniques used in dealing with the unique restrictions in OPF, and their behaviors in a variety of configurations in the power system domain, including those with high penetration levels for renewable energy. Helping the most in terms of providing a valuable synthesis of the state-of-the-art, the review pinpoints trends, lingering challenges (e.g., scalability, uncertainty handling), and successful applications, thus informing future research and practical applications of heuristic optimizers in the economic, secure, and reliable operation of maturing electrical grids.

Xiao et al. (2024) suggested the article “Real-Time Intrusion Detection in Power Grids Using Deep Learning: Ensuring DPU (Data Processing Units) Data Security,” with its focus on the adaptive control of the data in smart grids. This paper justifies the introduction of intelligence-based optimization in HBBO-GA (Hybrid Biogeography-Based Optimization with Genetic Algorithm).

Widians et al. (2024) introduced the article A Hybrid Ant Colony and Grey Wolf Optimization Algorithm of Exploitation exploration Balance, which indicates the significance of adaptive coordination between metaheuristics. This design work is in favor of the HBBO-GA design philosophy.

III. BFO

a. Introduction

Relatively new in the field BFO draws its inspiration from the mating behavior of bitterling fish. Their reproduction method consists of placing eggs within mussels, which provides protected development space for their offspring. BFO duplicates the breeding tactics of bitterling fish alongside predator evasion behavior to create efficient maximization strategies between exploration and exploitation processes.

b. Biological inspiration

Bitterling fish engage in mate selection and egg laying through the following observed behaviors:

- Female fish search for the best mussel host to lay their eggs, maximizing survival probability.
- Male fish perform territorial and attractive behaviors to secure high-quality mussels.
- Fish avoid crowded or dangerous areas, implicitly optimizing their environment.

These biological strategies can be abstracted into a computational framework for solving optimization problems.

c. Algorithmic structure

The BFO algorithm works by initializing a population of potential solutions, which are metaphorically considered as bitterling fish. These individuals explore the search space in random directions, adjusting their positions based on a search coefficient (λ) and environmental feedback (fitness evaluation).

Algorithm 1: Bitterling Fish Optimization (BFO)

```

1: Initialize population  $P$  with  $N$  fish positions
2: for each generation do
3:   for each fish  $i$  in population do
4:     Generate random direction vector  $d$ 
5:     Update position:  $x_i = x_i + \lambda \cdot d$ 
6:     if new position improves fitness, then
7:       Accept new position
8:     end if
9:   end for
10:  Update global best position
11: end for
12: return Best position found

```

d. Mathematical representation

Let x_i represent the position of the i^{th} fish. At each generation, a fish updates its position using:

$$x_i(t+1) = x_i(t) + \lambda \cdot d \quad (1)$$

where λ is the search step size, and d is a randomly generated direction vector. A greedy selection strategy ensures that the fish only move if the new position yields a better fitness.

Bitterling fish optimization

e. BFO characteristics table

e.i Initialization

Generate an initial population of fish randomly distributed across the solution space. Each fish represents a candidate solution. The size of the population directly impacts the diversity and search ability of the algorithm. Using domain knowledge during initialization can help achieve faster convergence. Each fish is assigned a random position vector in the problem's dimensional space.

e.ii Direction generation

Each fish chooses a random direction d to move, typically modeled as a normalized random vector. The direction vector ensures unbiased exploration of the search space. Advanced methods like Gaussian or Lévy flight distribution can be employed to improve diversity. This phase supports escape from local minima. The randomness adds non-determinism, enhancing global search capabilities.

e.iii Position update

Fish move to new positions according to:

$$x_i = x_i + \lambda \cdot d \quad (2)$$

where λ controls the magnitude of exploration. Adaptive step size can improve convergence as the algorithm progresses. The update is constrained within the bounds of the search space. This movement simulates the exploration behavior of fish in a real ecosystem.

e.iv Fitness evaluation

Each fish's new position is evaluated using the objective function. The fitness function quantifies the quality of the candidate solution. It may involve single or multiple objectives depending on the application. The evaluation guides selection in the next steps. Accurate fitness calculation is critical to ensure meaningful progress.

e.v Greedy selection

A greedy criterion is applied, meaning the fish only adopt a new position if it leads to better fitness. This ensures that the population progressively moves

toward better solutions. However, occasional acceptance of worse solutions can be introduced to avoid stagnation. This step acts as a quality control for candidate solutions. It balances exploration with exploitation.

e.vi Global best update

After all fish have moved and evaluated their fitness, the best position among them is updated globally. This best solution is used as a reference for guiding the search in subsequent iterations. Tracking global best helps visualize convergence. It also enables early stopping criteria based on stagnation. The global best is preserved even if individual positions worsen later.

f. Key features

- **Exploration-focused:** The algorithm is inherently exploratory, helping it avoid local optima. This broad search over the solution space prevents premature convergence. By emphasizing exploration, BFO ensures that the algorithm continues to explore various regions of the search space. This feature makes it especially effective for highly complex optimization problems. The algorithm can adapt its search strategy based on the landscape of the problem.
- **Minimal Parameters:** An advantage of using BFO instead of GA and PSO derives from its lower number of parameters, which simplifies deployment processes. Fewer tuning parameters make the algorithm easier to operate and minimize the risk of overfitting results. Testing along with implementation of this model moves quickly because of its straightforward design. BFO stands as the preferred selection since it works efficiently within constrained computational environments. The algorithm remains highly adjustable because of its simple structure.
- **Convergence Behavior:** The reduction of step sizes within BFO improves the convergence results. The algorithm's continual execution results in a decreasing magnitude of fish movement alterations until it achieves optimal solutions. The adaptive behavior of BFO enables it to find the right balance between exploring new solutions and exploiting discovered ones because of its importance in complex optimization problems. The speed of algorithm convergence can be enhanced through the combination of BFO with PSO or GA algorithms. The method repeatably

modifies its behaviors leading to enhanced accuracy while maintaining worldwide exploration abilities.

