J. Appl. Environ. Biol. Sci., 6(8)98-111, 2016

ISSN: 2090-4274
Journal of Applied Environmental and Biological Sciences
www.textroad.com

Exploring and Analyzing Evolutionary Optimization in Different Environments

Arslan Ellahi and Waseem Shahzad*

National University of Computer and Emerging Sciences, Islamabad, Pakistan

Received: March 31, 2016
Accepted: June 14, 2016

ABSTRACT

Pervasive and ubiquitous real world environments are generally complex and multifarious, they restrains perfection as well as imperfection in them and keeps on changing with the passage of time. Focusing on evolutionary algorithms, we are fascinated to solve complex optimization problems in dynamic as well as imperfect environment. Evolutionary optimization in different environment has magnetized numerous researchers in last two decades and it becomes a hottest research field in soft computing. In this paper we carry out a detail survey of the state of art and latest techniques related to evolutionary optimization problems both in real and theoretical world. We discuss the working of different optimization techniques in detail, their strengths and weaknesses, problems with them, loop holes inside them and research challenges in those optimization problems. Along with this, we also discuss dynamic and imperfect environment in detail and improvements that can be integrated in different scenarios to enhance performance for solving optimization problems. Along with this, we also discuss dynamic and imperfect environment in detail and improvements that can be integrated into different scenarios to enhance performance for solving optimization problems.

KEYWORDS: Dynamic optimization, imperfect environment, soft computing, evolutionary algorithms and complex optimization

1.INTRODUCTION

To embark upon the complex optimization problems, evolutionary algorithms play a vital role in solving them up to their utmost capacity [1-5]. These bio-inspired algorithms evolved from the idea of Darwinian evolution, ubiquitous researchers are working and trying to find the optimal solutions of multi-objective problems, real world problems are hardly static and simple, therefore having dynamic nature dynamic environment problems are obviously complex. This dynamicity in the environment affects the objective function, instance of the problem and environmental constraints that transpire in the battle field of those problems [4], [6], [7]. Finding optima and movement due to arrival of change is goal in optimization problems [8]. Evolutionary algorithms are very successful in real world problems, but there always exists challenges to cater on, nothing is ultimate in the world. There are many parameters of these algorithms and sometimes they become problematic in solving multi-objective optimization problems [9].

In the stated paper, our contribution is to explore the insights of the work that have been done in the different environments. We are interested to find the conditional search that includes coverage of every element inside the environments. A lot of amount of work has been done on the field of dynamic environment, many algorithms and technique have been developed in this environment and research is still going on. We have also investigated the imperfect environment scenario in our survey. Our major focus is covering different aspects of optimization in dynamic environments that includes the benchmark problems and solvers, performance of EA's in different environments, approaches of different algorithms, applications in real world, strengths and weaknesses of different algorithms and their approaches along with theoretical investigation of different algorithms. Our purpose is to review about the questions, that how present approaches are working, we have given a detailed explanation for that, secondly their assumptions, strengths and weaknesses, gaps in the present techniques along with the challenges and contributions that can be made in future research in evolutionary algorithms in different environments.

Optimization problems in today's world occur in all vicinities of industry, research and management. To exemplify the word optimization, we need to take help from the real life problem. An example can be utilization of present resources in optimized fashion, maintaining the cost as low and enhancing the quality as high. In almost every business, construction and in technical project there is a need of optimization in from every perspective. In the context of evolutionary algorithms, there exists large number of optimization problems. Most of the problems are NP-hard problems, for those no solutions is possible in polynomial time, scheduling problems, optimization using ant colony algorithm on non ideal iris image segmentation, routing problems and travelling sales person problems are common in them [10], [14], [19] [21].

Shahzadeh et al [20] proposed an improved grouping genetic algorithm in which problems related to clustering algorithm are made to resolve with the improved model than the previously proposed models. In this model an improved algorithm for clustering is presented and tested in which the results showed are having good performance that reaches the optimal solution. The results showed in it are much better than previous ones.

Corresponding Author: Waseem Shahzad, National University of Computer and Emerging Sciences. Islamabad, Pakistan (waseem.shahzad@nu.edu.pk)

2. Evolutionary Algorithms

Evolutionary algorithms are basically populations based meta-heuristic that are population based consists of multiple solutions rather than a single solution. Nature is composed of many biological experiences each of which has different running criteria. Idea of biological algorithms came up from the Darwin evolutionary systems of animals and plants. Evolutionary algorithms are inspired from nature and environmental behavior [6], [7]. Evolutionary algorithms are mostly applied on global optimization problems and also in function optimization of large dimensional data. Inspired by biological evolutionary methods, it focuses on population of species. Evolutionary algorithms are inspired from the umbrella of soft computing evolutionary algorithms mostly handle non-linear problems that show adaptive behavior. They have high quality of learning with optimization problems. Robustness is the key metrics by which they are known to fast and explicit optimized algorithms. Convergence varies from problem to problem but they are highly effective and their predictive accuracy is high. Strategy building and strategy optimization are the core focus on which the complete evolutionary process runs. Most of the evolutionary algorithms can be described as some population at time t, variation operators and selection operators [20].

2.1 Evolutionary Computation

Evolutionary computation is a general term to use in a multiple complex single objective as well as multi objective complex optimization problem they work on the basis of Darwinian natural biological evolution theory. They share the idea of populations their individuals, present populations that use their reproductive operators of crossover and mutation to produce new offspring's. They produce better individuals from the previous populations and new inhabitants produced have better characteristics than previous population. New solutions from the previous ones are produced by applying recombination phenomenon that combines two or greater than two individuals from the population and produce new off springs. The most important thing in evolutionary algorithms is concept of fitness and fitness function. The individuals with high fitness value will get survival in the population's space and the inhabitant or individuals with low fitness will not survive. In the past few years there have been many diverse surveys and applications that have been developed by the researchers mostly in solving the problems related to complex optimization problems, combinatorial optimization problems, multi objective optimizations problems, data mining problems and dynamic and imperfect environmental problems. There are other methods that are self-adaptive those algorithms that have a capability and tendency to control their parameters by themselves are self-adaptive [2].

2.2 Evolutionary Strategies

Evolutionary strategies are one of the techniques proposed in evolutionary algorithms for the optimization of continuous functions. Two steps are included in this regard one include parent in selection and one exclude it. Evolutionary strategies started with mutation only with mostly individual inhabitant. Evolutionary strategies major focus is on behavior of individuals [2].

2.3 Evolutionary Programming

Evolutionary programming is another technique being used in evolutionary algorithms. Major purpose of proposing this technique is to build strategies of optimization for evolutionary games and prediction of sequence; learning is the basic purpose of every evolutionary algorithm. When these algorithms are trained they work on function optimization of different parameters. Evolutionary programming also has a major application in training of artificial neural networks. Crossover operator genetic algorithm is not involved in evolutionary programming this means that it has major focus on behavior of species [2].

2.4 Genetic Programming

Genetic programming is another population based meta heuristic algorithm that work on tree derivative fashion. GP consists of operators and operand with terminal nodes and leaf nodes. The variables and constants in the program are named as terminals in genetic programming. GP works in such a way, first of all population of individuals is initialized, in most of the cases the population is initialized randomly. After the initialization of population in GP, those generations are evolved on the basis of Darwinian evolutions, crossover and mutations, to create next population individuals, the probabilistic selection is taken place, so after that genetic operators starts their working and produce off springs in the next generations. There is multiple type of cross over and mutations that can be applied and each combination of that can result in diverse population's production. GP can be successfully applied in machine learning as a core solver for data mining problems, this works on the phenomenon of survival of the fittest solutions as the best one in its domain.

2.5 Genetic Algorithm

Genetic algorithm is one of the major revolutions in the field of soft computing. It is basically the mostly common used populations based evolutionary algorithm in many problems with many dimensions in them. In basic GA, that is very generic, cannot perform well in imperfect and dynamic environments, this is considered as randomized optimization algorithm, which uses searching techniques in its population space that is broadened across its sides. It is inspired by biological intelligence used to find difficult optimization problems. It consists of solutions as a population; population contains individuals as chromosomes [24].An improved grouping based genetic algorithm with improved model of clustering is proposed in [20]. Results showed on the basis of comparing GGA and IMGGA showed clear difference that grouping genetic algorithm IMGGA performs much better than GGA and in 70% of cases it shows better result [20]. Basic concept is the survival of the fittest chromosome in the population space. Genetic algorithm is composed of two reproduction operator crossover and mutation they are used to create offspring from selected parent. This works on the phenomena same as that of other evolutionary algorithms like the survival of the fittest solution or individual and that is determined by fitness function. The reproduction operators that are used have their own importance, crossover is between the individuals to produce new off springs that also consists of varying types, like single point crossover, double point crossover, uniform crossover, similarly mutation introduces some randomness in the population, but its value should be maintained on a level, so next generation produced cannot het highly differentiable from the previous one. Genetic algorithm has now many variants, researchers are still applying to develop more of its variants; it is used in multi-objective optimization, distributed problems and many other real world problems [21], [22], [23], [24].

