Supervised By:

**Syed Qamar Askari**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| LITERATURE REVIEW 1.0  Global Optimization Using Meta-Heuristics | |  |  | | --- | --- | | Faiza Shanawar | 15140070 | | Haider Ali | 15140101 | | Mohsin Qamar | 15140104 | | Usama Imran | 15140098 | |

Contents

[1. Introduction: 3](#_Toc6443518)

[1.1. Mathematical Optimization: 3](#_Toc6443519)

[1.1.1. Types: 4](#_Toc6443520)

[1.1.2. Applications: 4](#_Toc6443521)

[1.2. Meta-Heuristics: 4](#_Toc6443522)

[1.3. Properties of Meta-Heuristic Algorithms: 5](#_Toc6443523)

[1.3.1. Exploration: 5](#_Toc6443524)

[1.3.2. Exploitation: 5](#_Toc6443525)

[1.3.3. Convergence: 5](#_Toc6443526)

[1.4. Benchmark Functions: 5](#_Toc6443527)

[1.4.1. Types: 5](#_Toc6443528)

[1.4.2. Overview: 5](#_Toc6443529)

[2. No Free Lunch Theorem: 9](#_Toc6443530)

[3. Literature Review: 9](#_Toc6443531)

[3.1. Genetic Algorithm: 9](#_Toc6443532)

[3.1.1. Inspiration: 9](#_Toc6443533)

[3.1.2. Working: 9](#_Toc6443534)

[3.1.3. Applications: 10](#_Toc6443535)

[3.2. Gravitational Search Algorithm: 11](#_Toc6443536)

[3.2.1. Inspiration: 11](#_Toc6443537)

[3.2.2. Working: 11](#_Toc6443538)

[3.2.3. Experimental Results: 12](#_Toc6443539)

[3.2.3.1. Unimodal Functions: 13](#_Toc6443540)

[3.2.3.2. Multimodal Functions: 13](#_Toc6443541)

[3.2.3.3. Multimodal with fixed Dimensions: 14](#_Toc6443542)

[3.2.4. Applications: 14](#_Toc6443543)

[3.3. Gray Wolf Optimization: 15](#_Toc6443544)

[3.3.1. Inspiration: 15](#_Toc6443545)

[3.3.2. Working: 15](#_Toc6443546)

[3.3.3. Experimental Results: 16](#_Toc6443547)

[3.3.3.1. Unimodal functions: 17](#_Toc6443548)

[3.3.3.2. Multimodal Functions: 17](#_Toc6443549)

[3.4. Soccer League Optimization: 17](#_Toc6443550)

[3.4.1. Inspiration and working: 17](#_Toc6443551)

[3.5. Social Evolution: 18](#_Toc6443552)

[3.5.1. Inspiration and Working: 18](#_Toc6443553)

[3.6. Simulated Annealing: 18](#_Toc6443554)

[3.6.1. Working and inspiration: 18](#_Toc6443555)

[3.7. Particle Swarm Optimization: 19](#_Toc6443556)

[3.7.1. Working and Inspiration: 19](#_Toc6443557)

[3.8. Related Work: 19](#_Toc6443558)

[References 20](#_Toc6443559)

***Document Information***

|  |  |
| --- | --- |
| User |  |
| Project | Meta-Heuristics for Global Optimization |
| Document Version | 1.0 |
| Document ID | AR-02 |
| Identifier | Literature Review 1.0 |
| Status | Draft |
| Authors(s) | Usama Imran |
| Approver(s) | Syed Qamar Askari |
| Issue Date |  |

# Introduction:

## Mathematical Optimization:

Mathematical Optimization can be defined as a technique of finding a maximum or minimum value of a function of several variables subject to set of constraints. More generally, optimization includes finding "best available" values of some objective function given a defined domain (or input), including a variety of different types of objective functions and different types of domains.



Graph of a paraboloid given by

z = f(x, y) = −(x² + y²) + 4. The global maximum at (x, y, z) = (0, 0, 4) is indicated by a blue dot.