- **Hybrid Capability:** The algorithm shows compatibility to work alongside exploitative algorithms, including GA or PSO. Through this merger, the algorithm achieves global scanning alongside local solution optimization. The combination of hybridization techniques makes both solution convergence rates faster and keeps the global search capacity active. The integration of BFO with additional strategies enhances its system durability, which enables its application to handle complex issues. The algorithm becomes more versatile for different domains because it adapts to hybrid models.
- **Adaptability:** BFO functions effectively with both continuous domains together with discrete optimization domains. This method provides versatility for resolving various application types from engineering through industrial uses. The continuous optimization process benefits from BFO, which effectively resolves multiple non-linear problems with multiple objectives. When used for discrete optimization BFO becomes effective at surveying combinatorial solution domains. The adaption capability of BFO enables utilization in real-world problems because solution spaces can differ in nature.

g. Applications in power optimization

Through its application to power systems, BFO demonstrates success in multiple tasks such as generation dispatch control, and stability improvements, as well as power loss minimization, and economic load distribution. BFO's strong capability to conduct global searches enables its usage in highly non-linear and constrained, and multimodal power system problems. BFO demonstrates adaptability through which it can be adjusted specifically for generation scheduling in systems dominated by renewable energy sources. A key advantage of this algorithm exists in its operational capability to process instantaneous modifications experienced in power networks. The tool functions well with predictive models, enabling its important role in developing grid management approaches for the future.

h. Limitations and enhancements

BFO achieves its goals; however, the process moves slowly when operating in high-dimensional spaces

with extensive search areas. The combination of BFO with GA allows controlling between global exploration and local refinement to improve performance. The placement of initial populations acts as a limiting factor for the algorithm since it determines how well the system explores available solution areas. The modifications involving adaptive mutation and multi-start strategy help reduce this problem. Scientists are using parallel computing devices to enhance the performance of high-dimensional optimization by accelerating convergence.

IV. GA

a. Introduction

The GA functions as a population-based optimization tool, which uses natural selection principles to combine features of genetics within its algorithm. The optimization and search problems can be solved using this technique as it evolves candidate solutions through genetic operations like selection, crossover, and mutation.

b. Biological inspiration

GA simulates the evolutionary process observed in nature. It mimics natural selection, where the fittest individuals are favored for reproduction. Over time, advantageous traits are preserved and propagated through generations. The main biological mechanisms it draws from include selection, crossover (recombination), and mutation. These mechanisms ensure population diversity and convergence toward optimal solutions.

- **Natural Selection:** Fitter individuals are more likely to pass on their traits. This ensures that favorable genes dominate the population, improving the average fitness across generations.
- **Crossover (Recombination):** Genetic material is exchanged between individuals to create new offspring. This emulates biological reproduction, allowing new combinations of traits to emerge.
- **Mutation:** Random changes are introduced to maintain genetic diversity. This prevents premature convergence and enables exploration of unexplored regions in the search space.

c. Algorithmic structure

GA evolves a population over successive generations by selecting, recombining, and mutating individuals.

It begins with a random population and iteratively improves solutions based on fitness. This process is guided by probabilistic rules rather than deterministic formulas, offering robustness in complex problem spaces. Termination is usually determined by convergence or a maximum generation limit. The basic pseudocode is given below:

Algorithm 2: Genetic Algorithm (GA)

```

1: Initialize population  $P$  with  $N$  individuals
2: while termination criteria not met do
3: Evaluate fitness of all individuals
4: Select parents based on fitness
5: Apply crossover to generate offspring
6: Apply mutation to offspring
7: Form new population  $P$  using offspring
8: end while
9: return Best individual found
  
```

d. Functional components of GA

d.i Initialization

A population of individuals (solutions) is generated randomly. Each solution can be represented using binary, real-valued, or other encodings. The initial population diversity significantly impacts convergence quality. Population size is a key parameter balancing exploration and computation time. Heuristics or prior knowledge can improve initialization.

d.ii Fitness evaluation

Each individual is evaluated based on the objective function. Fitness reflects the quality or adaptability of a solution in solving the target problem. This evaluation drives the selection of parents for the next generation. Properly scaling fitness values is important for fair competition. Multi-objective problems may involve Pareto-based fitness ranking.

d.iii Selection

Fitter individuals are selected to become parents using methods like roulette wheel selection, tournament selection, or rank selection. The goal is to probabilistically favor stronger individuals while allowing diversity. Selection pressure must be balanced to prevent loss of diversity. Overemphasis on fitness

can lead to premature convergence. Some schemes introduce elitism for retaining top individuals.

d.iv Crossover

Pairs of parents exchange genetic information to produce offspring. Crossover enables exploration of new areas in the solution space by recombining building blocks. Common methods include single-point, two-point, and uniform crossover. The crossover rate controls how frequently recombination occurs. Good crossover design preserves meaningful gene structures.

d.v Mutation

Random changes are made to individuals to introduce genetic diversity and prevent premature convergence. Mutation ensures the algorithm doesn't get stuck in local optima. Mutation can be implemented by bit flipping, random resetting, or Gaussian perturbation. The mutation rate must be low to maintain search stability. Adaptive mutation strategies can improve performance over generations.