3. Dynamic optimization problems

One of the major parts of this paper is to cover the complete aspects of dynamic optimization problems. Substantial amount of work have been done in the field of dynamic optimization. The main purpose in these problems is to locate the optimality of the solution at certain interval of time, and to track that optima in search space, evolutionary computations and swarm intelligence algorithms are very robust and attractive in solving the complex optimization problems this is due to their natural evolving power and learning capability, focusing on the definition of dynamic optimization problems, we are stating it as problems in which the optima from is changing from the begin time to end time interval constantly that cause the landscape to also change in order to locate the new optimal solutions.

Since the research development in this field in not very old, it's just a progress of last 25 years and research is still going on in this field. There are many books, journal issues like, IEEE Transaction on Evolutionary Computations, Soft Computing and many other active journals working on them, along with highly prestigious conferences like IEEE Congress on Evolutionary Computations. There are hundreds of PhD theses in this field. Substantial amount of work have been done, including technical developments, literature studies and surveys in the field of static and dynamic environments, some of the work in this division held in synthetic dynamic problems, like function optimization, like Uludug et al [8] had proposed a population framework that involves offline as well as online learning, a comprehensive survey has been made by Crus et al [9] about the optimizing dynamic problems, problems in them and the performance measurements in all those, Eriksson and Simoes [55] works on optimization using linear regression models, Comelius [10] in his PhD thesis discussed the use of differential evolution in solving dynamic optimization problems, Brank et al [11] proposed a work about the evolutionary optimization in uncertain environments, Montazeri *et al.,*2013 [21] proposed the effect of ant algorithm use on non ideal iris image segmentation, merging strong image segmentation and metaheuristic algorithm by optimizing iris borders, reduce selected features and increase precision, results are obtained through localization of ACO. By comparing the results obtained from system of image processing incorporation with metaheuristic optimization algorithm, lowest number of features, cost and precision was achieved in it. This method has still some of the areas to improve. Amold et al [12], [13] proposed a evolutionary gaming strategy in solving a checkers game, than a noisy fitness function work and work on noisy environments had been proposed by different researchers like Jensen et al [15], Eiben et al [16], Golberg et al [17], Johnston et al [18], Brank et al [27], [30], [31], [32], [33], [35], [36]proposed a scheduling dynamic optimization problems using genetic algorithms, Kluwer et al [29] proposed a work on swarm optimization used in dynamic environments, Brank et al [39] proposed a algorithms in dynamic environments that creates robust solution by making the optimization of fitness function in the solution. There are also some memory based techniques working on the dynamic environments like [41], in [42] an optimized dynamic environment had been proposed, Brank et al [40] had proposed a application in caching route for internet using dynamic optimization strategy, Brank et al [48] proposed a multi-population based approach to solve dynamic optimization problems, Brank [50] proposed a evolutionary algorithm in solving dynamic optimization problem, Zabihinpour *et al.,*2014 [51] proposed a fuzzy based optimization approach to calculate the mean and range of product quality, in this method a fuzzy control charts which are triangular fuzzy numbers are constructed on the basis of fuzzy based approach. It is based on the sample mean which has two advantages that are it maintains process information and that can be used to make process modification that make it in control. Studies showed that the performance of proposed approach is better and still need further research.

On the other hand there have been many works for real world applications like, Akbar et al [54] proposed face recognition using hybrid space features with support vector machine, it is the intelligent model of recognition problems in soft computing to get good features they have used different algorithms as learner classifiers. To judge the performance of proposed model, both transformation and local pattern based techniques are used to extract features from dataset. Various algorithms are used to extract the features, they are assessed using 10 fold cross validation and 92.1% accuracy is achieved using SVM. Bull et al [43] proposed a model framework in evolutionary computation used to solve dynamic optimization problems, Bumham [44] has done a work on design of aerospace, Cantu-Paz [45] has given contribution in evolutionary robotics. Carlisle [46] has proposed a technique for combinatorial optimization problems solved using dynamic optimization, this is a knapsack based technique, Branke [48] has given a technique to solve multi-objective optimization in dynamic environments. Goh [49] has given a technique of self adaptation and mutation in noisy and uncertain environments, Brank [11] has given a survey on uncertain and dynamic environment and given a other state of the art techniques that has been developed yet. Branke [50] proposed a dynamic benchmark function generator to solve combinatorial optimization problems.

Simeo [59] proposed a genetic algorithm to solve complex problems in dynamic environments, Branke [28] proposed a memory based indexing evolutionary algorithm, that uses a concept of memory based dynamic evolutionary algorithm, Branke [29] proposed a associative memory model in a dynamic environment, in which new individuals are created from the information gathered from the environment. In [38] they proposed a adaptive model that formulates evolutionary strategies and adapt them in their environment. Cultural algorithms like in [61] they proposed an introduction of cultural algorithms in dynamic optimization, In [62] proposed a cultural algorithm that works on a rule based system to solve dynamic optimization problems, Rossi et al [28] proposed the used of cultural algorithms in function optimization in dynamic environments, their impact on environmental dynamics, evolving a strategy to adapt the change in an environment, knowledge learning in dynamic environments. Richter et al [56] also used function optimization in dynamic environments and they perform very effective in dynamic environments. There are many other examples of dynamic environments used in multi modal optimization of functions, networks routing optimization, vehicle routing optimization, traveling sales person, market stocks analysis.

Surveys studied for dynamic optimization problems in the past covers the detail aspect of work done in this field and the techniques that have been developed for the study, but the gap always remains there and to fill that gap we have put our contribution in such a way, we have given a detailed review about the existing approaches and why they fail in some scenario and what are the challenges of this field, for this we have given the detailed strengths and weaknesses of optimization problems. We have focused on reviewing many aspects of optimization problems like benchmark problems, performance measures, algorithmic and practical techniques and many other aspects.

4. Benchmark problems

To check the performance and measurement of dynamic environment problems, Branke et al [42] has created moving peak benchmark in order to solve the dynamic optimization problems and to check their performance by setting on different parameters. The benchmark is composed of different moving peaks, so that algorithms can be tested on diverse scenarios and we can check the performance of dynamic environments in multiple environments and parameters. In a multi dimension problem space of the moving peak function has several peaks in it with optima and height of that peak along with its shape [10].

For n number of peaks in n number of dimensions the moving peak benchmark has following parameters in it: that are number of peaks, dimensions, maximum and minimum height and width of peaks, period of change that come in it, severity of change, frequency of change, function and correlation are the parameters of moving peak benchmark. There are three environments in moving peak benchmark problem and by changing parameters in all we can adjust the severity of change and height. Brank et al [39] had suggested all these three scenarios. However most of the researchers use the scenario 2 in their moving peak benchmark as in early they have been used mostly [27-40].

4.1State of the Art Benchmark Techniques

There are many state of the art benchmark problem that have been developed until now, based on the literature we have investigates the characteristics of those benchmark problems, their application in different scenarios. There are large number of benchmark problems used to solve combinatorial optimization problems, multi-modal function optimization, many real life problems and scheduling problems. While seeing those benchmark problems, we categorize them on the basis of their problem fitting scenario. They hold some of the characteristics that are having different classification criteria as time dependencies of algorithm on that benchmark, predictability for that benchmark that the solution it is predicting has better optima or not, visibility of the changes that are made in optimization algorithm, the number of constraints that are effecting that changes over an interval of time, total number of objectives, the types of changes involved, factors influencing on those changes.

Coming towards the benchmark problems that have been done in past are spread in different window like, there are most of the benchmark problems that are non time dependent problems, but there are some that are dependent with time problems like in [64], [65], there are some of the benchmarks in which change is early detected but there are some in which it's not easy like [69], [70]. In many benchmark problems change taken place in objective function but in some of the cases like in [66], [67], [68] constraints also change along with the objective function. The existing state of the art benchmark problems are mostly for the solving the single objective optimization problems but there are some examples like in [75], [76], [77] they are dynamic multi-objective optimization algorithms.

Talking about the contentious benchmark problems for optimization are mostly categorized on the basis of the general working of that benchmark, dependencies with time, detection of changes, changes detection duration, single or multi-objective, types of changes, factors that play in change are the main working judgments for optimization benchmark problems. Those includes switching function [69], moving peak benchmark we have discussed above, oscillating peaks benchmark [66], DF1 [80], [81], Gaussian peak benchmark [82], disjoint landscape benchmark [70], dynamic rotation benchmark [83], dynamic problem generator based on multi objective optimization [75], dynamic multi-objective optimization benchmark set [76], ZJZ in [77] and hybrid evolution in optimization in [77] GDBG [71], dynamic rules to control the steps of change occurrence [81], constrained benchmark set for dynamic optimization [66], [67], [68].The other part is the coverage of combinatorial optimization benchmark problems like dynamic fitness match [84] that uses bit matching function to match the strings, XOR [90], [91] generates binary coded problems used to solve dynamic optimization problems, dynamic DTF [63] is also a combinatorial benchmark for solving optimization problems.