### Types:

Optimization is classified into following types:

* Discrete optimization
* Continues optimization
* Single Objective Optimization
* Multi Objective Optimization
* Many Objective Optimization
* Combinatorial optimization
* Unconstrained optimization
* Constrained optimization

### Applications:

Optimization techniques are being used in real world problems some of their applications are as follows:

* Feature Selection
* Automatic Clustering
* Time Scheduling
* Wireless Sensor Network Optimization
* Vehicle Routing Problem
* Watermarking
* Bioinformatics
* Circuit Designing
* Game Strategy Planning
* Power Supply Management

## Meta-Heuristics:

Heuristic is a Greek word which means “to solve”. It pertains to trial-and-error method of problem solving used when an exact algorithmic approach is impractical. Main characteristic of meta-heuristics is that they are problem independent. Meta-heuristics give us a way to solve complex problems that are not solvable in polynomial time (NP-Hard Problems). Although they don’t give exact solution of a particular problem, meta-heuristics provide guidelines that can give best solution available

## Properties of Meta-Heuristic Algorithms:

In order to be successful a search algorithm needs to establish a good ratio between exploration and exploitation[1].

### Exploration:

Exploration is process of visiting entirely new regions of search space [1] . It is the ability to evaluate candidate solutions that are not neighbor to the current solution (or solutions). This operation serves to escape from a local optimum.

### Exploitation:

Exploitation is the process of visiting those regions of a search space within the neighborhood of previously visited points[1]. It is when a search is done in the neighborhood of the current solution (or solutions). It can be implemented as a local search.

### Convergence:

It is often called termination condition of algorithm. More precisely, it stops the algorithm because each individual of population becomes identical as far as their fitness is concerned.

## Benchmark Functions:

After the algorithms are proposed, they are tested on some evaluation functions which are known as benchmark functions. Algorithms are subjected to benchmark functions their performance is checked through comparison with already existed algorithms.

### Types:

Following are the types of benchmark functions:

1. Unimodal
2. Multimodal
3. Differentiable
4. Non-differentiable
5. 1- Dimensional
6. 2-Dimentional
7. 3-Dimensoinal
8. N-Dimensional
9. Convex
10. Non-convex
11. Parametric
12. Separable

### Overview:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Function name*** | ***Graph*** | ***Equation*** | ***Nature*** | ***Range*** |
| Ackley N. 2 Function |  | f(x,y)=−200e−0.2√x2+y2 | N-dimensional, unimodal, convex, differentiable | [-6,6] |
| Booth Function |  | f(x,y)=(x+2y−7)2+(2x+y−5)2 | 2-dimensional, continuous, convex, differentiable, non-separable, unimodal convex | [−10,10] |
| Brent Function |  | f(x,y)=(x+10)2+(y+10)2+e−x2−y2 | 2-dimensional continuous convex differentiable non-separable unimodal convex | [−10,10] |
| Drop-Wave Function |  | f(x,y)=−1+cos(12√x2+y2) (0.5(x2+y2)+2) | 2-dimensional continuous unimodal non-convex | [−5.2,5.2] |
| Exponential Function |  | f(x)=f(x1,...,xn)= −exp(−0.5n∑i=1x2i) | n-dimensional continuous differentiable non-separable unimodal convex | [−1,1] |
| Leon Function |  | f(x,y)=100(y−x3)2+(1−x)2 | 2-dimensional continuous differentiable non-separable unimodal non-convex | [0,10] |
| Deckkers-Aarts Function |  | (x,y)=105x2+y2−  (x2+y2)2+10−5(x2+y2)4 | 2-dimensional continuous differentiable non-separable multimodal non-convex | [−20,20] |
| Styblinski-Tank Function |  | f(x)=f(x1,...,xn)=  12n∑i=1(x4i−16x2i+5xi) | n-dimensional continuous multimodal non-convex | [−5,5] |
| Bartels Conn Function |  | f(x,y)=  |x2+y2+xy|+|sin(x)|+|cos(y)| | 2-dimensional non-separable multimodal non-convex non-differentiable | [−500,500] |
| Schwefel 2.20 Function |  | f(x)=f(x1,...,xn)=n∑i=1|xi| | 2-dimensional continuous differentiable non-separable unimodal non-convex | [−100,100] |
| Egg Crate Function |  | f(x,y)=x2+y2+25(sin2(x)+sin2(y)) | 2-dimensional continuous differentiable separable multimodal non-convex | [−5,5] |
| Shubert Function |  | f(x)=f(x1,...,xn)=  n∏i=1(5∑j=1cos((j+1)xi+j)) | n-dimensional, continuous, differentiable non-separable, multimodal, non-convex | [-10,10] |
|  |  |  |  |  |

# No Free Lunch Theorem:

This theorem states that there is no such algorithm that could give same results on every problem. Different algorithms perform better in different types of problem. Therefore, research in this area is open.