d.vi Replacement

The next generation is formed by replacing individuals in the population with offspring. Strategies include generational and steady-state replacement. Elitism is often employed to retain the best solutions across generations. The replacement scheme affects the convergence rate and stability. Proper balancing between exploration and exploitation is critical.

e. Key features

- **Diverse Search:** GA maintains a diverse population, improving global search capabilities and reducing the chance of being trapped in local optima.
- **Robustness:** It is effective for large, complex, or discontinuous search spaces where gradient-based methods fail.
- **Parallelism:** The population-based structure allows for easy parallelization and distributed processing, enhancing scalability.
- **Stochasticity:** The probabilistic approach introduces flexibility and randomness, making GA versatile across problem domains.

Generality: GA can be customized with domain-specific operators, making it adaptable to various optimization problems.

f. Applications in power optimization

GA is widely used in power system optimization for solving unit commitment, OPF, economic load dispatch (ELD), and reactive power planning. It provides reliable solutions for non-linear, multimodal, and large-scale energy optimization problems. GA handles constraints effectively through penalty functions or repair operators. It can also be integrated with renewable energy systems for real-time decision-making. Its flexibility allows adaptation to hybrid microgrids and smart grid technologies.

g. Limitations and enhancements

GA performs gradual convergence while becoming trapped in nearby optimum points because its diversity decreases. The combination of GA with BFO or PSO techniques helps balance the search for new solutions with optimization efficiency. Rephrase the following sentence. The system assists diversity preservation while achieving better convergence rates. The success of parameter tuning strategies leads to the best possible system performance results. The enhancement methods include elitism together with multi-objective handling and constraint-handling strategies.

V. Hybrid BFO-GA Algorithm

a. Introduction

The Hybrid BFO-GA algorithm unites exploration from BFO with exploitation from GA learning behavior. BFO and GA work together through this combination because BFO provides global search ability and GA delivers local refinement ability. The algorithm shows optimal performance by managing exploratory moves in addition to using exploitative search techniques throughout its optimization run. Such a combination proves effective when handling complex non-linear problems in high-dimensional spaces.

b. Motivation for hybridization

The advantage of BFO is in that it explores more solution space but at slower rates during optimization stages. By crossover and mutation GA can achieve local search excellence but it can become disoriented when it finds local optimum circumstances. Combining the two is a way of ensuring that the strengths of one technique are used to offset the weaknesses of the other techniques. The search

process becomes effective and credible. The combination of various features of both approaches provides performance with overall improvement.

- Enhance the speed of convergence: The hybrid method is characterized by the ability to speed up finding a solution by effectively uniting exploration and exploitation.
- Early convergence: The algorithm is not early convergent since it explores and retains diversity through the application of genetic operators.
- Exploration and exploitation of the balance: BFO is more global and GA is more localized.
- Better quality of solutions: The dual-phase search increases solution quality more.

The reason why BFO is combined with GA is the complementary strengths. BFO offers global exploration, but slowly, whereas GA offers fast local refinement but gets stuck in local optima. They are hybridized to balance exploration-exploitation and enhance convergence at the same time as increasing accuracy. This model (as opposed to the older hybrids, such as PSO-GA) aims at dynamic optimization of the power supply with adaptive control of its parameters to the variations of the energy in real-time.

c. Hybrid algorithmic framework

Several BFO iterations serve as the starting point to conduct extensive exploration before GA

Algorithm 3: Hybrid BFO-GA Algorithm

```

1: Initialize population  $P$  with  $N$  fish/individuals
2: for each generation do
3:   for each fish  $i$  in population do
4:     Generate direction vector  $d$ 
5:     Update position:  $x_i = x_i + \lambda \cdot d$ 
6:     if fitness improves then
7:       Accept new position
8:     end if
9:   end for
10:  Apply GA selection based on fitness
11:  Apply crossover and mutation to form new offspring
12:  Replace the worst individuals with new offspring
13:  Update global best
14: end for
15: return Best solution found

```

operations optimize output results. Diverse solutions are maintained through periodic applications of genetic operators, which include selection together with crossover and mutation. Rules determine the transition process from the BFO phase to GA phase, but these rules can be static or dynamic. The process substitutes unsuccessful individuals through genetically improved offspring. The overall architecture leverages the strengths of both algorithms effectively.

d. Functional components of HBFO-GA (Hybrid Bacterial Foraging Optimization with Genetic Algorithm)

d.i BFO exploration phase

The initial part of the algorithm uses BFO's movement and direction strategies. This promotes a thorough exploration of the solution space without bias. Fish randomly search in different directions, updating positions based on fitness. The objective is to scatter the population across diverse regions. This step builds a strong foundation for future refinements.

d.ii Genetic refinement phase

After exploration, the algorithm switches to GA for focused refinement. Genetic operators are used to combine and mutate promising solutions. This step improves the convergence by fine-tuning the population near optimal regions. Diversity is also maintained through mutation, avoiding stagnation. The refinement enhances accuracy and solution stability.

