International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor:6.887
Volume 5 Issue IX, September 2017- Available at www.ijraset.com

Extending Single Objective Optimization to Multi Objective Optimization in Nature Inspired Algorithms – A Review

Sunita Sharma [1], V.K. Panchal [2]
[1]*Research Scholar, Mewar University, Rajasthan, India*
[2]*Founder President, Computational Intelligence Research Group, New Delhi, India*

Abstract: ***Multi objective optimization problems are very common in real life where we have to optimize many simultaneously conflicting objectives. Due to increasing complexity of these algorithms in real life, here these problems are tackled by nature inspired algorithms. Nature inspired algorithms are the algorithms which are developed by taking inspiration from nature, a number of algorithms are proposed in this regard like ant colony optimization, firefly algorithms, evolutionary algorithms etc. In this paper, we are going to review the extension of some of the famous nature inspired algorithms from single objective optimization to multi objective optimization problems.***

Keywords: Multi-Objective Optimization, Single Objective Optimization, Nature Inspired Algorithm, Ant Colony Optimization, Firefly Algorithm, Bat Algorithm, Cuckoo Search

I. INTRODUCTION

Multi objective optimization problems [1] contain many objectives to be optimized simultaneously which is mostly conflicting [2] i.e. improving one could be done only at the cost of other. For these problems there do not exist a unique solution but a set of solutions which is known as Pareto optimal solution. Mathematically it could be represented as:

Min/Max F(x)=[f1(x),f2(x),……..,fk(x)] i=1,2.....M (1)

Subject to gj(x) >=0 j=1,2,.....j

Hk(x)=0 k=1,2,.....,k

F(x)=[f1(x), f2(x).....fk(x)] is the set of k objective functions to be optimized and

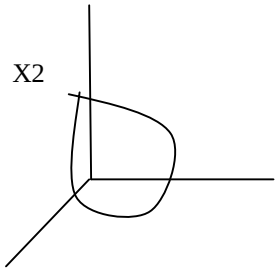
g1(x)>=0,g2(x)>=0,.....,gj(x) are the set of j inequality constraints.

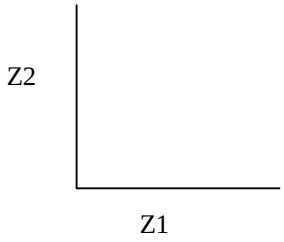
H1(x),h2(x),.....,hk(x) are the k equality constraints.

X=(x1,x2,x3,….,xn) Є S, where S is the feasible region.

F[x) is an objective vector with k>=2, the k objective functions to be optimized can be either maximization or minimization or both. Most of the algorithms are developed to solve only minimization problems and other problems can be converted to minimization type by using duality principle i.e. by multiplying it by -1, similarly the j inequality constraints are treated mostly as 'greater than equal to type' can be converted to less than or equal to by multiplying by -1.
X=(x1,x2,x3,….,xn) is a decision vector The space in which the feasible set S is a subset is called decision space and X is a decision vector in decision space and if X Є S then X is called a feasible solution. The space from which objective vector is a sub set is is called objective space.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor:6.887
Volume 5 Issue IX, September 2017- Available at www.ijraset.com





Let x Є IR^3 i.e. decision vector x contains 3 decision variables

zЄ IR^2 i.e. bjective vector z have 2 objective functions to be optimized.

Now since the objective functions to be optimized are mostly conflicting with each other the optimized solution of one objective function may not be the optimized solution of other so there does not exists a unique solution which simultaneously optimize all the objective functions therefore we have to find a trade of or a compromising solution which optimize each one the objective functions as close as possible to the optimized value such a set of solutions is known as Pareto Optimal set. Pareto optimal set contains all the solutions which are non-dominated with respect to each other i.e. by moving from one Pareto solution [3] to other there is a certain amount of compromise in one objective if we want to achieve a certain amount of gain in other.

Let us assume that the set of non dominated solution i.e. Pareto set is represented in the decision space X and there exits an function f:X -> Y which evaluates quality of the each of the solution by assigning it a value. The set of these values is called objective space. Let (x1,x2,x3,...,xn) be one of the decision vector in the decision space X and (y1,y2,y3,....,yn) be objective vector in the objective space.

In case of single objective optimization problem a solution x1 < X is said to dominate x2< X if y1>y2 where y1=f(x1) and y2=f(x2). But in case of multi objective optimization problem comparison of two decision vector is more complex. So the concept of Pareto dominance is used where an objective vector y1 is said to dominate another vector y2 if no component of y1 is smaller than corresponding component of y2 and at least one component is greater, so we can say that vector x1 is better that vector x2 or x1 dominates x2 if f(x1) dominates f(x2), so there does not exit a unique solution to these objective functions but set a of solution exists representing different trade off between the solutions such a set is known as Pareto optimal set.