# Literature Review:

## Genetic Algorithm:

### Inspiration:

Genetic Algorithm is one of the most famous optimization algorithms. It is inspired by the Darwin’s Theory of Evolution (natural selection). This algorithm exhibits the process of natural selection until the fittest population is obtained.



### Working:

This algorithm has five phases in which two are major phases: “Crossover” and “Mutation”. Algorithm starts from instantiating a random population and their fitness is calculated by **Objective Function**[[1]](#footnote-1)**.** Each member of population is named “**gene**”.This is 1st phase which can be called as “population initialization”. In the 2nd phase, fittest genes are selected which are to be subjected to crossover and mutation. This 2nd phase is called “**selection**”**.** After the selection of fittest genes, they undergo “**crossover”** phase which is 3rd phase of this algorithm. After crossover, genes go through “**mutation**” phase which is 4th phase of this algorithm. After mutation, fitness is calculated again and fittest genes after crossover and mutation are replaced by least fit genes in initialized population. This process is repeated until fittest population is obtained and then it is terminated in the last phase of this algorithm which is called “**termination**”.



### Applications:

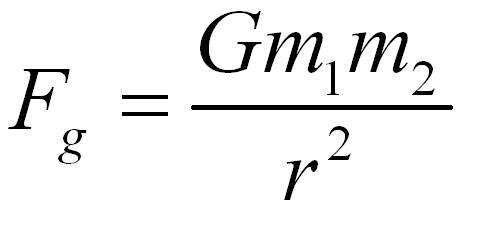
Several problems have been solved by this algorithm, some of them are as follows:

|  |  |
| --- | --- |
| ***Application*** | ***Description*** |
| Product Designing | Product design is increasingly recognized as critical activity that has a significant impact on the performance of firm, when firms undertake a new or existing product or redesign, it is important to employ techniques that will generate optimal solution[2]. |
| Molecular Geometry Optimization | The geometry of a molecule determines many of its physical and chemical properties. This is why it is very important that we understand the geometry of a molecule when running computations[3]. |
| Automotive Design | Automotive design is the process of developing the appearance, and to some extent the ergonomics, of motor vehicles, including automobiles, motorcycles, trucks, buses, coaches, and vans[4]. |
| Optimization of architectural shapes | Architectural design optimization (ADO) is a subfield of engineering that uses optimization methods to study, aid, and solve architectural design problems, such as optimal floorplan layout design, optimal circulation paths between rooms, and the like[5]. |
| Capacitated Vehicle Routing Problem | The vehicle routing problem with simultaneous pickup and delivery considering customer satisfaction is based on a time window at each customer location. In such a problem, the transportation requests have to be performed by vehicles, each request having to be met as early as possible[6]. |

## Gravitational Search Algorithm:

### Inspiration:

Gravitational Search Algorithm is inspired by Newton’s Law of Gravitation which states that every object of mass m1 in a universe attracts other object of mass m2 with a force with is directly proportional to product of masses of objects and inversely proportional to square of distance between them.



### ***Working***:

**Gravitational Search Algorithm** is Physics based algorithm which is, as the name suggests, inspired from Newton’s law of Gravitation. In this algorithm, every solution is treated as an object. And the object’s fitness is determined by the mass of that object, greater the mass, greater would be the fitness. The object of higher fitness attracts the object of lower fitness by following the rules of physics. The attraction of lighter mass towards heavier mass promotes exploration of algorithm.

