12-20-2018

Study the Effects of Multilevel Selection in Multi-Population Cultural Algorithm

Dilpreet Singh
University of Windsor

Study the Effects of Multilevel Selection in Multi-Population Cultural Algorithms

By

Dilpreet Singh

A Thesis

Submitted to the Faculty of Graduate Studies

through the School of Computer Science

in Partial Fulfillment of the Requirements for

the Degree of Master of Science at the

University of Windsor

Windsor, Ontario, Canada

2018

Study the Effects of Multilevel Selection in Multi-Population Cultural Algorithms

by

Dilpreet Singh

APPROVED BY:

R. Razavi Far

Department of Electrical and Computer Engineering

M. Kargar

School of Computer Science

P. Moradian Zadeh

School of Computer Science

Z. Kobti

School of Computer Science

September 27th, 2018

Declaration of Co-Authorship/ Previous Publication

I. Co-Authorship

I hereby declare that this thesis incorporates material that is result of joint research, as follows: Chapter 3 of the thesis was co-authored by Pooya Moradian Zadeh and Ziad Kobti. In all cases, the key ideas, primary contributions, experimental design and results and writings were performed by the author and the co-author and the supervisor provided feedback on the refinement of the ideas and editing of the manuscript.

II. Previous Publication

This thesis includes 1 original paper that have been previously published/ submitted for publication in peer reviewed conference, as follows:

Section	Full Citation	Publication status
3.2	Dilpreet Singh, Pooya Moradian Zadeh and Ziad Kobti. ”A Multilevel Cooperative Multi-Population Cultural Algorithm”. In IEEE International Conference on Innovations in Intelligent Systems and Applications (INISTA 2018) , Thessaloniki, Greece, 2018.	Accepted

I certify that I have obtained a written permission from the copyright owner to include the above published material in my thesis. I certify that the above material describes work completed during my registration as a graduate student at the University of Windsor.

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any propriety rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices.

Furthermore, to the extent that I have included copyright material that surpasses the bounds of fair dealing within the meaning of Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner to include such material in my thesis.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies Office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

This is a study on the effects of multilevel selection (MLS) theory in optimizing numerical functions. Based on this theory, a new architecture for Multi-Population Cultural Algorithm is proposed which incorporates a new multilevel selection framework (ML-MPCA). The approach used in this paper is based on biological group selection theory that states natural selection acts collectively on all the members of a given group. The effects of cooperation are studied using n-player prisoner's dilemma. In this game, *N* individuals are randomly divided into *m* groups and individuals independently choose to be a either cooperator or defector. A two-level selection process is introduced namely within group selection and between group selection. Individuals interact with the other members of the group in an evolutionary game that determines their fitness. The principal idea behind incorporating this multilevel selection model is to avoid premature convergence and to escape from local optima and for better exploration of the search space. We test our algorithm using the CEC 2015 expensive benchmark functions to evaluate its performance. These problems are a set of 15 functions which includes varied function categories. We show that our proposed algorithm improves solution accuracy and consistency. For 10-dimensional problems the proposed method hs 8 out 15 better results and for 30-dimensional problems we have 11 out of 15 better results when compared to the existing algorithms. The proposed model can be extended to more than two levels of selection and can also include migration.

Dedication

I would like to dedicate this to my family and friends.

Acknowledgments

There are many people to whom I would like to acknowledge for their help and support for my journey of the master thesis.

First and foremost I would pay my gratitude to my supervisor Dr. Ziad Kobti. Under his guidance, I had enjoyed a lot working on my research work. It was a great pleasure to work with him. Without his support, this won't have been possible. I would also like to appreciate the amount of time he has invested in me, the funding he provided and also the knowledge he shared with me.

I would also like to thank Mrs. Gloria Menash, secretary of the director who helped me arrange meetings with the professor. Also, my sincere thanks to Mrs. Karen Bourdeau, for help and support, and taking care of all the other issues related to my master's degree.

I am thankful of my committee members Dr. Pooya Moradian Zadeh my co-supervisor, Dr. Roozbeh Razavi Far my external program reader, Dr. Mehdi Kargar my internal program reader for their support.

I am also very thankful to my friends for their moral support and listening to my problems for long hours.