Set of optimal solutions in the decision space X is known is Pareto Set X^* and its image Y^* in the objective space Y is known as Pareto front i.e. $Y^*=f(X^*)$. Knowledge of such a vector helps the decision maker to choose the solutions which suits best to the situation under consideration.

In this paper, we have presented the different concepts that have been extended from single objective to multiple objective optimization problems. This section has well explained about the multi objective optimization problems. Further, section 2 presents the different available approaches to solve the multi-objective problems. Section 3 presents the nature inspired concepts that have been extended from single objective to multi-objective algorithms. Section 4 presents the performance analysis and Section 5 concludes the paper.

II. AVAILABLE METHODS TO SOLVE MULTI-OBJECTIVES PROBLEMS

There are different approaches to solve multi objective optimization problem:

A. Classical Approach

1) Weighted Sum Approach:

In this approach all the multiple objectives are summed up to a single objective by assigning weights to each of the objective and then solving the single objective optimization problems [4].

$$F(x)=\sum w_i x_i$$

The optimization of single objective functions thus optimizes the multi objectives summed up. Weights could be assigned by knowing the relative importance or priorities of the objectives. And the outcome of the problem depends upon the chosen weights

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor:6.887
Volume 5 Issue IX, September 2017- Available at www.ijraset.com

2) €-constraints Approach:

In this approach one objective is chosen to be optimized and other are converted to constrains by limiting the value of each of them within a certain pre defined limits. In this approach too multi objective optimization problem is converted to single objective optimization problem and the solution of the problem depends upon choice the function to used for the optimization and constrained values taken for rest of the objectives.

3) Weighted Metric Approach:

It creates an metric I from all the objectives by assigning weights and minimizing the metric obtained.

4) Goal Programming:

Here the deviation of the objective from a target is set and we aim to minimize the weighted sum of the deviations and then solving the single objective problems thus obtained.

B. Pareto Dominance Approach

Here complete set of Pareto dominate solution [5] is found which produces all the trade off solutions without the prior knowledge problem domain. The knowledge of such a set of trade off solutions in turn helps the designer to compare andchoosethe most suitable solution for the problem in hand.

1) Issue in Pareto Dominance Approach:

Generating the Pareto Set [6] for multi objective Optimization problem is computationally expensive and sometimes infeasible because of the complexity of the underlying domain and conflicting nature of the objectives so a number of algorithms have been proposed to find Pareto set like Evolutionary, Ant Colony, Particle Swarms optimizations etc. Each one of them does not guarantee to find the optimal trade off but try to find a good approximation of the Pareto set i.e. the solutions whose objective vector is no too far from the optimal objective vector (true Pareto Set).

Also the goal of approximating a Pareto set is itself multi objective i.e. we have to consider the following objectives while developing Pareto Set Approximation:

a) Maximizing the number of solutions in the Pareto set approximation so that a good number of trade off solutions could be generated.
b) Minimizing the distance the generated Pareto Set approximation from the true Pareto set.
c) Maximizing the diversity in the generated Pareto set approximation.

2) Relation between Pareto Set: The quality of a Pareto set cannot be described in terms of a numeric criterion. Let A and B be the Pareto set obtained by two optimization algorithm. Then we can say A is better than B if any objective vector if B is dominated by or equal to at least one objective vector in A. In that case we say A>B.

But if neither A>B nor B>A, we say both the sets are incomparable in terms of Pareto Dominance. So by using Pareto Dominance we can't say which set is better or preferable from the two. So how to choose a set which are incomparable in terms of Pareto dominance is still an open question in multi objective. So several different criteria are used to choose the set such as distance from the true Pareto set, diversity of solutions etc.