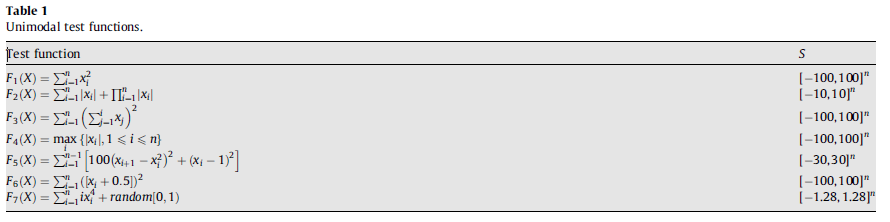


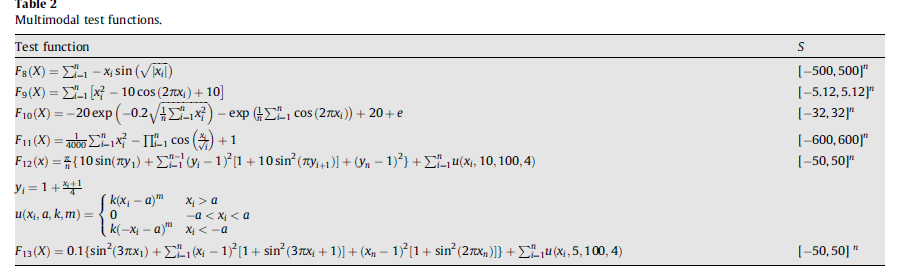
### Experimental Results:

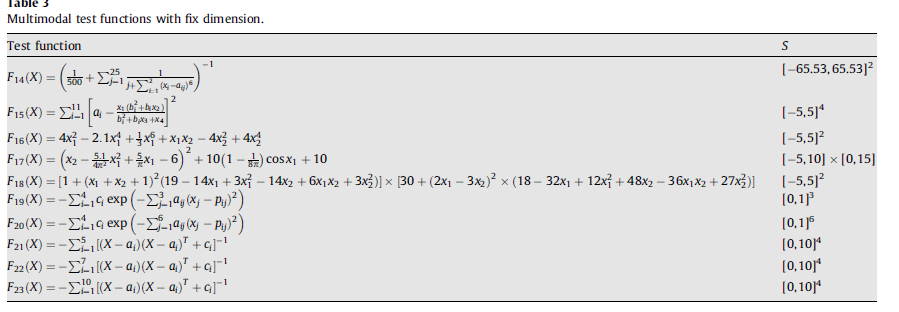
This algorithm was compared with:

* 1. PSO (Particle Swarm Optimization)
  2. CFA ()
  3. RGA (Regularized Global Approximation)

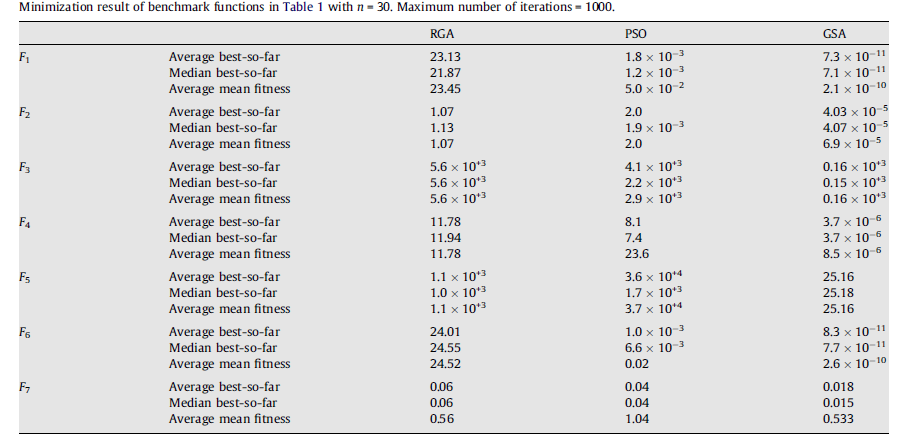
This Algorithm is tested on following benchmark functions:



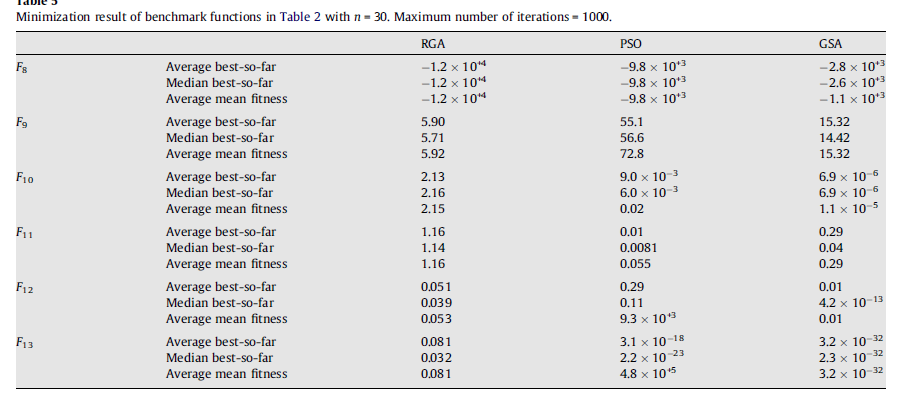




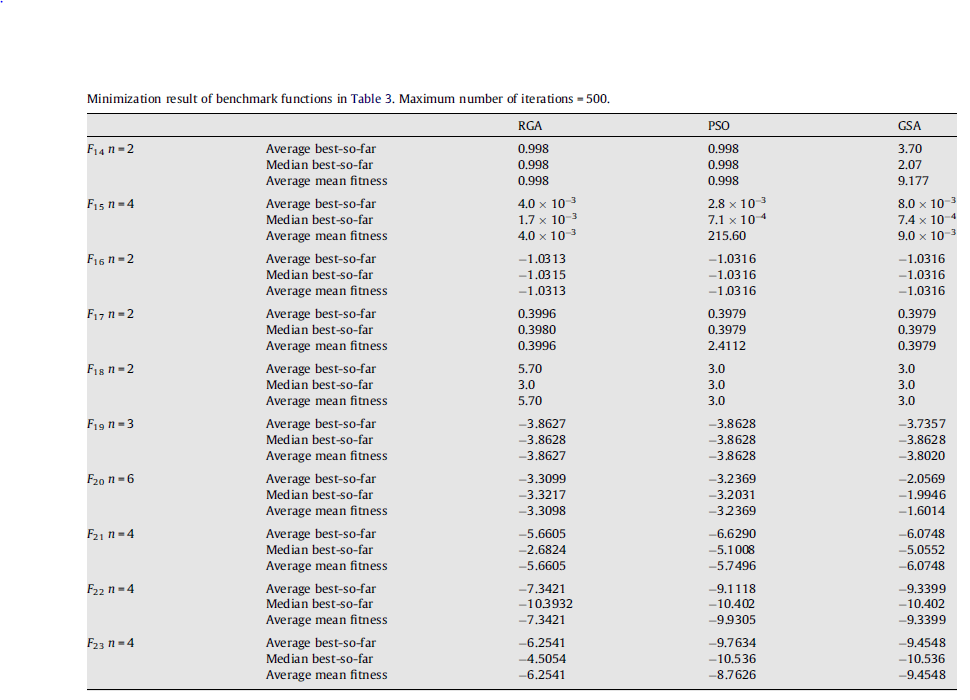
### Unimodal Functions:



### Multimodal Functions:



### Multimodal with fixed Dimensions:



### Applications:

|  |  |
| --- | --- |
| ***Application*** | ***Description*** |
| Economic Load Dispatch Problem (ELD) | Economic Load Dispatch (ELD) is a method of determining the most efficient, low-cost and reliable operation of a power system by dispatching available electricity generation resources to supply load on the system most economically. The ELD problem is multimodal, non-differentiable and highly nonlinear [7]. |
| Energy Management System | EMS problem in an MG including different types of DG units with particular attention to the technical constraints. The EMS consists of a stand-alone wind turbine (WT), photovoltaic (PV), microturbine (MT) and energy storage (ES) system. The objective function for the optimization of the EMS is the total general cost. The application includes the implementation of some variation in load consumption model considering accessibility to the ES and demand response. The GSA provides a good compromise between computation time and precision of the solution[7] |
| Feature Subset Selection | Pattern recognition, data mining and knowledge discovery problems require feature subset selection (FSS) to represent the patterns to be classified (Zalama et al. 2014; Amezquita-Sanchez, Adeli 2015). The FSS problem refers to the task of identifying and discovering a useful subset of features to represent a pattern from a larger set of features which may be redundant and even irrelevant causing unnecessary computational complexity and cost (Jackowski et al. 2014). The research problem is then how to select the minimum subset of features to represent the original knowledge effectively. Thus, FSS can be viewed as a search problem. The FSS problem can be treated as an optimization problem in a search space of 2N. Han et al. (2013) introduced a binary GSA to solve the FSS problem[7]. |
| Unit Commitment | Unit commitment (UC) problem in power systems aims to schedule the most cost-effective combination of generating units to meet the forecasted load and reserve requirements while adhering to generator and transmission constraints. The commitment schedule takes into account the inter-temporal parameters of each generator (minimum run time, minimum down time, notification time, etc.) but does not specify production levels which are determined five minutes before delivery. The determination of these levels is known as economic dispatch and is the least-cost usage of the committed assets during a single period to meet the demand. The objective is to minimize the total system cost of generating power from N units over a specific time Roy (2013) applied GSA to the UC problem[7]. |
| Training the Neural Networks | In general, feed-forward neural network (NN) consists of one input layer, one or more hidden layers with a nonlinear activation function and one output layer with a linear activation function (Adeli, Park 1998). Biases can be set to non-zero or zero. The problem is to find the connection weights of the network for the given architecture to produce the correct output for the function for each corresponding input[7]. |

## Gray Wolf Optimization:

### Inspiration:

This algorithm is inspired by pack of wolves which are going out for hunting.

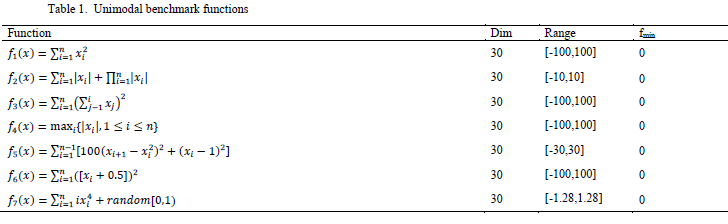
### Working:

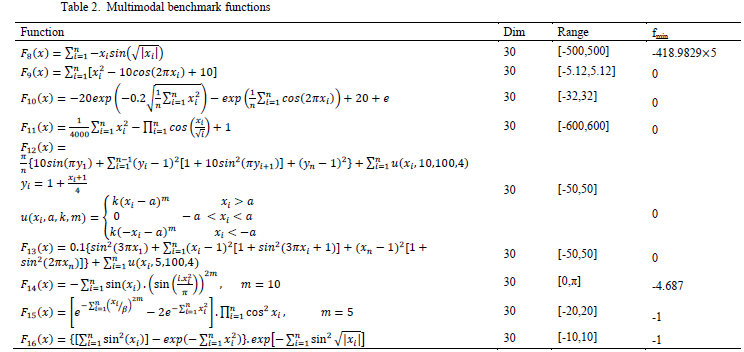
In this algorithm, each solution is treated as wolf and the highest fitness wolf is considered as “alpha”, second highest fit wolf is known as “beta”, third one is called “delta” and all the others are known as “omega”. All wolves follow the alpha wolf and they get the global best solution which is called “prey”[8]



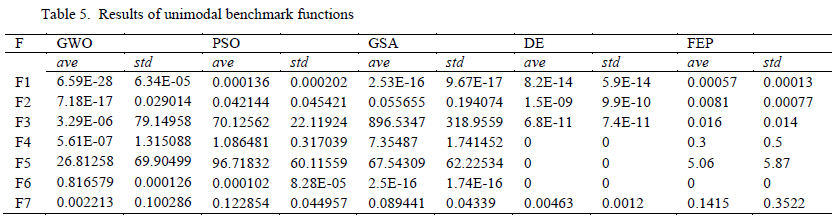
### Experimental Results:

This algorithm is tested on following benchmark functions:

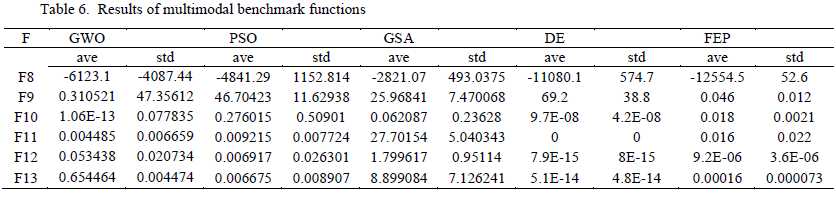




### Unimodal functions:



### Multimodal Functions:



## Soccer League Optimization:

### Inspiration and working:

**Soccer League Competition Algorithm** is also an optimization algorithm inspired by the optimization of football league competitions. All teams play 2 matches with other respective team. Total matches depend upon the total number of teams competing in the tournament by (M\*(M-1))/2 (where M is the total no of teams). Each team wants to top the table at the end of each iteration. Teams which consists high performed or high fitness players has more probability to win matches against opponent teams. The team fitness is calculated by the average total fitness of the players. Each team has 11 fixed players and 11 substitute players. Every team has a SP (Star Player) and the tournament has an SSP (super star player) which has best fitness among team and best fitness among the whole tournament players respectively. The winning and losing team applying different strategies to perform better in next matches. Winning team fixed players try to imitate SP (star player) of the team and SSP (super star player) of the team. Substitutes of the winning team tries to improve their performance by making their fitness at least at the average of fixed players of the team. On the other hand, fixed players of losing team tries to improve their performance by changing position of players. The losing team substitutes pairs are being entered by a certain probability to make winning probability chances. At the end of the tournament, best teams buy players with best fitness and average and weak players are bought by weak team. SSP is the optimal and SP is the local optima of the solution[9].

## Social Evolution:

### Inspiration and Working:

**Social Evolution Algorithm** is inspired by human’s interactions and beliefs. The individuals interact and share information to its neighbor. This Algorithm have three phases: initialization phase, evaluation phase and interaction phase. In this Algorithm, von Neumann Neighborhood architecture is adopted for building the neighborhood. Each Individual’s fitness and probability is calculated. The individuals Evaluate the neighbor based on co-operation factor (controlled parameter) and ability and productivity of the neighbor, then interact with the identified neighbor. The individual will not interact with any random solution in the society instead, they may interact more with the random neighbor in the von Neumann neighborhood architecture because of affinity and trust worthiness, but they are free to explore the society based on NCF (controlled parameter). Also, once the individual is selected for the interaction, the individual solution interacts with the selected individual for all the dimensions of the problem. Once the interaction is performed, individual evaluate the quality of interaction (QI). If the quality of interaction is inferior, interaction’s indecisive factor IDF is evaluated to decide on the interaction as negative or indecisive. All the indecisive interactions will undergo a second opinion process. In the second opinion process, the individual can consult an expert either from the neighborhood or from the society or a non-existing individual with the average capabilities to further evaluate the indecisive interaction before adopting the change to emerge and evolve. After the interaction phase, evaluate the fitness of the updated solutions and compare with the respective original solution to consider the best for next generation. Before the above process is repeated until a termination condition (maximum cycle number), calculate the probabilities of the individuals, average solution the best in the society for the next generation. [10]

## Simulated Annealing:

### Working and inspiration:

**Simulated Annealing** is also well known meta-heuristic algorithm. It mimics the annealing process in material processing when a metal cools and freezes into a crystalline state with minimum energy. The annealing process involves the careful control of temperature and its cooling schedule. [11]

## Particle Swarm Optimization:

### Working and Inspiration:

**Particle Swarm Optimization** is Swam based intelligence algorithm which is inspired by swarm of fish, and birds. In this algorithm, every solution is treated as particle and there runs a swarm and all the particles moves around the global best solution. This algorithm is used in: multimodal optimization problems, production scheduling, power system operations, cryptarithmetics and many more. [12]