Functions	Changes Predictable?	Changes detection by using few detectors?	Single/Multi Obj?	Changes are cyclic, periodical and recurrent?	Factors that changes			
					Objective functions	Variables Domain	Variables Numbers	Constraint functions
Switching Functions [69]	Mostly No	Yes & no	Single	Yes	Not Mention	No	No	No
Moving Peaks [63]	Mostly No	Yes	Single	Configurable	Yes	No	No	No
Oscillating Peaks [63]	Mostly No	Yes	Single	Yes	No	No	No	No
DF1 [80,81]	No	Yes	Single	No	Yes	No	No	No
Gaussian Peaks [82]	No	Yes	Single	No	Yes	No	No	No
Disjoint Landscapes [70]	Mostly No	Depends on Number of Peaks	Single	Yes	Yes	No	No	No
Dynamic Rotation [83]	Mostly No	Partly Detectable	Single	Yes	Yes	No	No	No
MOO Generator [75]	Yes	Yes	Both	Not	Yes	No	No	No
FDA [76], ZJZ [78], HE [77]	Configurable	Yes	Multi Obj	Yes	Yes	No	No	No
Dynamic Test [84]	Partly	Yes	Single	Yes	Yes	No	No	No
CDOPG XOR [85]	No	Yes	Single	Yes	Yes	No	No	No
CEC 09 [71]	No	Yes	Single	Yes	Yes	No	No	No
G24 Constrained Set [66, 67]	Yes	No	Single	Yes	Yes	No	No	No
Dynamic Constrained [68]	No	No	Single	Partly	Yes	No	No	No

Table I. Benchmark Generators in Continuous Problems

5. Performance Measures in Optimization Problems

Performance measurement is one of the major features for optimization problems, either in dynamic or imperfect evolutionary problems. In this section, we have analyzed the existing techniques and identify the strengths and weaknesses of each measure, also discussed to overcome the existing disadvantages in these techniques, these measures can be grouped in two further sub groups like, based on optimization performance and behaviors in problems. The following two criteria are explained below as follows:

5.1 Performance Measures used in Optimization Problems

One of the measures to in optimization problems is to look upon their optimality and evaluate those algorithms on that basis. Fitness function or objective function can be one of the measures in them. The fittest solution will always be on global optima, distance measure is used to calculate the global optima's. After that performance measures are further sub divided in groups as:

A)Total amount of possible generations in number of runs of any optimization algorithm, there are multiple runs in every algorithm that are necessary to find the optimal performance of curves in algorithms. First of all this measure had been used by [32], [33], [27], [34], [35]. Researchers are still using this measure as one of the most accurate and realistic performance measure in optimization algorithms. This is due to the reason that this measure always shows the best possible curves in the graph that has been drawn on the basis of some algorithmic runs. Performance may vary from algorithms to algorithms in term of number of runs.

B)Performance on the basis of error, errors are mostly offline and online in optimization algorithms [67], [68]. These are performance evaluation on the basis of average errors are calculated on the number of generations. In most of the cases online errors are calculated. Some researchers have made modification in offline and online errors like in [86] to get better results in term of performance in optimization problems.

C)The other type of error is to measure it before the occurrence of change in environments, proposed in [70], these errors are calculated as the difference between the most optimal solution and the best solution and those differences are then averaged to give this error. It is useful in the scenarios, where we want to achieve to error before the occurrence of change in the problem. After that we compare this error with other optimization algorithms. However analyzing this, we drew following disadvantages. First of all it does not compute the current performance of algorithm or does tell about the measurement of current performance by the algorithm, secondly this error calculation does not seems to be normalized on certain scales which can possibly be biased towards the errors with large number. Lastly this error method requires knowing the global optima value whenever the change occurs.

5.2 Performance Measures Based on Behaviors of Algorithm

This is the second performance measure in optimization problems solved under evolutionary umbrella. Every optimization algorithm exhibits certain behaviors under its domain that lead to help us in determining the qualities of algorithm, mainly in evolutionary optimization algorithms that behaviors exists in the form of maintenance of diversity through the possible number of runs, maintaining the fitness of solution. All these measures are very necessary and part of performance evaluation. We are now explaining the main behaviors below:

A)One of the behavior that optimization algorithms show is diversity, diversity measurement in environments are very necessary that deal and behaves like a heat of evolutionary algorithms. In the literature there are many measure exist based on the diversity like entropy proposed in [41], hamming distance proposed in [42], [43], [44] the most important and widely used distance measure, coverage of peaks in graph proposed in [81], and the maximum spread coverage in search space proposed in [46].

The most important diversity measure is the hamming distance that has been used widely in optimization problems, it calculates the distance from every individual in the population to the global optima as proposed in [42], in later research this old measure of diversity calculation has been changed and now it used the nest only individuals in the population to calculate the distance measure like proposed in [43]. Moment of inertia [45] is also statistical measure inspired from the concept of inertia of physics researcher applied this method to measure the diversity in evolutionary algorithms. This proposed inertia is mostly equal to concept of hamming distance measurement search space however this measure moment of inertia has an advantage that it is computationally inexpensive and more effective than a hamming distance measurement. There is another diversity measurement method that is peak cover [77], this algorithm works on the number of peaks to evaluate algorithm behavior and landscapes that require full information about the peaks.

B)The second behavior in optimization algorithms is the fall of performance after the occurrence of change in it. In some of the dynamic optimization algorithms; performance drops whenever a change enters in it. The performance is dropped in a sense of fitness, for hat a method of stability [37] was proposed that evaluates the fitness based on the accuracy between the algorithms. This method gives us the evaluation of the fitness how much fitness is dropped after the entrance of change in the environment.

5.3 Performance Assessment in Multi-Objective Optimization

Many studies had been done in measuring the performance of dynamic MOO problems, in multi objective optimization problems there are multiple criteria's to measure the performance at certain interval of time like proposed in [46], [48], [49]. Accuracy is also an important measure in multi objective optimization problems like in [51] they defined the ratio between the current volume and the maximum volume achieved and the wise versa of it. On the basis of this accuracy measure in multi objective optimization problems we can calculate the stability and performance.

5.4 Some Research Issues/Questions on Optimization Problems

As previously we have discussed the performance measure and behavioral measures in dynamic optimization problems in detail and drill down the factors that are affecting them. There are some research issues and questions about the evolutionary dynamic optimization, it is vague in dynamic optimization problems that to achieve the global optima is the only objective in them, performance measure settled down for optimization problems really exhibits the required measurements or not, there are some works like [33], [17], [43] gave some the measure to judge the performance in optimization problems. In [33] some efforts are made to find out the answers regarding the goal of real world optimization problems and requirements needed for them.

Another research question is related to the optimality measure that are made for optimization problems, most of the dynamic optimization problems focuses on absolute fitness values, and ignores the relative ones, relative fitness values should not be ignored and can be used, may be interesting results seems while comparing them with different algorithms. In [37] there are some accuracy measures that have made an effort to normalize the fitness value on a certain scale. Therefore to have such a normalization factor it needs information about the changes in maximum and minimum period interval.

Thirdly, the behavioral measures used in judging the performance of optimization problems also does not employ that the previous results will correlate with next coming once or not. In [47] it has been shown that stability in the behavioral measure is not only the measure for the quality of the solutions [52]. The main thing is the relationship between the behavioral measures and the performance measures in dynamic optimization problems.

6. Techniques Used for Optimization

In this section, we have discussed the dynamic evolutionary optimization algorithms, state of the art work, their strengths and weaknesses of the all the approaches that have been recently developed.

6.1 Evolutionary Algorithms Goal in Optimization Problems

In the static environment, the main purpose is to locate the optima as soon as possible, but when there comes a change and time variation in the environment and the optima constantly changes its position, then optimization algorithms use their previous learning that have been used in searching of the local optima. There are approaches in dynamic optimization that quickly respond the occurrence of change in environments; we have discussed different dynamic optimization techniques with their strengths and weaknesses.

6.2 Change Detection in Dynamic Optimization Problems

As described above, we have said that in many dynamic environments, change detection is rapid where as in some of them change detection is slow, here we have analyzed the change detection into categories as, detecting change by re-evaluating the solutions and other is detection of change on the basis of behaviors of algorithms.

A)*Re evaluation:* The first methods of change detection is the evaluations of the solutions that are already evaluated, in this technique algorithm is always busy in evaluating those solutions that gave better fitness, so in order to better learn them algorithm re evaluates them to detect the possible change in the environment, there are some current approaches like [53], [54], [55] for obtaining best solutions, other is memory based approaches [56]. Other than the search population based approaches, there can be possibility to adjust the population in a single point like in [57], or the combination of multiple random solutions in the populations like in [58], [59], or a peaks that are found in the solutions like in [60], [61].