III. NATURE INSPIRED ALGORITHMS

These algorithms are inspired from the nature and apply nature like processes to solutions [7]. The algorithms can be classified in two categories:

1) Evolutionary Algorithm: Genetic Differential Evolution
2) Swarm Algorithm: Ant Colony, PSO

A. Cuckoo Search

This algorithm [8] is based on the behaviour of bird cuckoo described below:

1) Single Objective Cuckoo Search
a) Take objective function(x)
b) Initialize n nests with one egg (each egg represents one solution)

c) Repeat

Take a cuckoo x_i randomly by levy flight.
Evaluate its fitness $f(x_i)$
Choose a nest, say j from n, evaluate its fitness $f(x_j)$
If $f(x_i)>f(x_j)$
Replace j by new solution i
End
Sort all the nests according to fitness.
Abandon a fraction pa of worst nests and generate new nest
Again sort the solutions and find current best

d) End repeat

The best nest (containing the best solution) will be the best solutions after a number of iterations.
In the single objective Cuckoo Search the following rules are used:
Each cuckoo lay one egg at a time, the egg represents the solution to the Single Objective Problem.
The cuckoo dumps the egg in one the nest chosen randomly after comparing the quality of the egg with the egg already present in the nest i.e. comparing the value of the objective functions to be optimized with the new solution to the already existing solution.
The number of hosts nests are fixed each containing one egg, the host bird can discover the alien egg with the probability pa after which it throw the egg away and abandon the nest altogether.
The best nests with the high quality of eggs will carry over to the next generation
Here, Step 2 represents the selection phase in nature inspired algorithm where the best of new and present solution is chosen.
Step 3 represents the mutation so that worst solutions are discarded and new solutions are generated. Also the generations of new solutions helps to explore the search space so that the algorithm is not confined to the limited space.
Step 4 represents the elitism so that the best solutions of the present generation are passed to the next generation so that algorithm converges to the optimal solution properly.
Also the generations of new cuckoo through Levy Flight helps to explore the search space in more efficient way as compared to simple randomization.

B. Extending Cuckoo Search from Single Objective to Multi Objective Optimization Problem

In multi objective Cuckoo Search the above rules are modified [9] as mentioned:
Here each cuckoo lays K eggs at a time, egg k represents the solution of kth objective.
The cuckoo dumps the egg in one the nest chosen randomly after comparing the quality of the egg with the egg already present in the nest here the comparison is done according to the dominance rule i.e. if new solution dominates the existing solutions then the existing solutions are discarded.
The number of hosts nests are fixed each containing k egg, the host bird can discover the alien egg with the probability pa after which it throw the egg away and abandon the nest altogether.
The nests with the high quality of eggs i.e. with non dominated solutions will carry over to the next generation

C. Multi objective Cuckoo Search [10]

1) Take objective function f1(x), f2(x).....fk(x) to be optimized.
2) Initialize the initial population of n nests N1, N2,... , Nn each with k eggs(each egg is a solution of 1 objective function)
3) While (termination)

Get a cuckoo by Levy Flight, say x_i
Evaluate and check if it is Pareto Optimal
Choose a nest say j among n randomly, Evaluate its k solutions
If new solution of x_i dominates those of nest x_j
Replace nest j by nest i
End
Abandon a fraction pa of worst nests and generate new nest
Keep the nest with non dominated solutions and find current best

End while

D. *Bat Algorithm*

1) Single objective Bat Algorithm [11]

Take objective function f(x) to be optimized.
Initialize Bat population of n bats randomly with position x1,x2,.....,xn and velocity v1,v2,.....,vn respectively. Initialize pulse rate ri and amplitude Ai.
While (t<Maximum Number of Generations)
Generate new solution by adjusting Frequencies, Updating velocities and positions.
If rand>ri
Select a solution among best solutions
Generate a local solution around selected best solution
End If
Generate new solution by flying randomly
If rand <Ai and f(xi)<f(xi*) then
Accept new solution
Increase ri and reduce Ai
End if
Rank the bats and find the current best.
End while

2) Multi Objective Bat Optimization Algorithm [12]

In this algorithm classical weighted sum approach to combine all objectives f_k into a single objective

$F(x)=\sum w_i f_i \sum w_i=1$

Here weights are generated randomly from a uniform distribution; it is possible to vary the weights with sufficient diversity so that the Pareto Front can be approximated correctly.

Algorithm

Take objective functions f1(x),f2(x),.....fk(x)
Initialize the bat population xi and vi.
For j=1 to N (points on the Pareto Fronts)
Generate K weights wk>=0 so that ∑wk=1
Form a single objective f=∑wkfk
While (t<Maximum Number of Generations)
Generate new solution by adjusting Frequencies, Updating velocities and positions.
If rand>ri
Select a solution among best solutions
Generate a local solution around selected best solution
End If
Generate new solution by flying randomly
If rand <Ai and f(xi)<f(xi*) then
Accept new solution
Increase ri and reduce Ai
End if
Rank the bats and find the current best
End while
Record x* as a non dominated solution
End for

E. Firefly Algorithm

1) Single Objective Fire Fly Algorithm [13]

Take Objective function f(x)
Randomly spread n fireflies.
while (t<maximum number of Generations)
For i=1 To n
For j=1 to n
if (f(xi)<f(xj))
Move Xi towards Xj
End if
End for
End for
Rank Firefly and find current Best
End while

In this algorithm we initially take n fireflies randomly using uniform random distribution which corresponds to n points in the search space distributed uniformly. Then we compare brightness of the fire fly with each other one by one. Here brightness of the firefly is simply the value of the objective function to be maximized. The lesser bright fire fly will be attracted and thus move towards the brighter fire fly. The process is repeated to certain number of iterations and in each iteration the position of each of the fire fly is updated. The best positioned fire fly is the optimized solution of the objective function after a certain number of iterations.