## Related Work:

Following are the lists of some optimization Algorithms:

|  |  |
| --- | --- |
| **Algorithm** | **Inspiration** |
| Whale Optimization | This algorithm is inspired by Whale hunting. |
| Dragonfly Optimization | A novel swarm intelligence optimization technique is proposed called dragonfly algorithm (DA). The main inspiration of the DA algorithm originates from the static and dynamic swarming behavior of dragonflies in nature. Two essential phases of optimization, exploration and exploitation, are designed by modelling the social interaction of dragonflies in navigating, searching for foods, and avoiding enemies when swarming dynamically or statistically. The paper also considers the proposal of binary and multi-objective versions of DA called binary DA (BDA) and multi-objective DA (MODA), respectively |
| Centipede Optimization | This algorithm is inspired by centipede’s hunting behavior |
| Bat Optimization | This algorithm is inspired by bat’s echolocation |
| Ant Colony Optimization | Ant colony optimization (ACO) takes inspiration from the foraging behavior of some ant species. These ants deposit pheromone on the ground in order to mark some favorable path that should be followed by other members of the colony. Ant colony optimization exploits a similar mechanism for solving optimization problems. |
| Artificial Bee Colony Optimization | This algorithm is inspired by Bees |
| Water flow optimizer | The WWO takes inspiration from shallow water wave models for solving optimization problems. Without losing generality, suppose we have a maximization problem with objective function f. In WWO, the solution space X is analogous to the seabed area, and the fitness of a point is measured inversely by its seabed depth: the shorter the distance to the still water level, the higher the fitness is. It should be noted that by analogy the 3-D space of the seabed is generalized to an n-dimensional space. |
| Gravitation Field Algorithm | Gravitation field algorithm (GFA) is a novel optimization algorithm derived from the Solar Nebular Disk Model (SNDM) in astronomy, based on the formation of planets, in recent years. In this research, an improved GFA with Optimal Detection (GFA-OD) is proposed for unconstrained optimization problems. Optimal Detection can efficiently locate the space that more likely contains the optimal solution(s) by initializing part of dust population randomly in the search space of a given problem, and then improves the accuracy of solutions. |

# References

|  |  |
| --- | --- |
| [1] | MATEJ Cˇ REPINSˇ EK, SHIH-HSI LIU, MARJAN MERNIK, "Exploration and Exploitation in Evolutionary Algorithms: A Survey," p. 33. |
| [2] | P. V. (Sundar) Balakrishnan and Varghese S. Jacob, "Genetic Algorithms for Product Design," *INFORMS,* 1996. |
| [3] | "The Shodor Education Foundation, Inc.," [Online]. Available: https://www.shodor.org/chemviz/optimization/teachers/background.html. |
| [4] | "Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Automotive\_design. |
| [5] | P. a. K. H. Kán, "Automated interior design using a genetic algorithm," pp. 1-10, 2017. |
| [6] | Seohyun Jeon ; Jaeyeon Lee, "Vehicle routing problem with pickup and delivery of multiple robots for hospital logistics," *IEEE,* 2016. |
| [7] | Nazmul SiddiqueHojjat Adelib, "Applications of gravitational search algorithm in engineering," *Journal of Civil Engineering and Management 22(8):981-990,* 2016. |
| [8] | Seyedali Mirjalili a,⇑, Seyed Mohammad Mirjalili b, Andrew Lewis a, "Grey Wolf Optimizer," *Advances in Engineering Software,* p. 46–61, 2014. |
| [9] | N. Moosavian, "Soccer League Competition Algorithm," *International Journal of Intelligence Sciences,* 2014. |
| [10] | R. S. P. a. Gursaran, "Social Evolution: An Evolutionary Algorithm," *Advances in Intelligent Systems,* 2014. |
| [11] | S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi, "Optimization by Simulated Annealing," *Science,* vol. 220, no. 4598, pp. 671-681, 1983. |
| [12] | M. L. Settles, "An Introduction to Particle Swarm Optimization," 2005. |
| [13] | H. A. Nazmul SIDDIQUEa, "APPLICATIONS OF GRAVITATIONAL SEARCH ALGORITHM IN," *JOURNAL OF CIVIL ENGINEERING AND MANAGEMENT,* p. 10, 2016. |
| [14] | "Science Direct," [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2095263512000490. |
| [15] |  |

1. Objective Function is basically a function that describes the fitness of each sample of population which needs to be either maximized or minimized. Every problem on which metaheuristic algorithms are applied, requires objective function which is minimized of maximized according to the need. [↑](#footnote-ref-1)