B)*Strengths and weaknesses of re evaluation method:* change detection is its self a heavy computational process, by including detectors in it will add more functional computations there should be optimal strength of detectors so that the performance of algorithm can be improved. In most of the existing approaches the used number of change detectors are less in order to avoid extra functional computations, but there are some cases of change in the search spaces like in [66], [67], [85] the less number of change detectors might not work, for this we need extra detectors to guarantee the optimal change detection in the search space environment. For this problem some of the attempts have been made like in [22], [23], [26]. Now the thing that matters is the size of detector and its computational complexity, the main benefit that this re evaluation process get its ensure and guarantee up to absolute detection of change, in [26] it is clearly shown that if the change detection will be difficult then this re evaluation method will help significantly. This process has a disadvantage as well, like in every generation detectors will have to re evaluate in every step that will put up an additional cost every time.

Other disadvantage can be that this re evaluation mechanism might not be suitable with the noisy fitness function problems, noise can de track the algorithm in wrong direction and change detection purpose can be mislead.

6.3 Change Detection Based on the Behaviors of Algorithms

We have previously defined the behaviors of optimization algorithms we are interested to find out the solution in which we can get the maximum fitness of solution on average. In a literature of swarm intelligence like mentioned in [62], change detection is based on the observation of changes. Behaviors shows some extra ordinary properties due to which change in the environment can be detected, as mentioned in [22] change is detected based on the diversity and its relations with the fitness value.

There are some of the advantages of this behavioral model that extra computations can be avoided, but it's a fact that no change detector is used therefore it cannot ensure the detection of change always in it [26]. There may be a possibility that this technique might react unusual than normal and might results in false positives like in [62]. There can be another disadvantage that the change detection in this might be specific to algorithm.

6.4 Initiating Diversity on the Arrival of Change

In dynamic optimization problems, diversity is the heart that needs to be introduced on the arrival of the change. Therefore maintaining it and enhancing it at the certain level is very important.

A)General idea: we have seen that in static environment the main success is to find the optimal point, where as in dynamic optimization, convergence vary with respect to the arrival of change, in dynamic fitness landscapes the optima is always changing its area and location and it is difficult to keep track it, moving slowly from the objective towards the global optima is our main purpose. Increasing the diversity can be one solution for this, but there might be that this solution not ensures us the optimal results. The main study that goes on in variable search in local optima is [63], [64], feature partitioning algorithm is also proposed in [65]. Genetic programming adaptation idea had been proposed in [66] that increase the mutation rate and also reduces the elitism and after change it also increases the probability of crossover. The main idea of diversity introduction had been extensively used in multi-objective optimization problems that can help the population to cater the occurrence of change in its environment. Swarm intelligence also has a taste of introducing diversity after change occurrence. In [53] a randomization methods have been proposed that introduces diversity in change scenario and swarms in the search space will be re-diversified.

B)*Strengths and weaknesses of diversity initiation on the arrival of change:* The methods proposed in this approach focus on maintaining diversity and enhancing it. They have an advantage that they can focus on search process only and have edge on that and whenever a change is detected it reacts towards the detection. These methods are good in solving the cases in which the amount of change is very high however optimums can be tracked in their own like in [65], [63]. Some of the disadvantages are also along with them like, dependence of the thing that whether the changes that are occurring are known or not, or they are easily detectable or not, another curious thing is the identification of mutation size, if it is too small than the algorithm will converge quickly and random searching will take place, lastly there is a very little information extracted from the previous search algorithms, irrelevant information is the main difficulty in it.

7. Dynamic Environments

Real world problems are hardly static in nature, most of the real life problems and optimization problems are dynamic in nature. For every time interval, the environmental conditions, their behaviors and nature get a slight change. Talking about the mathematical context, we have a optimization problems in a certain intervals of time t and it performs x number of steps at that time, then after a certain interval of time same number of steps will take t+1 interval of time. We can express this scenario in the form of an equation [10], [11].

$$F(\sim x; t) \neq F(\sim x; t') \qquad (1)$$

Where "F" is the function of dynamic optimization problem in equation 1. There are several problems exist in real world related to dynamic optimization problems that are of diverse nature and characteristics. We are trying to classify dynamic environments on the basis of classification schemes that they hold [25-28]. In that classification schemes, we are explaining some of them as follows:

7.1 Fitness Landscape Composition

Landscape composition is very important for dynamic environments, which is majorly responsible for the occurrence of change in these environments that has two dividend branches, homogenous and heterogeneous landscapes. As the name depicts, homogenous landscape is for single underlying function, where on the other hand heterogeneous landscapes have multiple functions in them. Both are responsible for maximum value of functions at that time and to locate the local and global optima of that function. There is also a concept of hardness in landscapes fitness this is actually related to complexity of the optimization problem and its effects on the environment. Usually complex optimization problems take longer time than a simple one and it takes a longer time to locate the presence of local optima in that problems. This means that for hard problems algorithm will converge to a non optimal point which will be a major drawback. Some of the researchers have used statistical and probabilistic measure to solve this problem. It is obvious that in hard surfaces it is difficult to locate the optima point usually these problems have curves and ridges in the graph which makes them hard [29].

7.2 Types of Changes in Dynamic Environments

Most vital obsession for dynamic environment is the change that occurred, that put a pressure on environment to change its rules and blown a windy situation in the environment. Now coming towards the change that occurs in the environment, we are interested in finding the nature of change, severity of change and several other changes that can occur in dynamic environments. Major objective is to recover from these changes and capable to adapt these changes in the environment. There are some changes that are random changes in which there is no pattern exists. There is some change that is chaotic in the environment, which is completely dependent on the previous changes, that consists of further two more sub changes that are linear changes which are highly similar changes and the other one is cyclic changes in which environment come back on the same state [10], [29],[30].

7.3 Change Frequency

As we have discussed early that change occurrence is the metric and considered as the major attribute that effect those changes. The main thing is the optima location that gets change in environment, its values along with new position of optima in the search space. This frequency of change in the dynamic environments can be whether continuous or discrete, this is up to the dynamic optimization problem scenario. This basically tells the tracking conditions in static and dynamic landscape fitness scenarios, if algorithm tracks global optima in static landscape than its not obvious that it can track it in dynamic landscape as well. If the frequency of change in dynamic environments is very high the algorithm will search its search space for quite a long interval of time and locating chance of local optima will be increased, on the other hand if the change frequency will be low than there are less chances of locating local optima and algorithm may not find it best time that can lead to pre mature convergence or high computational cost [31-43].

7.4 Severity of Change

Dynamic environment cater changes depending upon the fitness landscapes in the surface. The measurements for a change is noticed as change occurrence before fitness landscape and then after that. Major influencing factors that influence change in dynamic environments are locating optima, amount that tell location of optima that are changed in dynamic environments. The second one is amount of values that get changed and the last one is overture of new optima and the amputation of the older one. If the severity of change is so diverse and effective that the search space that came has drastically different nature than the search space before change in the dynamic environments. The information that came should be resides in secret in order to main good optimization performance. However the small severity of changes actually exploit the algorithm and its behaviors with the global and local optima that its use less to start from scratch now. In the fig.1 [10] we can see the relationship between the frequency of change and its severity, likelihood is such that by increasing the change in frequency too much will lower the severity and if the frequency is too much low we can see that the severity is immensely high in the graph. There exist a upper bounds between both severity and frequency of changes, which shows their maxima positions in which both have unviable effects and performance for the dynamic optimization problems in dynamic environments [10], [11].

7.5 Using Evolutionary Algorithms in DOP's

In the previous chapter we have discussed about evolutionary algorithms, in whole process of natural evolution information is transferred from one individual to another in the population. We are talking about dynamic environments in which their come a rapid change always and it has to make itself self adaptive to those changes in the environment. Whenever a change arrives, first of all environment should detects it and then it absorbs the new changes in the environment. Fitness information is very important in evolutionary algorithms for solving the dynamic optimization problems that came from the major concept of survival of the fittest in the population [5-6]. There are some of the convergence issues in evolutionary algorithms that are:

A)Whenever a change occurs, we have to generate diversity. In dynamic environments change is obvious and diversity should be enhanced when a change in the environment occurs. Mutations rate should be increased to see the after effects of change. Gradually mutation rate is increased. Determination of diversity in a search space for a particular problem of dynamic environment is difficult, decreasing too much will make high resemblance problems with the parents or previous population and increasing too much will also make high differences in it.

B)During the complete process of running algorithm in dynamic environment, we have to maintain diversity in order to solve the convergence problems.