Finally, I would like to thank my parents for their unconditional love and support.

Dilpreet Singh

Contents

List of Tables

List of Figures

List of Abbreviations

EA	-	Evolutionary Algorithms
CA	-	Cultural Algorithms
GA	-	Genetic Algorithms
MPCA	-	Multi-Population Cultural Algorithms
D-MPCA	-	Dominance Multi-Population Cultural Algorithm
Haploidy	-	Single set of a chromosome
Diploidy	-	Two sets of chromosomes of the same gene
H-MPCA	-	Heterogeneous Multi-Population Cultural Algorithm
MCAKM	-	A Novel Multi-Population Cultural Algorithm Adopting Knowledge Migration
ML-MPCA	-	Multilevel Cooperative MPCA

Chapter 1

Introduction

Evolutionary optimization has achieved great success on many numerical and combinatorial optimization problems in recent years [1]. However classical evolutionary algorithms (EAs) often lose their efficacy and advantages when applied to large and complex problems. Their performance deteriorates rapidly as the dimensionality of the search space increases. Optimization is finding the best result by maximizing the desired factors and minimizing the undesired ones. Optimization problems are used to find the best solutions from all the feasible solutions. They are applied to a wide range of areas like energy utilization, supply chain management, job scheduling, solving mathematical problems and much more. Optimization problems are used for minimization and maximization. Evolutionary algorithms (EA) have been used widely by the researchers to solve the optimization problems. EA optimizes the problem efficiently as it contains the search space and searches for the best possible solution in it. The solutions can be either near optimal or optimal. EA allows the exploration and exploitation of the search space. Exploration helps to search the broader space and exploitation helps tune the solution. The model designed balances exploration and exploitation using a hierarchical multilevel multi-population approach.

The problem with EA is that it can fall into local optima (solutions think its optimal solution, but it is not) and easily lose diversity (solutions create clones).

Diversity can be maintained among the population by using Multi-Population Cultural Algorithm (MPCA)[18]. MPCA is a class of EA which is most widely used to solve multi-objective problems. Introducing multilevel selection framework in MPCA can address the issue of falling into local optima. The proposed model uses operators to introduce diversity by expanding the scope of search process at the expense of less promising members of the population. A number of evolutionary algorithms have been developed, each of them introduced novel mechanisms and improvements. Current trends are towards more complex algorithms based on mathematical and computational concepts, and advanced evolutionary-based concepts are often pushed aside. This work introduces modern evolutionary concepts which can lead to an increase in performance of MPCA. One of them is multi-level selection theory originally proposed by Sober and Wilson [2][3]. We will study the effects cooperation using our proposed technique with n-player prisoners dilemma problem and also test it on well-known benchmark problems. [13]

1.1 Evolutionary Computation

Evolutionary computation is often viewed as an optimization process, as it draws inspiration from the Darwinian principle of variation and natural selection. EC which is used for metaheuristic and stochastic optimization of complex problems. There are various algorithms which come under EC, such as:

1. Cultural Algorithms [17]

2. Genetic Algorithms

3. Differential Evolution

4. Particle Swarm Optimization

The underlying concept of evolutionary algorithm is common: there is a given set of population which, under environmental pressure causes natural selection. The fitness function measures the fitness of the candidates, and the better candidates survive for the next generation, discarding the worst ones. Evolution of each individual is carried out by applying mutation and recombination operators on it. Mutation is applied on one candidate and as a result, we get one new candidate while in recombination two candidates (called parents) are selected, and it results in one or more new candidates (called offspring's). Mutation and recombination operators lead to a new set of candidates (offspring's) which replace the existing old candidates for the next generation. This process iterates until a termination condition is achieved. Figure 1 depicts the pseudocode of the evolutionary algorithm.

```
Evolutionary Algorithm();
        Initialize population;
        Evaluate initial population;
        WHILE convergence criteria IS NOT satisfied, DO
                Selection technique;
                Crossover operations;
                Mutation operators;
                Evaluation operators;
                Update Population;
        END WHILE
```