2) Multi Objective Fire Fly Algorithm [14]

In this both the approached are used to extend Single Objective Fire fly to multi objective one. The classical approach where all the objectives are combined with weighted sum is used by Apostolopous and Vlachos and the single objective problem thus obtained is solved by Single Objective Bat Algorithm as explained above.

Other approach is to obtain the Pareto Optimal Front directly. In this first fireflies are distributed randomly in the search space using uniform distribution. The Iteration starts by comparing the brightness of each of the firefly for all the objectives. If some firefly dominates the other in Pareto front move the other firefly towards it. But if no non dominated solutions can be found i.e. no point in Pareto front than generate random weights wk and find the best solution g* among all the fireflies which minimizes $u(x)==\sum w_i f_i$. Then random walk around g*. Pass the non dominated solution to the next iterations.

Initialize Objective functions f1(x),f2(x),.....fn(x)
Initialize the generations of b fire flies.
While (t<max number of Generations)
For i=1 to n
For j=1 to n
Evaluate the approximation PFi and PFj to the Pareto front
If PFj dominates PFi then
Move firelyi towards fire fly j
End if
If non Firefly dominates other
Generate random weights wk and obtain objective function $U(x)=\sum w_i f_i$
Find the best solution g* among all fire flies which minimizes U(x)
Random walk around g*
End if
Update and pass the non dominated solutions to next iteration
End for
End for
Sort and find current best solution
t=t+1
End while

D. Ant Colony Optimization

3) Single Objective Ant Colony Optimization [15]

Initialize pheromone trails and parameters
Generate population of m solutions (ants)
For each individual ant k<m calculate its fitness f(k)
For each of the ants determine its best position
Determine the best Global ant.
Update Pheromone trail
Check if Termination is true

There are different approaches adopted for extending Single Objective Ant Colony Optimization to Multi Objective ant colony optimization [16].

a) Using Single Pheromone

The initial approach was to use a single pheromone matrix for one objective only. The solution is obtained only for this objective as if this is the only objective to be optimized. The solutions thus obtained are relatively good only for this objective. So this approach is not appropriate without the prior knowledge of the relative importance of the objectives.

A separate colony for each objective function. Each of the individual objectives is optimized using its colony of ant using single objective ant colony optimization method. The common best so far solution is used to update the pheromone information in all the colonies.

In other method heuristic information is constructed by taking all the objectives into consideration but to update pheromone only the most important objective is considered.

In another method the order is imposed on the colonies corresponding to the order of importance of the multiple objectives. In each generation each ant from a certain colony receives a partial from previous colony and tries to complete the partial solution with respect to the objective associated with that colony, when last colony completes the solution combined non dominated solution are used to update the pheromone information.

None of the above method can be applied when objectives can't be ordered.

b). Separate Pheromone for each Objective:

We consider multiple colonies of ant. For each if the colony one pheromone for each objective and a set of weights. Each ant for each colony uses a different weight but same for each iteration. After each iteration non dominated solutions are used to update pheromone. Each colony focus on approximating a certain region of Optimal Pareto Set. Collaboration between the colonies is obtained by using:

Solution of other colony to detect the dominated one.
By using non dominated solution from other colonies to update pheromone.

IV. PERFORMANCE ANALYSIS

The performance of each of the multi objective optimization algorithm is different. The performance of Multi Objective Optimization algorithm depends on

Single objective optimization upon which it is based.
Approach to solve Multi Objective Optimization problem: Classical or Pareto Dominance.
Classical approaches have some problems:
Only one Pareto optimal solution is obtained in one simulation run of the algorithms.
Classical approach requires some knowledge of the problem domain like the priorities in weighted sum approach or the limiting values of constraints in €-constraints approach etc.
Some algorithms are sensitive to the shape of Pareto front.
The spread of Pareto optimal solutions depends upon the efficiency of the single objective optimizer.
Pareto Dominance Approach:

Generating the Pareto Set for multi objective Optimization problem is computationally expensive and sometimes infeasible because of the complexity of the underlying domain and conflicting nature of the objectives.