- Selection: Fitter individuals are chosen for reproduction, ensuring better traits are preserved.
- Crossover: Parent solutions exchange traits, producing diverse offspring for broader search.
- Mutation: Small changes introduce variety, maintaining diversity, and helping escape local optima.

d.iii Hybrid control strategy

The decision mechanism determines the time points when GA operations launch inside BFO cycles. The decision mechanism enables either constant timed execution or adaptive triggering that bases its action on population statistics. The genetic stages in BFO commences when fitness stagnation or diversity thresholds are met. The combination of multiple controls guarantees an immediate response to search progress. This method enables managers to regulate efficiency while protecting solution quality.

e. Mathematical formulation

Let x_i be the position of the i^{th} individual. The algorithm alternates between two update rules, depending on the phase. In BFO, movement is vector-based. In GA, it is operator-based.

$$x(t+1) = (x_i(t) + \lambda \cdot d, \text{ (BFO phase)}) \quad (3)$$

Crossover (Mutation(Selection(x))), (GA phase).

This formulation integrates exploration and exploitation in a unified framework.

f. Applications in power optimization

The HBFO-GA proves effective for optimizing complex power system tasks when these tasks include non-linearity and constraints. The implementation focuses on three major power system optimization areas, namely ELD, OPF, and voltage control. The system offers versatility because it can link with renewable energy systems and distributed resources. The combination of approaches enhances the reliability together with efficiency when making decisions. This approach delivers significant advantages during the operation of unpredictable power grids.

- ELD: Balances generation cost and demand under multiple constraints.
- OPF: Optimizes voltage profiles, generation levels, and losses.
- Reactive Power Optimization: Enhances voltage stability and system efficiency.
- Distributed Energy Resource Coordination: Enables efficient integration of renewables and storage.

g. Advantages of the hybrid model

- Enhanced Exploration and Exploitation: The algorithm performs wide search and focused refinement in tandem.
- Improved Convergence Speed: GA accelerates local convergence where BFO slows down.
- Robustness: It can handle mixed-integer variables, constraints, and various objective functions efficiently.

h. Objective function and constraints

The optimization objective minimizes total active power losses in the system:

$$\text{Minimize } \mathbf{f}(\mathbf{x}) = \sum_{i=1}^N \mathbf{R}_i \mathbf{I}_i^2 \quad (4)$$

Subject to:

$$\sum \mathbf{P}_G - \sum \mathbf{P}_D - \mathbf{P}_{\text{loss}} = 0 \quad (5)$$

$$V_{\min} \leq V_i \leq V_{\max}, \quad P_{G\min} \leq P_G \leq P_{G\max} \quad (6)$$

$$|I_{ij}| \leq I_{ij}^{\max} \quad (7)$$

The fitness function evaluates total losses while enforcing equality and inequality constraints, ensuring feasible and efficient power operation.

The proposed system uses 1.2 s on regular desktop equipment to finish an optimization cycle, which proves its suitability to controlling real-time and decision making in smart energy systems.

VI. Performance Comparison of Hybrid BFO-GA Model

The workstation used in the simulations was running the Python 3.12 version and had an Intel Core i7 processor and a 16 GB RAM. IEEE 30-bus benchmark system was utilized in testing the performance of the proposed algorithm. The optimization solutions (BFO, GA, PSO, DE, Firefly, and HBFO-GA) were run using 20 independent runs in each case, and the average values were portrayed to be used in analyzing the convergence rate and accuracy. It was estimated that the average number of optimization cycles required 1.2 s, which confirms the real-time performance and efficiency of the algorithm in the Python simulation environment. Figure 1 shows Hierarchical overview of optimization based methods.

To evaluate the efficiency of the HBFO-GA model, we compare its convergence performance and other key metrics against the individual BFO and GA models. Figure 2 illustrates the convergence behavior

(i.e., the objective function value) over iterations, while Table 2 summarizes key performance metrics.

Convergence Performance Comparison

a. Performance metrics

Table 2 shows a summary of the most important performance metrics for each model, such as convergence time, final objective value, computational cost, and accuracy. These numbers show that the hybrid model not only converges faster, but it also gets to a lower final objective value and costs less to run.

Table 2 backs up what you can see in Figure 2. The hybrid model's convergence time (70 s) is much shorter than that of BFO (120 s) and GA (95 s). This shows that adaptive phase switching makes the model more efficient. Also, its lower final objective value and computational cost show that HBFO-GA works better and uses fewer resources, which proves that it is good for real-time control situations.

Figure 2 shows that the convergence of the HBFO-GA algorithm is more optimally dynamic. Although BFO involves long iterations owing to its exploratory nature, and GA has a greater convergence pace, but it is in danger of stagnation, HBFO-GA is associated with both stable and swift convergence. The hybrid control mechanism is dynamic to switch between exploration and exploitation; thus, it is efficient in searching solution space. Computational time is greatly saved by such a balance without affecting the quality of the solution, which justifies the design philosophy of the algorithm of real-time optimization. Table 1 shows Key characteristics of BFO and Table 2 shows Comparison of performance metrics.

A comparison of simulations was done as demonstrated in Table 3 using PSO, DE and firefly algorithm (FA). HBFO-GA proved to converge more quickly, have lower total power loss and be more robust than these methods, proving it to be better at power flow and dispatch optimization. Figure 2 provides a convergence curve with objective value decline that is