C)Memory based techniques are also very helpful in evolutionary algorithms in getting the useful information from the past generation. In this approach the results of that particular interval sets are saved and can be used for the future generations for their purposes. Case base reasoning is one of the examples from them. Individuals from the population can reinstate their previous positions.

D)The last convergence issue is multi population approaches in this approach we divide the population in multiple sub populations in order to keep track of multiple peaks in the landscape of fitness. When the multi population is created then each sub population maintain information in itself about the existing regions in the search space.

8. Imperfect Environments

Change detection in any environment come under an intelligent sense of environment, but the point is to handle that change timely and effectively according to situation that does not affect the whole scenario is the main thing in it. Human minds are creative and intelligent too, they have an astonishing behavior to cope with the changing environment and adapt that too. In the recent couple of years there have been a lot of works done in different environments that are static, dynamic or imperfect keeping in regards the evolutionary process [6], [7].

Imperfection in evolutionary environments is quite unusual from the conventional static and dynamic environments in evolutionary domain. This concept of imperfection in environment is first proposed by Kendall and Su [6] in their work presented, they have proposed new endeavors in environments in their effects on them. Naturally complex environments are not static most of them are dynamic as the changing effects are diverse in them. We are interested to raise some major research questions in order to inspire the deepness effects of traditional environments and imperfect environments under the umbrella of evolutionary systems. We are interested to care for each and every parameter that is a part of imperfect evolutionary environment and we will discuss its effect on environment and the strategies that it follows, we are interested to present a improvements that can be added and other flaws if any exists in the previous one to explore and find why is that and how it can be improved by giving some simple strategic examples.

Talking about any real life problem, if we drill down its scenes of making it imperfect and cope with those scenes, we will see that problems are diverse everywhere, problem is not only with the rules, conditions, skills, nature and characteristics of two entities in real world problems but there come other things also that also create problems, other main problems that can happen is the rapid change of algorithm, software and hardware that can be changed, we have to answer those challenges as well and we have to cope with those challenges in order to make a robust imperfect evolutionary environment. So change in that environment is the vital part and in first the idea of imperfect evolutionary environment given by Kendall and Su. They have given a statement about imperfect evolutionary system in which the intelligent individuals have the capability to optimize according their own obtainable resources and they adapt themselves the challenges and the changes that are occurring and evolving in an imperfect environment [6].

Fig.2. represents the diagrammatic view of imperfect environment in different stages, where E^T represents imperfect evolutionary system at time T. There are set of information's available as I_n^T. There are entities present in the environment in which information is present in it. We are considering an imperfect evolutionary environment in which we have stated before that information will be changed with every interval of time. Whenever a change will come it will create an expansion of information and it will enhance the previous information and will be transformed into new information with T+1 interval of time, it may be possible that it can remove or either sustain the old repository of information that is stored as T in the evolutionary environment.

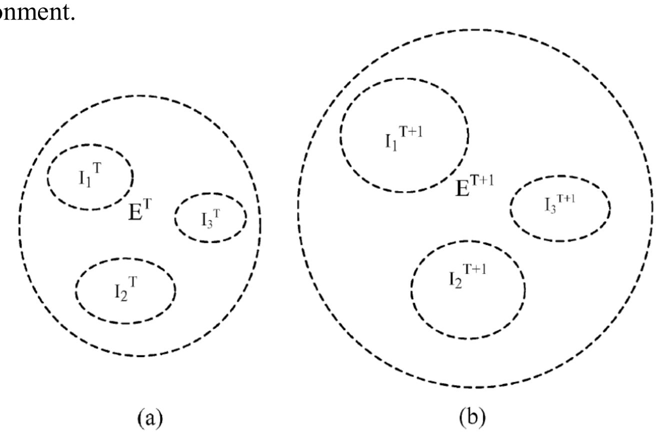

Fig. 2. Imperfect Evolutionary Environment [6].

These imperfect individuals in the environment cannot understand their environment completely and can cause increase in noise and on the other hand we have to make some limitations in the environment surrounding. The main thing is to understand the surrounding environment and to cope with the changes that occur in it. There are agents in the environment that sense the behaviors and changes in the environment. They have their own reaction whenever a change is detected and they respond according to their own in that situation [6].

8.1 Imperfect Evolutionary Environments Components

Every environment is made up from collection of components in their surroundings that makes them a complete system. The basic idea of imperfect systems under evolutionary umbrella was given by Kendall and Su. That system consisted of many small things that make them a complete environment. We are interested about the collective materials used in imperfect environments and we will discuss each component in detail in the discussion below.

As we have drawn the structure of imperfect evolutionary environment in the fig. 2. Those consist of environment, their individuals, information in each environment, changes that occur with them and strategies to cope with those challenges. We can call the imperfect systems as incomplete systems too, having lack of information and sudden change makes them incomplete systems. So, the equation that describes the imperfect environment is as follows:

$E^T = \{ I_1^T, I_2^T, I_3^T, \ldots\ldots I_n^T \}$ (2)

The above equation 2 holds the set of environment variables or information's in each set of environment that resides in it. In each environment E^T there is a complete set of information's that is perceived from it. The information I collected in the set of environment will certainly have some observations from the environment and those set of observations will definably be recorded. Those set of observations by each individuals will have a following equations. There can be many observations by one individual in the environment that is represented by an equation. First of all we will make all possible observations that an individual can take from the environment. Then we will be able to derive an equation for that.

Suppose that an individual has set of participation efforts or a sense from the surrounding environment that could be denoted as S^T, there can be set of rules governed in the environment at that particular time represented by R^T, that rules are very strong and have diverse effects on the environment, individuals will observe them and those will be part of observation set of each individual. There can be entities that can added in the observations those entities, we can called those entities as features that have a diverse effect on individuals and on their environment, we represents them as F^T. There can be impurities in the environment that are noticed by the individuals, we can call them as hybrid impurities that exist in the environment as H^T. There can be mediators M^T present in the environment that evolve with the passage of time. Any change and any type of change can diversely affect all these set of observations in the imperfect environment. So the individuals will be made through the collection of these observations. We are interested in the relationship of these observations with the individuals and their affect on environment, for this we are explaining each observation keeping in view the relationship status as follows:

A)There are participation efforts from the surrounding that are playing their role in the environment as S^T, these efforts are playing very important role at a particular intervals of time, usually the individuals will be affected by these efforts, they are handled by these and they are dependent on these efforts, learning is the basic purpose of this problem. Strategies can get a diverse change by changing these efforts from the external environment.

B) There are some common factors that affect the environment and play a part in their learning we have to consider them very carefully. We cannot ignore them at all. These set of rules R^T are core part of environments that user collected from their observations. These rules are actually controlling the imperfect evolutionary environment of the system. As we are capturing the observations from the environment, so the rules is one of the part of it, rules are grown with the passage of time and also get changed with the intervals of time, some rules are very important and slight change in them can put a assorted affect on the complete scenario of the environment and there are some of the rules whose change is not very effective, system has to cope all the changing that are being caused by the rules and the strategies should be robust to handle the highly concentrated rules in the environment. We have to cope with the severity of rules and their frequency also. Sometimes there can be rules which can cause drastic effects on the complete system they can change the whole scenario of system implicitly.

C) The third observation from the environment by the individual is the number of features FT that are present in the environment from every dimension. Those features can be live and cannot be. The observer is made such intelligent that can collect the behaviors of the environment and its elements that resides inside it. These features are either already presents entities or non-present entities the thing is the important features have to be collected. These features are very important they describe complete environment and its compositions in the surroundings we want to store every feature from the environment so that most of the observations are recorded in the features. Though the complexity of imperfect evolutionary environment will be increased but we have to make sure that we include the most important set of features in the environment so that irrelevant and useless observations cannot be inserted by the individuals in the set.

D) The fourth observation from the user can be the set of observable hybrid impurities H^T, in a surrounding environment there are pure observations as well as impure observations too. Each of which have their own effect. These hybrid impurities are not exactly related to environment but they are playing there roll in the environment. In a learning environment there are strategies which have to be made adaptable by us. Some impurities can be natural incidents related to it, like speed of everything. These impurities are of very much importance and individual have to keep it in the observation list so to keep the environment set more reliable and to make its worth more comprehensible.

E) The last observation is the presence of actors in the environment that are being observed by the individuals in their surroundings. These are named as mediators $M^{T,}$ every mediator or an actor will possibly have contrast characteristics from the other one. Every actor will have unique characteristics and they have their own effect on the environment that can put different observations in the environment. There can be some mediators that are intelligent and can have some mediators that are non intelligent.