Figure 1.1: Pseudo-code for EA

Unlike traditional EAs, Cultural algorithm uses knowledge of the agent to solve complex search and optimization problems. To make use of the knowledge possessed by the individuals or population Reynolds [17] introduced Cultural Algorithms (CA). Cultural Algorithm incorporates knowledge to direct the search process. In CA the knowledge is extracted and incorporated to revise its search mechanism. The extracted knowledge helps the CA to find solutions with better quality and improves the convergence rate. CA is inspired from the biological model of human culture and beliefs. Cultural Algorithm have two components: population space and belief space. Population space is consist of individuals in the population and belief space stores the knowledge of the best individual of the population in the current generation. Cultural Algorithm incorporates different knowledge sources like situational, topological, historical, normative and domain.

Cultural Algorithms with single population have a high chance of losing diversity and can be difficult to implement on real world problems with dynamic populations. To overcome this Multi-population Cultural Algorithm (MPCA) were introduced. The major problem with standard EA used for dynamic optimization problems appears to be that EA eventually converges to an optimum and loses its diversity which is necessary for exploring the search space. MPCA consist of multiple populations which increases diversity in the population. They resemble more with the real world problems where the nature of problems is more dynamic and continuously varying over a range. In MPCA there are more parameters which can be adjusted when compared to CA. MPCA also allows exploring the large region of search space due to its widespread population. Incorporating different sub-population can solve the complex optimization problems with dynamic nature. Group selection can be incorporated in MPCA to increase the convergence rate and to escape from local optima,. Group selection also increases the diversity in the population. The proposed model in this thesis shows the potential for better results.

1.2 Evolution of Cooperation

Cooperation is needed for evolution to construct new levels of organization. [2] The emergence of genomes, cells, multi-cellular organisms, social insects and human society are all based on cooperation. Evolutionary Computation may fail to solve problems which require a set of cooperative individuals to jointly perform a computational task. Hence special mechanism should be implemented to evolve cooperation in Evolutionary algorithms. In this work one of the mechanisms is used from the work by Nowak[2] for the evolution of cooperation. Five mechanism described by Nowak are: kin selection, direct reciprocity, indirect reciprocity, network reciprocity and group selection. Here group selection mechanism, also known as multilevel selection, is introduced in MPCA for numerical function optimization.

Cooperation means that selfish replicators forgo some of their reproductive potential to help one another. Natural selection implies competition and therefore opposes cooperation unless a specific mechanism is at work. Evolution is constructive because of cooperation. New levels of organization evolve when the competing units on the lower level begin to cooperate. Cooperation allows specialization and thereby promotes biological diversity. Perhaps the most remarkable aspect of evolution is its ability to generate cooperation in a competitive world.

Cooperation on many levels of biological organization is observed. Genes cooperate in genomes. Chromosomes cooperate in eukaryotic cells. Cells cooperate in multi-cellular organisms. There are many examples for cooperation among animals. Humans are the champions of cooperation: from hunter gatherer societies to nation states, cooperation is the main organizing principle of human society. No other life form on earth is engaged in the same complex games of cooperation and defection[2]. The question how natural selection can lead to cooperative behavior has fascinated evolutionary biologists for several decades.

A cooperator is someone who pays a cost, c, for another individual to receive a benefit, b. A defector has no cost and does not deal out benefits. Cost and benefit are measured in terms of fitness. Reproduction can be genetic or cultural. In any mixed population, defectors have a higher average fitness than cooperators Therefore, selection acts to increase the relative abundance of defectors. After some time cooperators vanish from the population. Remarkably, however, a population of only cooperators has the highest average fitness, while a population of only defectors has the lowest. Thus, natural selection constantly reduces the average fitness of the population. (Fig 2).