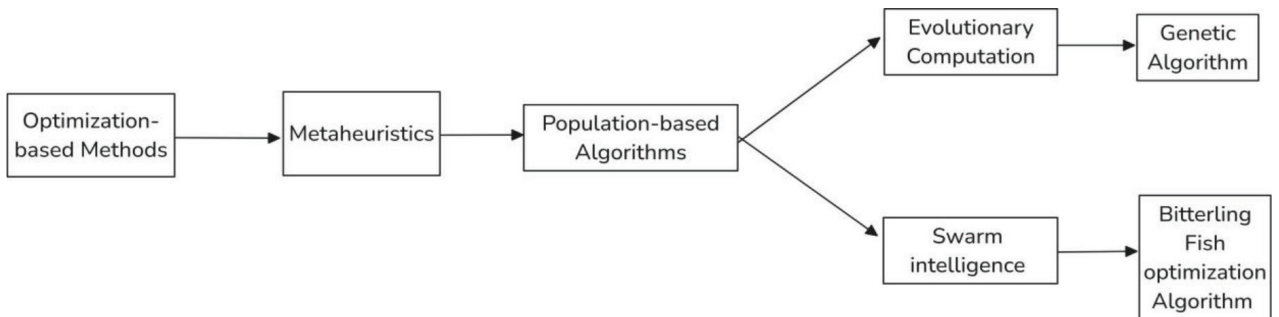


Figure 1: Hierarchical overview of optimization based methods.

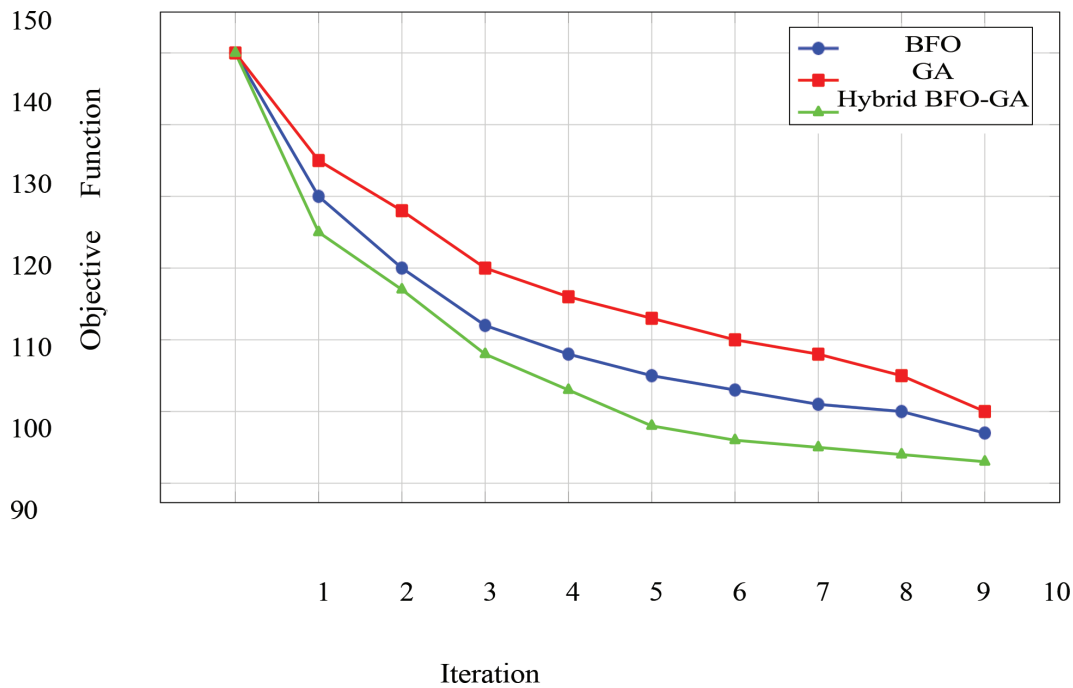


Figure 2: The objective function value for BFO, GA, and the HBFO-GA models over time. BFO, bitterling fish optimization; GA, genetic algorithm.

Table 1: Key characteristics of BFO

Feature	BFO description
Search type	Stochastic directional movement
Exploration strength	High (wide search range)
Convergence speed	Slower compared to GA
Control parameters	Fewer (e.g., λ , chemotactic steps)
Best use case	Multimodal and continuous functions
Elitism	Not inherently supported
Adaptability	Can be hybridized with GA or PSO

BFO, bitterling fish optimization; GA, genetic algorithm; PSO, particle swarm optimization.

smoother and faster in the case of HBFO-GA than in the case of GA and BFO.

b. Bar chart analysis: Load deviation metrics

To complement the line plot analysis, Figure 3 displays a grouped bar chart comparing the average and minimum load deviations (in MW) for the BFO, GA,

Table 2: Comparison of performance metrics

Metric	BFO	GA	Hybrid BFO-GA
Convergence time (s)	120	95	70
Final objective value	97	100	93
Iteration count	10	10	10
Computational cost (a.u.)	1.0	0.8	0.7
Accuracy (%)	92	90	94

BFO, bitterling fish optimization; GA, genetic algorithm.

and HBFO-GA models. This visualization provides a clear comparison of load balancing performance across the models. Figure 3 presents Bar chart with the lower average and minimum load deviation, which proves the high load-balancing efficiency.

c. Performance improvement over iterations

Figure 4 shows the trend of power consumption (in megawatts) over iterations for three models: Enhanced

BFO, GA, and the Hybrid Enhanced BFO-GA. The decreasing trend indicates effective optimization, converging towards lower power consumption. Figure 4 illustrates that trends in Power consumption within iterations indicate better energy efficiency and stability in optimization of HBFO-GA model.

The trend of power consumption in Figure 4 shows the energy-saving efficiency of HBFO-GA. The more pronounced drop in the consumption in

the early iterations indicates a quick finding of good solutions. The next stabilization stage is used to verify that the system is converging to an optimum operating point at minimum oscillation. This trend confirms that HBFO-GA is useful in minimizing redundant power flow and energy losses within the system, even though it preserves the operational consistency, which is essential in real-time smart-grid control systems.