Every observation has its own importance and makes its contribution, the one set of observations are taken at T intervals of time, where next interval of observation is taken as T+1 intervals of time. The most important thing is the change in the environment that we have to cope with, we are making to make such imperfect environment in which there are impurities and with the passage of time, environment learns and adds new changes in it with the quality of adaptation. As we have described the possible observations by the observer in the environment and the collection of these observations will make a one individual purely. Any change in a single observation can completely change the scenario of imperfect learning environment. There can be more observations that can be introduced and they can have diverse effect on the environment

8.2 Change that Makes Environment Imperfect

The recognition of imperfect environment always lies in a occurrence of a new change in the environment, that makes it imperfect. We have discussed about the dynamic environment that is, in dynamic environment peaks are always changing and position of every individual in changing but new change does not come, in imperfect environment there is always a new change occurrence that makes them imperfect. We are now explaining it with the help of equations. As we have already mentioned that in imperfect environment, environment is composed of set of individuals that collectively makes to form a complete scenario of environment. Individual in the environment are composed of set of observations that in our analysis composed of participation effort, set of rules, set of features, hybrid impurities and mediators or actors, there can be other observations that depends upon the scenario, we can enhance that set of observations after all. In the fig. 4 of imperfect evolutionary environment, it is mentioned that at time T the individuals observation is this and at time T+1 their observations is this. At time T+1 there comes a possible new change that makes environment imperfect, we have to cater that new change and make our system adaptable to that change. For that purposes we are elaborating our scenario through an equations.

There are set of individuals that are playing their part in environment and each individual is composed of environment variables that the individual learn and observes it. They are represented in the following equation 3.

$$E^T = \{S^T, R^T, F^T, H^T M^T\} \quad (3)$$

Equation 3represents the composition of information's in the environment that makes the whole environment at time T. we have already explained the purpose and information in the above set.

$$I_1^T = \{S^T, R^T, F^T, H^T M^T\} \quad (4)$$

$$I_2^T = \{R^T, F^T, H^T M^T\} \quad (5)$$

$$I_3^T = \{\} \quad (6)$$

Now at time T+1, if the new environment variable or information get inside in the set of environment, than the above equations4, 5 and 6 will be written as following:

$$E^T = \{S^T, R^T, F^T, H^T M^T, \text{New Change}\} \quad (7)$$

Equation7 shows the arrival of the new change in the environment and now the scenarios of individuals will also be changes that are shown below:

$$I_1^T = \{S^T, R^T, F^T, H^T M^T, \text{New Change}\} \quad (8)$$

$$I_2^T = \{R^T, F^T, H^T M^T, \text{New Change}\} \quad (9)$$

$$I_3^T = \{\} \quad (10)$$

Now at time T+1, if the new environment variable or information get inside in the set of environment, than the change will be noted by the individuals as shown in the equation 8, 9 and 10.

We have noticed that at time T, no change has been entered in the equations of individuals and at time T+1, when the change come the new change becomes the part of individuals in next time interval. The main objective is to make the system adaptive to that new change so that it learns the new change, only imperfect individuals can adapt new changes and others cannot. We have proposed the perceptron learning algorithm that might help new change to learn and make the system adaptive to those new changes.

8.3 Association of Every Individuals in its Environment

We have mentioned above the individuals behaviors in the environment, where each individual have a diverse effect on the environment and its surroundings. The above mentioned observations are subject to follow by every individual in their environment. Every individual is composed of set of observations, we have maintained five observation that could be possible in this regards and mentioned in our problem, every individual has its own collection, that collection will be same in the observational points that we have mentioned that are efforts, rules, features, hybrid impurities and mediators but the observation of every individual will obviously be different because it's a imperfect learning framework and environment, every individual behaves independently and works according to its own. The main task of individuals is to gather the observation and note down the every change in their environment that is occurring in it. Every individual is directly related to its environment and performs its contribution in it [6].

8.4 Imperfect Evolutionary Environment Intelligent Model

As we have discussed individuals in detail in the last topic, now we are heading towards finding and introducing an intelligence approach with imperfect environment systems in our model. We want to build a relationship with our surrounding that is responsible for the change in it. Our major purpose is to gather the change information and to make IES to cope with those changes. Individual's major purpose is to gather all possible observations and form in a set [6].

In everyday life human gain experience, learn from the outer environment and evolve according with the passage of time, same is the case with imperfect evolutionary environment, which consists of individuals that are present inside the learning environment, they gain experience, learn from the outer world, delete the previous gather information in them and update them with new information that came. We are interested to that individuals must add some new information in them and remove the older one, so in this way they can keep their knowledge updated. Likewise humans have intelligence in them, environments are not intelligent, and it's the human that makes them intelligent.

8.5 Improvements in IES Intelligent Model

We are proposing a learning algorithm that will introduce intelligence in the imperfect learning environment that will help the environment to learn and cope with the changes that came. We are proposing a multi-layer perceptron a neural networks based approach of learning. Perceptron is used to train the mediators or actors in the environment [3], [4], [5], [6]. We know that a slight change in imperfect evolutionary environment can change the search space and we cannot find the optimal points in any case. This may be thought likewise of dynamic environment but in that optimum is changing its location in the search space not the search space is moving, in imperfect scenario search space is taking jumps. On movement of search space from one environment to other makes the previous environment useless and we have to find the optimum points now in the new environments. We can use multi layer perceptron to manage the conditions after the change come. Individuals can learn better and formulate better strategies by using perceptron in their learning. Intelligence purpose is to make an environment adaptive for change.

We are proposing multi layer perceptron which is an artificial neural network. All the individuals gather information about their environment and perceptron is used to take all the information of individuals. If a change comes than a change detection is detected through a multi layer perceptron [2-10]. Working of perceptron will be such that, there are number of input layers denoted by x, there can be n number of input layers in it. There are weights for every input. There is a bias for every inputs, the weighted summation formula is applied in which every input is multiplied with weights of that inputs and summed. There is a activation function that acts like operating unit for perceptron, there are many activation functions exist, after finding the net input y, we check the output y according to the activation function that is attached with it. There is a target class attached with input layers. After iterations weights are updated and bias is also updated by using both old weights and bias. In this way multi layer perceptron works, our purpose is to make multi layer perceptron for imperfect evolutionary environment learning. That takes inputs from individuals and transfer that information to environment, but before individuals gathers the observation from their surroundings and keep it in.

9. Future Research Directions

In last 2 decades magnificent amount of work have been done in different environment to solve complex optimization problems using evolutionary algorithms, however this research area is very young and emergent, still needs a lot to be done. Rigorous efforts needs to be put up and brought up this field in different tastes of environment, we have seen most of the research is focused on static, uncertain and dynamic environments, where as very small amount of work have been done in imperfect environment, among all of them imperfect environment should be more and more importance. After writing a comprehensive survey on optimization problems, we have suggested the flowing future work:

A) We have surveyed benchmark problems are mostly evaluated and tested on theoretical problems, now the gap to be filled is, we can find the common properties of theoretical problems, and their criteria of evaluation, and if these common properties and evaluation criteria can be applied on real world optimization problems or not. In the literature we have seen that [83] has detailed surveyed and give some answers regarding theoretical research and pointed out the holes in them that can be filled, so we recommend that in future further exploration of dynamic optimization problems can be carried out in real world problems.

B) Large number of techniques have been developed and research is still going on in dynamic optimization problems, researchers are focused to develop more efficient algorithms, we have discussed various number of research techniques used in optimization problems, new addition can be added to make hybrid techniques with less computational cost and high performance in optimization problems that contain more adaptive power to change and memorization ability. Also for imperfect environment more algorithms should be developed that can cater the change on run time that came.

C) Behavioral analysis of dynamic optimization problems needs to be address, it is very necessary for the problem to track the performance of algorithm on the optima locations and solutions, error tracking and velocity tracking can be incorporated along with only optima tracking in the optimization problems.

In the comprehensive survey we have noticed that most of the researchers took the examples from academic research not from the real world problems, but there are some of them from real world problems, but the number of real world applications is low and that are high, in future we recommend to focus on real world problems and also to incorporate swarm intelligence in dynamic optimization problems as well.

10. Conclusions

In this paper, we have reviewed and divide optimization problems particularly dynamic optimization problems from many outlooks, benchmark, performance measures and methodology. We have investigated about the environments, their behaviors, learning in different conditions and performance on environments. We have used the help of evolutionary algorithms that are nature inspired biological algorithms used to solve complex optimization problems in static, dynamic, uncertain and in imperfect environments. We have given a detailed discussion on evolutionary algorithms, strengths and weaknesses and their use in dynamic and in imperfect environments. We tried to address and answer the major research issue about imperfect evolutionary systems, relationship between dynamic and imperfect evolutionary environments and the improvements that can be made in dynamic environment in solving the complex optimization problems. We have analyzed the steps in dynamic environments and use of evolutionary algorithms in solving dynamic optimization problems.

From this survey paper, we can conclude that each evolutionary optimization problem is suitable only in one particular problems due to this the related studies shows that different approaches have been combined to solve the complex optimization problem because single approach is not solving that problems in a better sense. We have also noticed that theoretical work have also been done in dynamic optimization problems but it is quite immature and needs major improvements, but on the other hand these theoretical studies act like a baseline for the new researchers in this field. They can understand the evolutionary optimization problems better by studying the state of the art work, also the work done in imperfect evolutionary environment is too little and it needs major attraction for researchers, as IES is a young and emergent addition in evolutionary systems and a lot of amount of work can be done in this field.