Also the goal of approximating a Pareto set is itself multi objective i.e. we have to consider the following objectives while developing Pareto Set Approximation:

Maximizing the number of solutions in the Pareto set approximation so that a good number of trade off solutions could be generated.

Minimizing the distance the generated Pareto Set approximation from the true Pareto set.

Maximizing the diversity in the generated Pareto set approximation.

We can also see that there are different nature inspired single objective optimization algorithms which have been extended to multi objective optimization algorithms with different approaches as mentioned in table 1.

TABLE I
APPROACH TO CONVERT INTO MULTI-OBJECTIVE ALGORITHM

Algorithm	Approach to convert into Multi-objective Algorithm
Cuckoo Search	Pareto Dominance
Bat Algorithm	Classical Weighted Sum Approach
Firefly Algorithm	Both Classical & Pareto Dominance
Ant Colony Optimization	Pareto Dominance

V. CONCLUSION

This paper presents the concepts from single objective to multi-objective functions to solve multi-objective problems. Due to complexity of multi-objective problems, nature inspired concepts have been used. Classical or Pareto Dominance methods are the basic approaches to solve multi-objective problems. Further, these basic concepts are introduced with nature inspired single objective problems to convert them into multi-objective problems as discussed in performance analysis section. So, here nature inspired algorithms of Cuckoo Search, Bat Algorithm, Firefly algorithm and Ant Colony Optimization have been converted from single objective to multi-objective using basic concepts of Pareto Dominance, Classical Weighted Sum Approach, Both Classical & Pareto Dominance and Pareto Dominance approach.

REFERENCES

[1] Marler, R. Timothy, and Jasbir S. Arora. "Survey of multi-objective optimization methods for engineering." *Structural and multidisciplinary optimization* 26, no. 6 (2004): 369-395.
[2] Ulungu, Ekunda Lukata, and Jacques Teghem. "Multi□objective combinatorial optimization problems: A survey." *Journal of Multi□Criteria Decision Analysis*3, no. 2 (1994): 83-104.
[3] Abbass, Hussein A., Ruhul Sarker, and Charles Newton. "PDE: a Pareto-frontier differential evolution approach for multi-objective optimization problems." In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, vol. 2, pp. 971-978. IEEE, 2001.
[4] Marler, R. Timothy, and Jasbir S. Arora. "The weighted sum method for multi-objective optimization: new insights." *Structural and multidisciplinary optimization* 41, no. 6 (2010): 853-862.
[5] Zitzler, Eckart, and Lothar Thiele. "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach." *IEEE transactions on Evolutionary Computation* 3, no. 4 (1999): 257-271.
[6] Voorneveld, Mark. "Characterization of Pareto dominance." *Operations Research Letters* 31, no. 1 (2003): 7-11.
[7] Zang, Hongnian, Shujun Zhang, and Kevin Hapeshi. "A review of nature-inspired algorithms." *Journal of Bionic Engineering* 7 (2010): S232-S237.
[8] Yang, Xin-She, and Suash Deb. "Cuckoo search via Lévy flights." In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*, pp. 210-214. IEEE, 2009.
[9] Lidberg, Simon. "Evolving cuckoo search: From single-objective to multi-objective." (2011).
[10] Yang, Xin-She, and Suash Deb. "Multiobjective cuckoo search for design optimization." *Computers & Operations Research* 40, no. 6 (2013): 1616-1624.
[11] Yang, Xin-She. "A new metaheuristic bat-inspired algorithm." *Nature inspired cooperative strategies for optimization (NICSO 2010)* (2010): 65-74.
[12] Yang, Xin-She. "Bat algorithm for multi-objective optimisation." *International Journal of Bio-Inspired Computation* 3, no. 5 (2011): 267-274.
[13] Yang, Xin-She. "Firefly algorithm, Levy flights and global optimization." *Research and development in intelligent systems XXVI* (2010): 209-218.
[14] Yang, Xin-She. "Multiobjective firefly algorithm for continuous optimization." *Engineering with Computers* 29, no. 2 (2013): 175-184.
[15] Dorigo, Marco, Mauro Birattari, and Thomas Stutzle. "Ant colony optimization." *IEEE computational intelligence magazine* 1, no. 4 (2006): 28-39.
[16] Doerner, Karl, Walter J. Gutjahr, Richard F. Hartl, Christine Strauss, and Christian Stummer. "Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection." *Annals of operations research*131, no. 1 (2004): 79-99.