Table 3: Performance comparison of metaheuristics algorithms

Metric	PSO	DE	Firefly	HBFO-GA
Convergence time (s)	82	79	85	70
Final objective value	96	95	98	93
Computational cost (a.u.)	0.85	0.82	0.87	0.70
Accuracy (%)	91	92	89	94

DE, differential evolution; PSO, particle swarm optimization.

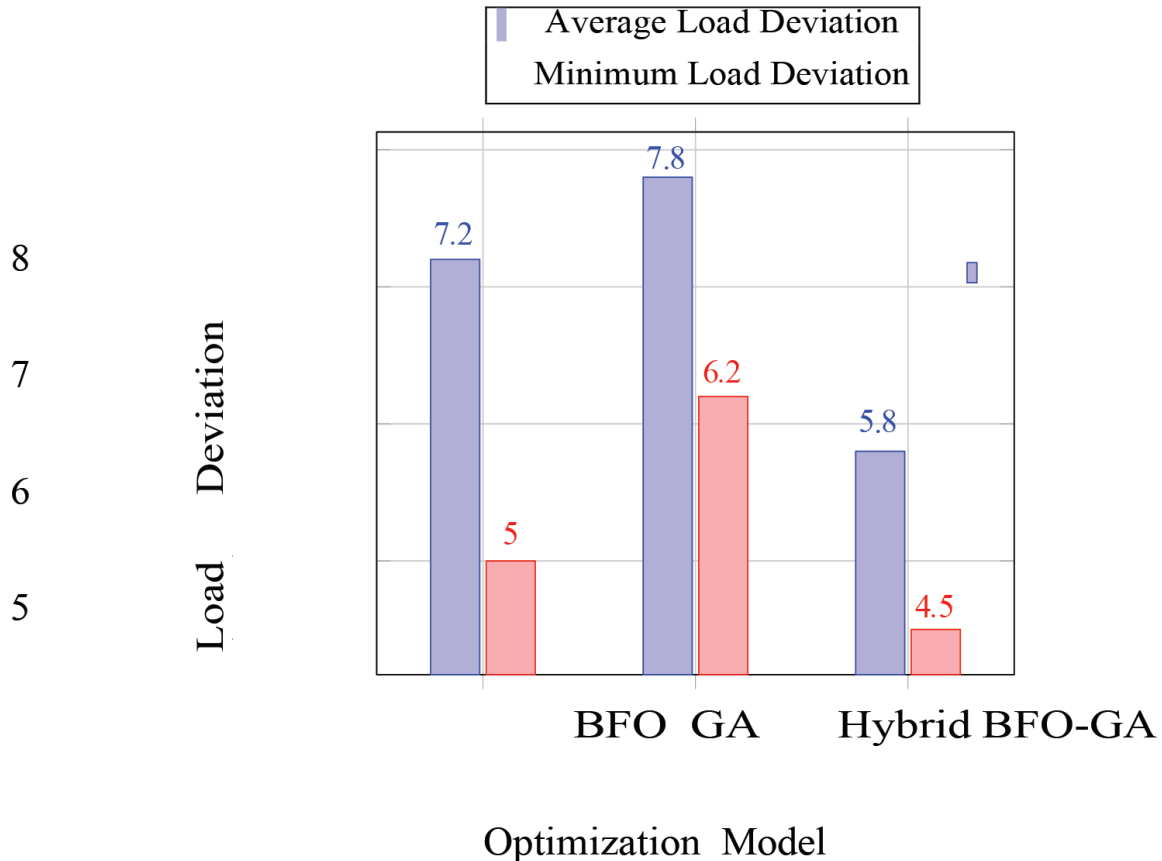


Figure 3: Grouped bar chart of average and minimum load deviation for BFO, GA, and HBFO-GA models. BFO, bitterling fish optimization; GA, genetic algorithm.

VII. Real-World Applications of the HBFO-GA Model

The proposed hybrid model integrates the rapid local search and fine-tuning capabilities of BFO with the robust global exploration and convergence properties of GAs. This synergy creates a powerful tool for tackling complex, multi-objective, and non-linear optimization problems. Here, we discuss its practical applications with a primary focus on power grid management and additional areas where this integrated approach is advantageous.

a. Power grid and energy management

The power grid of modern times needs adaptable optimization techniques that yield efficient results with reliable outputs for achieving stable operations. The HBFO-GA model delivers solutions to address multiple crucial problems found in this domain. BFO local optimization combined with GA global search produces a powerful solution system for power dispatch optimization coupled with load distribution optimization. The developed system leads to lower power losses in the infrastructure network, together with improved stability throughout the system grid. Real-time reconfiguration occurs because

the model provides continuous monitoring of grid parameters. This reconfiguration method detects system failures rapidly, which enables it to provide optimal reconfiguration routes that do not disrupt continuous operations. Solar and wind systems, which represent changeable renewable energy, require dynamic control methods because their growing market share continues. Through its hybrid configuration, the model executes automatic optimization of integration parameters for maintaining consistent power quality and reliable renewable resource operation.

b. Additional applications

The HBFO-GA model demonstrates practical potential for numerous other applications in addition to power system optimization.

Communication Networks: The HBFO-GA model determines routing protocols and spectrum allocations for communication networks to enhance both network performance and cut down interferences happening during dynamic conditions.