REFERENCES

[1] P. H. Winston, Artificial Intelligence. Reading, Massachusetts: Addison-Wesley. 1984.

[2] D. G. Bobrow, Artificial Intelligence in Perspective. The MIT Press, 1994.

[3] A.P. EngelBrecht, "Evolutionary Computations & Swarm Intelligence", in Computational Intelligence an Introduction, London, UK: John Wily & Sons Inc., 2002

[4] K.A. De Jong, "Introduction", in Evolutionary Computations a Unified Approach, London, UK: MIT Press, 2006

[5] A. Konar, Computational Intelligence Principles, Techniques and Applications. Springer-Verlag 2005

[6] G. Kendall, and Y. Su, "Imperfect Evolutionary Systems," IEEE Transactions on Evolutionary Computations, vol. 11, no. 3, pp. 294–307, 2007.

[7] G. Kendall, and Al. Khateeb, "Introducing Individual and Social Learining into Evolutionary Checkers," IEEE Transactions on Evolutionary Computations, vol. 4, no. 4, pp. 258–269, 2012.

[8] G. Uludug, B. Kiraz, A.Sima and E. Ozcan, "A hybrid multi population framework for dynamic environments combining online and offline learning," Soft Computing, vol. 17, no. 12, pp. 2327–2348, 2013.

[9] C. Cruz, J.Gonzalez and D. Pelta, "Optimization in dynamic environments: a survey on problems, methods and measures." Soft Computing, vol. 15, no. 12, pp. 1427–1448, 2011.

[10] M. Cornelius, "Adaptive Multi Population Different Evolution for Dynamic Environments," Ph.D. dissertation, Univ. Pretoria,, 2012.

[11] Y. Jin, and J. Branke, "Evolutionary optimization in Uncertain Environments-A Survey," IEEE Transactions on Evolutionary Computations, vol. 9, no. 3, pp. 303–317, 2005.

[12] D. V. Arnold, Noisy Optimization with Evolution Strategies. Norwell, MA: Kluwer, 2002.
[13] D. V. Arnold and H.-G. Beyer, "Efficiency and mutation strength adaptation of the ES in a noisy environment," in Parallel Problem-solving from Nature. ser. LNCS, M. Schoenauer et al., Eds. Berlin, Germany: Springer-Verlag, 2000, vol. 1917, pp. 39–48.
[14] NRC, United States National Research Council, Developments in Artificial Intelligence. Funding a Revolution: Government Support for Computing Research. National Academy Press. 1999
[15] R.Jensen, Computational Intelligence and Feature Selection: Rough and Fuzzy Approaches, Wiley-IEEE Press 2008.
[16] A.E. Eiben, and J.E. Smith, Introduction to Evolutionary Computing. Natural Computing Series, 2nd edition, 2007.
[17] D. E. Goldberg, Genetic algorithms in search, optimization and machine learning, Addison-Wesley, 1989.
[18] J. Johnston, The Allure of Machinic Life: Cybernetics, Artificial Life, and the New AI. MIT Press, 2008
[19] D. Goodman and R. Keene, Man versus Machine: Kasparov versus Deep Blue. Cambridge. MA: H3 Publications, 1997.
[20] S. Fazeli and Barkhordari, "An Improved Grouping Genetic Algorithm", J. Appl. Environ. Biol. Sci., 6(1) 38-45, 2016.
[21] Montazeri, Mehdi Zadeh and Mohammad Mosoud "Application of Ants Colony Algorithm for Non ideal Iris Image Segmentation", . J. Appl. Environ. Biol. Sci., 3(12) 30-33, 2013.
[22] C. Bierwirth and H. Kopfer, "Dynamic task scheduling with genetic algorithms in manufacturing systems," Dept. Econ., Univ. Bremen, Bremen, Germany, Tech. Rep., 1994.
[23] C. Bierwirth and D. C. Mattfeld, "Production scheduling and rescheduling with genetic algorithms," Evol. Comput., vol. 7, no. 1, pp. 1–18, 1999.
[24] J. A. Biles, "Genjam: A genetic algorithm for generating jazz solos," in Proc. Int. Comput. Music Conf., Aarhus, Jutland, Denmark, 1994, pp.131–137.
[25] T. Blackwell and I. Branke, "Multiswarm optimization in dynamic environments," in Applications of Evolutionary Computing. ser. LNCS, G. R. Raidl et al., Eds. Berlin, Germany: Springer-Verlag, 2004, vol. 3005, pp. 489–500.
[26] T. M. Blackwell and P. J. Bentley, "Dynamic search with charged swarms," in Proc. Genetic Evol. Comput. Conf., W. B. Langdon et al., Eds., 2002, pp. 19–26.
[27] J. Branke, "Creating robust solutions by means of evolutionary algorithms," in Parallel Problem Solving from Nature, ser. LNCS. Berlin, Germany: Springer-Verlag, 1998, pp. 119–128.
[28] J. Branke "Memory enhanced evolutionary algorithms for changing optimization problems," in Proc. Congr. Evol. Comput., vol. 3, 1999, pp. 1875–1882.
[29] J. Branke, "Evolutionary Optimization in Dynamic Environments. Norwell, MA: Kluwer, 2001.
[30] Brank, J "Reducing the sampling variance when searching for robust solutions," in Proc. Genetic Evol. Comput. Conf., L. Spector et al., Eds., 2001, pp. 235–242.
[31] J. Branke, P. Funes, and F. Thiele, "Evolving en-route caching strategies for the Internet," in Lecture Notes in Computer Science, vol. 3103, Proc. Genetic Evol. Comput. Conf., K. Deb et al., Eds., 2004, pp. 434–446.
[32] J. Branke, T. Kaußler, C. Schmidt, and H. Schmeck, "A multipopulation approach to dynamic optimization problems," in Adaptive Computing in Design and Manufacturing 2000, ser. LNCS. Berlin, Germany: Springer-Verlag, 2000.
[33] J. Branke and D. Mattfeld, "Anticipation in dynamic optimization: The scheduling case," in Parallel Problem Solving from Nature. ser. LNCS,
[34] M. Schoenauer et al., Eds. Berlin, Germany: Springer-Verlag, 2000, vol. 1917, pp. 253–262.
[35] J. Branke and H. Schmeck, "Designing evolutionary algorithms for dynamic optimization problems," in Theory and Application of Evolutionary Computation: Recent Trends, S. Tsutsui and A. Ghosh, Eds. Berlin, Germany: Springer-Verlag, 2002, pp. 239–262.
[36] J. Branke and C. Schmidt, "Selection in the presence of noise," in Lecture Notes in Computer Science, vol. 2723, Proc. Genetic Evol. Comput. Conf., E. Cantu-Paz, Ed., 2003, pp. 766–777.
[37] "Sequential sampling in noisy environments," in Parallel Problem Solving from Nature, ser. LNCS. Berlin, Germany: Springer-Verlag, 2004.
[38] "Fast convergence by means of fitness estimation," Soft Comput., vol. 9, no. 1, pp. 13–20, 2005.
[39] J. Branke, C. Schmidt, and H. Schmeck, "Efficient fitness estimation in noisy environment," in Genetic and Evolutionary Computation, L. Spector et al., Eds. San Mateo, CA: Morgan Kaufmann, 2001, pp. 243–250.
[40] J. Branke and W.Wang, "Theoretical analysis of simple evolution strategies in quickly changing environments," in Lecture Notes in Computer Science, vol. 2723, Proc. Genetic and Evol. Comput. Conf., E. Cantu-Paz et al., Eds., 2003, pp. 537–548.
[41] D. Büche, "Accelerating evolutionary algorithms using fitness function models," in Proc. GECCO Workshop Learn., Adapt. Approx. Evol. Comput., 2003, pp. 166–169.
[42] D. Büche, P. Stoll, R. Dornberger, and P. Koumoutsakos, "Multiobjective evolutionary algorithm for the optimization of noisy combustion problems," IEEE Trans. Syst., Man, Cybern., Part C, vol. 32, no. 4, pp. 460–473, 2002.
[43] L. Bull, "On model-based evolutionary computation," Soft Comput., vol. 3, pp. 76–82, 1999.
[44] K. Burnham and D. Anderson, Model Selection and Multimodel Inference, 2nd ed. Berlin, Germany: Springer-Verlag, 2002.
[45] E. Cantu-Paz, "Adaptive sampling for noisy problems," in Proc. Genetic Evol. Comput. Conf., 2004, pp. 947–958.
[46] A. Carlisle and G. Dozier, "Tracking changing extrema with adaptive particle swarm optimizer," in Proc. World Automation Cong., Orlando, FL, 2002, pp. 265–270.
[47] L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, pp. 338–353, 1965.
[48] J. Branke, Evolutionary Optimization in Dynamic Environments, Kluwer,1394 2001.
[49] C.-K. Goh, K. C. Tan, Evolutionary Multi-objective 1395 Optimization in Uncertain Environments: Issues and Algorithms, Springer Publishing Company, Incorporated, 2009.
[50] J. Branke, Memory enhanced evolutionary algorithms for changing optimization problems, in: IEEE Congress on Evolutionary Computation, CEC, Vol. 3, IEEE, 1999, pp. 1875–1882.
[51] S. Mojtaba, Ariffin, Tang S.H and A.S. Azfanizam, "Fuzzy Based Approach for Monitoring the Mean and Range of the Product Quality",. J. Appl. Environ. Biol. Sci., 4(9) 1-7, 2014.
[52] A. Younes, Adapting evolutionary approaches for optimization in dynamic environments, Doctor of Philosophy (PhD) in Systems Design Engineering, Faculty of Engineering, University of Waterloo, Canada, Faculty of Engineering, University of Waterloo, Canada (2006).
[53] C. L. Ramsey, J. J. Grefenstette, Case-based initialization of genetic algorithms, in: S. Forrest (Ed.), International Conference on Genetic Algorithms, Morgan Kaufmann, 1993, pp. 84–91

[54] Shahid Akbar, Ashfaq Ahmad, Maqsood Hayat and Faheem Ali, “Face Recognition Using Hybrid Feature Space in Conjunction with Support Vector Machine”, J. Appl. Environ. Biol. Sci., 5(7) 28-36, 2015.
[55] A. Simo˜es, E. Costa, Evolutionary algorithms for dynamic environments: Prediction using linear regression and markov chains, in: International Conference on Parallel Problem Solving from Nature, PPSN, Vol. 5199 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2008, pp. 306–315.
[56] H. Richter, S. Yang, Memory based on abstraction for dynamic fitness functions, in: . Mario Giacobini et al (Ed.), Applications of Evolutionary Computing, Vol. 4974 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2008, pp. 596–605.
[57] J. Branke, Evolutionary approaches to dynamic optimization problems – introduction and recent trends, in: J. Branke (Ed.), GECCO Workshop on Evolutionary Algorithms for Dynamic Optimization Problems, 2003, pp. 2– 4.
[58] C. Rossi, M. Abderrahim, J. C. D´ıaz, Tracking moving optima using kalmanbased predictions, Evolutionary Computation 16 (1) (2008) 1–30.
[59] A. Simo˜es, E. Costa, Improving prediction in evolutionary algorithms for dynamic environments, in: . Raidl et al (Ed.), Genetic and Evolutionary Computation Conference, GECCO, ACM, Montreal, Qu´ebec, Canada, 2009, pp. 875–882.
[60] P. A. N. Bosman, H. L. Poutr´e, Learning and anticipation in online dynamic optimization with evolutionary algorithms: the stochastic case, in: Genetic and Evolutionary Computation Conference, GECCO, ACM, New York, NY, USA, 2007, pp. 1165–1172.
[61] J. Branke, D. Mattfeld, Anticipation and flexibility in dynamic scheduling, International Journal of Production Research 43 (15) (2005) 3103–3129.
[62] R. K. Ursem, Multinational GA optimization techniques in dynamic environments, in: D.Whitley, D. Goldberg, E. Cantu-Paz, L. Spector, I. Parmee, H.-G. Beyer (Eds.), Genetic and Evolutionary Computation Conference, GECCO, Morgan Kaufmann, 2000, pp. 19–26.
[63] S. Yang, Constructing dynamic test environments for genetic algorithms based on problem difficulty, in: IEEE Congress on Evolutionary Compu1407 CEC, Vol. 2, 2004, pp. 1262–1269.
[64] P. A. N. Bosman, Learning, anticipation and time-deception in evolutionary online dynamic optimization, in: S. Yang, J. Branke (Eds.), GECCO Workshop on Evolutionary Algorithms for Dynamic Optimization, 2005.
[65] T. T. Nguyen, Z. Yang, S. Bonsall, Dynamic time-linkage problems - the challenges, in: IEEE RIVF International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future, 2012, in Press.
[66] T. T. Nguyen, X. Yao, Continuous dynamic constrained optimization- the challenges, IEEE Transactions on Evolutionary Computation (In Press)[online].
URL\url{http://www.staff.ljmu.ac.uk/enrtngu1/Papers/Nguyen_Yao_DCOP.pdf}
[67] T. T. Nguyen, X. Yao, Benchmarking and solving dynamic constrained problem in: IEEE Congress on Evolutionary Computation, CEC, IEEE Press, 2009, pp. 690–697.
[68] H. Richter, Memory design for constrained dynamic optimization problems, in: The European Conference on the Applications of Evolutionary Computation, Evo Applications, Vol. 6024 of Lecture Notes in Computer Science, Springer, 2010, pp. 552–561.
[69] H. G. Cobb, J. J. Grefenstette, Genetic algorithms for tracking changing environments, in: International Conference on Genetic Algorithms, Morgan Kaufmann, 1993, pp. 523–530.
[70] K. Trojanowski, Z. Michalewicz, Searching for optima 1430 in non-stationary environments, in: IEEE Congress on Evolutionary Computation, CEC, Vol. 3, IEEE, 1999, pp. 1843–1850.
[71] C. Li, S. Yang, T. T. Nguyen, E. L. Yu, X. Yao, Y. Jin, H.-G. Beyer, P. . N. Suganthan, Benchmark Generator for CEC 2009 Competition on Dynamic Optimization, Tech. rep., University of Leicester and University of Birmingham, UK (2008).
[72] M. Abello, L. T. Bui, Z. Michalewicz, An adaptive approach for solving dynamic scheduling with time-varying number of tasks - part I, in: IEEE Congress on Evolutionary Computation, CEC, 2011, pp. 1711 –1718.
[73] Adaptive encoding for aerodynamic shape optimization using evolution strategies, Vol. 1.
[74] Y. Jin, M. Olhofer, B. Sendhoff, On evolutionary optimization of large problems using small populations, in: The First International Conference on Advances in Natural Computation, ICNC 2005, 2005, Part II, 2005, pp. 1145–1154.
[75] Y. Jin, B. Sendhoff, Constructing dynamic optimization test problems using the multi-objective optimization concept, in: G. R. Raidl (Ed.), Applications of evolutionary computing, Vol. 3005 of Lecture Notes in Computer Science, Springer, 2004, pp. 525–536.
[76] M. Farina, K. Deb, P. Amato, Dynamic multiobjective optimization problems: test cases, approximations, and applications, IEEE Transactions on Evolutionary Computation 8 (5) (2004) 425–442.
[77] M. Helbig, A. Engelbrecht, Archive management for dynamic multi-objective optimization problems using vector evaluated particle swarm optimization, in: IEEE Congress on Evolutionary Computation, CEC, 2011, pp. 2047 – 2054.
[78] A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, E. Tsang, Prediction-based pop1ulation re-initialization for evolutionary dynamic multi-objective optimization, in: S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, T. Murata (Eds.), Evolutionary Multi-Criterion Optimization, Vol. 4403 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2007, pp. 832–846.
[79] X. Yu, Y. Jin, K. Tang, X. Yao, Robust optimization over time – a new perspective on dynamic optimization problems, in: IEEE Congress on Evolutionary Computation, CEC, Spain, 2010, pp. 3998–4003.
[80] R. W. Morrison, K. A. DeJong, A test problem generator 1465 for non-stationary environments, in: IEEE Congress on Evolutionary Computation, CEC, Vol. 3, IEEE, 1999, pp. 2047–2053.
[81] R. Morrison, Performance measurement in dynamic environments, in: J. Branke (Ed.), GECCO Workshop on Evolutionary Algorithms for Dynamic Optimization Problems, 2003, pp. 5–8.
[82] J. J. Grefenstette, Evolvability in dynamic fitness landscapes: A genetic algorithm approach, in: IEEE Congress on Evolutionary Computation, CEC, Vol. 3, IEEE, 1999, pp. 2031–2038.
[83] K. Weicker, N. Weicker, Dynamic rotation and partial visibility, in: IEEE Congress on Evolutionary Computation, CEC, 2000, pp. 1125–1131.
[84] W. Tfaili, J. Dr´eo, P. Siarry, Fitting of an ant colony approach to dynamic optimization through a new set of test functions, International Journal of Computational Intelligence Research 3 (2007) 205–218.
[85] R. Tinos, S. Yang, Continuous dynamic problem generators for evolutionary algorithms, in: IEEE Congress on Evolutionary Computation, CEC, 2007, pp. 236–243.
[86] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y.-P. Chen, A. Auger, S. Tiwari, Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization, Tech. rep., Nanyang Technology University, Singapore (2005).