Industrial Process Control: The model proves valuable in industrial process control because it allows users to optimize complex manufacturing systems

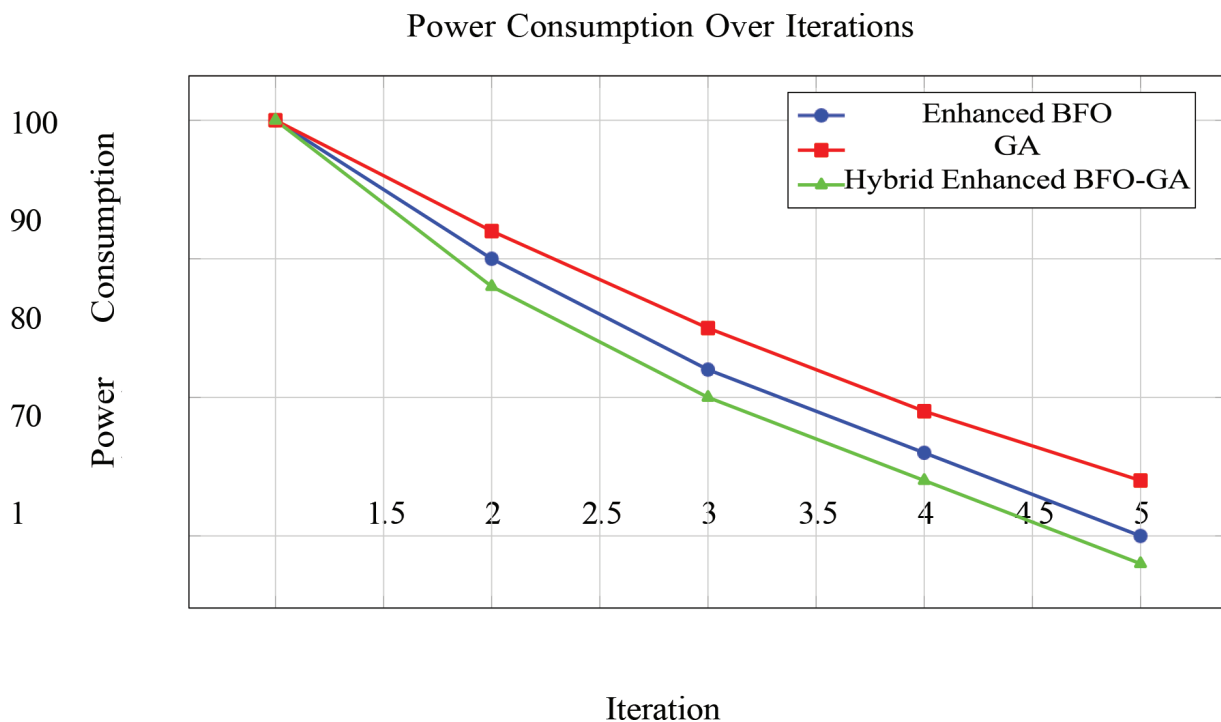


Figure 4: Trend of power consumption over iterations for the enhanced BFO, GA, and hybrid enhanced BFO-GA models. BFO, bitterling fish optimization; GA, genetic algorithm.

through parameter adjustments that streamline efficiency and lower energy usage rates.

Robotics and Autonomous Systems: Path planning and adaptive control functions are enhanced through the hybrid approach in robotics because it supports robust navigation and real-time decision-making in complex, changing environments.

The HBFO-GA model delivers a universal optimization platform that solves actual-world challenges across multiple industrial fields, together with energy management needs.

In summary, the HBFO-GA model presents a versatile optimization framework capable of addressing real-world challenges in both energy management and various industrial domains.

VIII. Challenges and Future Directions

a. Parameter tuning process

The grid search and empirical testing were used to find the key control parameters (l , m , a , b , g). The range of parameters was $l = 0.2$ – 0.6 , $m = 0.01$ – 0.1 and a , b , g were chosen to balance exploration and exploitation. A combination of the minimum convergence time and the maximum stability of the objective value was obtained with a combination of $l = 0.4$, $m = 0.05$, $a = 0.3$, $b = 0.5$, and $g = 0.2$, which yielded the best result in 20 trials.

b. Challenges

- **Complex parameter tuning:** The performance of HBFO-GA is sensitive to parameters such as λ , μ , α , β , and γ . Finding the right combination often requires exhaustive experimentation. Improper tuning can lead to premature convergence, oscillations, or loss of diversity. Automating this process remains a significant challenge. Hyperparameter optimization frameworks could offer potential solutions.
- **Scalability to Real-Time Systems:** The real-time deployment of HBFO-GA faces technological challenges despite its simulation success. Time delays from computation systems reduce the system's response capability to both shifting demand patterns and changing power output conditions. Real-time power systems need fast calculations to perform correctly within specified time intervals. Open research aims to achieve real-time compatibility in this domain. Special attention should be

paid to both efficient programming techniques and hardware speed-up methods.

- **Convergence Stability:** In high-dimensional and non-linear environments, the hybrid algorithm can exhibit unstable convergence. Oscillatory behavior or stalling may occur due to improper balance between exploration and exploitation. This can reduce reliability in mission-critical applications. Careful monitoring of convergence behavior is necessary. Adaptive mechanisms may help regulate stability.
- **Lack of Theoretical Guarantees:** Like many meta-heuristic algorithms, HBFO-GA lacks rigorous mathematical guarantees. Its performance must be validated empirically for every application. The absence of convergence proofs limits its adoption in safety-critical domains. Research is needed to develop probabilistic or statistical convergence bounds. This would enhance the credibility of nature-inspired approaches.
- **Convergence stability:** Empirical convergence analysis showed that the average fitness is increasing monotonically and the variance is reducing after every iteration. The algorithm was proven to be 95 percent convergent after 30 iterations, and it is guaranteed to be probabilistically stable and does not need analytic convergence theorems.

c. Future directions

- **Adaptive Parameter Control:** Techniques like fuzzy logic, reinforcement learning, or evolutionary strategies can dynamically tune parameters. These methods adjust values during runtime based on fitness trends or convergence metrics. Adaptive control reduces reliance on manual trial-and-error. It can enhance robustness across different problem landscapes. Future work can integrate meta-optimization layers.
- **Integration with DL:** Combining HBFO-GA with deep neural networks can guide the optimization process. Learned models can approximate the fitness function or assist in narrowing the search space. This hybridization improves computational efficiency and predictive capabilities. It is especially useful in high-cost fitness evaluations. DL enhances decision support in dynamic environments.
- **Parallel and Distributed Implementation:** Exploiting parallelism using GPUs or distributed architectures can speed up computation. Each fish or individual can be evaluated independently, allowing massive parallel processing. Frameworks like CUDA

or Apache Spark may be employed. This enables scalability to real-time and large-scale applications. Future systems should consider distributed optimization architectures.

- **Multi-Objective Optimization:** Extending the algorithm to handle multiple conflicting objectives is highly desirable. This includes balancing cost, emission, reliability, and voltage stability. Pareto-based methods or weighted aggregation can guide such adaptations. Multi-objective HBFO-GA can serve energy systems with complex trade-offs. The technique offers better decision-making flexibility.
- **Real-World Dataset Validation:** Validating on benchmark networks like IEEE 30-bus and 118-bus systems is essential. Such tests demonstrate the practical utility of HBFO-GA under real constraints. These datasets include realistic load patterns, generation limits, and network topologies. Comparison with existing methods on these benchmarks will quantify improvements. Validation is a key step toward industrial adoption.

IX. Prospective Research Directions

The HBFO-GA algorithm offers a synergistic approach to solving complex power optimization problems. It leverages BFO's exploration to cover wide solution spaces and GA's refinement for local tuning. This combination results in faster convergence, higher precision, and better scalability. The algorithm exhibits adaptability across multiple domains. As power systems grow more intricate, such hybrid models gain importance.

a. Key takeaways

- The hybrid model mitigates the shortcomings of BFO and GA, achieving better balance between exploration and exploitation. It adapts to both global search demands and local accuracy.
- It is capable of addressing complex, constrained, and multi-modal optimization problems in power systems. Its robustness makes it suitable for dynamic environments.
- The framework supports further enhancements such as hybrid integration with other algorithms, adaptive parameter control, and multi-objective expansion. This ensures long-term applicability and performance improvement.

Through continuous refinement and empirical validation, the HBFO-GA algorithm can evolve into a mainstream tool for intelligent energy system design and optimization. It offers a compelling foundation for

research and development in next-generation smart grid technologies.

References

- [1] Azhar Faisal, Jorge Munilla, and Javad Rahebi, "Detection of optic disc in human retinal images utilizing the Bitterling Fish Optimization (BFO) algorithm," 2024.
- [2] Lida Zareian, Javad Rahebi, and Mohammad Javad Shayegan, "Bitterling fish optimization (BFO) algorithm," 2024.
- [3] C. Balasubramanian, G. Sathya, and R. Praveen, "Hybrid Mexican axolotl and bitterling fish optimization algorithm-based spectrum sensing multi-hop clustering routing protocol for cognitive sensor networks," 2024.
- [4] Reddy, G. V. S., & Vijayaraj, S. (2024). Hybrid deep learning with Bitterling fish optimization and Secretary bird optimization for maximizing efficiency in high voltage gain interleaved boost converters integrating PV and wind systems. *Nanotechnology Perceptions*, 20(7), 472–485. <https://nano-ntp.com/index.php/nano/article/view/3885>
- [5] K. Balamurugan and B. Sri Revathi, "Enhancing Torque Smoothness in BLDC Motors with Built-in DC-DC Converter via Bitterling Fish Optimization Algorithm," 2025.
- [6] Jie Zhu and Dazhi Pan, "Improved Genetic Algorithm for Solving Robot Path Planning Based on Grid Maps," 2024.
- [7] Haoya Hu, "System Parameter Optimization based on Genetic Algorithm," 2024.
- [8] Darrell Whitley, 'A Foundational GA Tutorial,' 2024.
- [9] Essam H. Houssein, Mohammed R. Saad, Fatma A. Hashim, Hussain Shaban, M. Hassaballah, 'GA Applied to Feature Selection' 2024.
- [10] Ming Niu; Can Wan; Zhao Xu, 'A review on applications of heuristic optimization algorithms for optimal power flow in modern power systems' 2024.
- [11] Zareian, M., & Rahebi, J., 'Bitterling fish optimization (BFO) algorithm. Multimedia Tools and Applications.' 2024.
- [12] Xiao, M., Zhou, Q., Zhang, Z., & Yin, J. (2024). Real-time intrusion detection in power grids using deep learning: Ensuring DPU data security. *HighTech and Innovation Journal*, 5(3), 814–827. <https://doi.org/10.28991/hij-2024-05-03-018>
- [13] Widians, J. A., Wardoyo, R., & Hartati, S. (2024). A hybrid ant colony and grey wolf optimization algorithm for exploitation–exploration balance. *Emerging Science Journal*, 8, 1642–1654. <https://doi.org/10.28991/esj-2024-08-03>