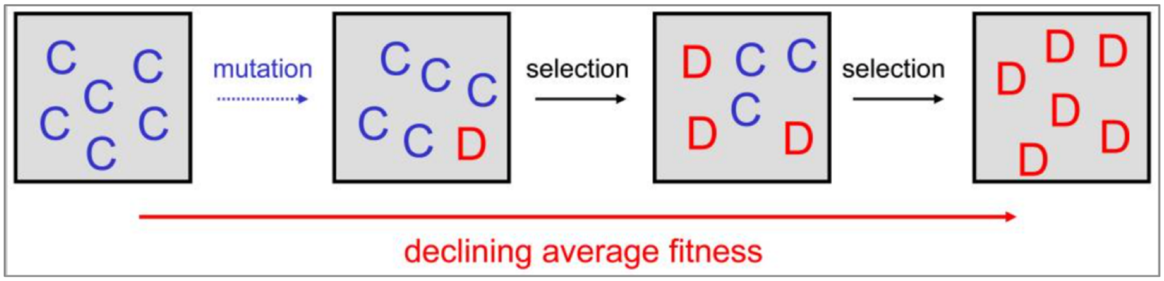

Figure 1.2: Without any mechanism for the evolution of cooperation, natural selection favors defectors.

1.3 Multilevel Selection Theory

Selection does not only act on individuals but also on groups. A group of cooperators might be more successful than a group of defectors. There have been many theoretical and empirical studies of group selection with some controversy, and most recently there is a renaissance of such ideas under the heading of 'multi-level selection'. The concept of multilevel selection is very simple, the different levels are like "Russian matryoshka dolls" (Wilson and Wilson, 2008) nested one within another. Many models have been based on this concept [7]. However the main focus is to investigate under which conditions the evolution of cooperation will occur or what mechanism can promote the evolution of cooperation.

A simple model of group selection works as follows: A population is subdivided into groups. Cooperators help others in their own group. Defectors do not help. Individuals reproduce proportional to their payoff. Offspring are added to the same group. If a group reaches a certain size it can split into two. In this case, another group becomes extinct in order to constrain the total population size. Note that only individuals reproduce, but selection emerges on two levels. There is competition between groups because some groups grow faster and split more often. In particular, pure cooperator groups grow faster than pure defector groups, while in any mixed group defectors reproduce faster than cooperators. Therefore, selection on the lower level (within groups) favors defectors, while selection on the higher level (between groups) favors cooperators. This model is based on ‘group fecundity selection’, which means groups of cooperators have a higher rate of splitting in two. We can also imagine a model based on ‘group viability selection’, where groups of cooperators are less likely to go extinct. Martin A. Nowak[2] gave the equation to evolution of cooperation by group selection :

$$b/c > 1 + n/m.$$

n is the maximum group size and m the number of groups.

1.4 Research Motivation

The inspiration of this research comes from group selection theory. The results provided by Banzhaf and Wu 2010, 2011 on using group selection model in evolutionary algorithm motivated us to apply the multilevel selection approach in the population space of multi-population cultural algorithm. The group selection promotes the emergence of cooperation through evolution. We will use the designed algorithm to study the effect of cooperation and also evaluate our proposed technique on well-known benchmark problems.

1.5 Thesis statement

In this thesis work, “Study of Multilevel Selection in Multi-Population Cultural Algorithms” we will study the effects of cooperation using our proposed multilevel cooperative multi-population cultural algorithm(ML-MPCA) to evolve cooperative agents. We will also test our algorithm on CEC’15 expensive benchmark problems and analyze the results.

1.6 Thesis Contribution

In our work, we aim to study the effects of Multi-Population Cultural Algorithm. The approach used in this paper is based on biological group selection theory. We have developed our MPCA framework based on the work done by Raessi [18] and implemented multilevel selection framework to study the effects of cooperation among populations. CEC 2015 [13] expensive benchmark functions have been used to test our framework and compare it with other existing algorithms. Testing is done on both 10 and 30-dimensional functions of CEC. The functions consist of different types like unimodal, simple multimodal, hybrid and composite functions.

1.7 Thesis Outline

The chapters of our research are organized in the following manner:

Chapter 1 contains the background, motivation and contribution of our research.

Chapter 2 describes in details the related work done in this field. It contains literature review of Cultural Algorithms, Multi-population Cultural Algorithms, Evolution of Cooperation and Multilevel Selection theory.

Chapter 3 describes the proposed algorithm and its implementations.

Chapter 4 provides all the details of experimental approach which contains outline

of CEC functions, experimental setup and all the assumptions made.

Chapter 5 contains the discussion on the results.

Chapter 6 contains the conclusion and future work of our dissertation.

Chapter 2

Related Work

This chapter consists of all the related work used for the building of the fundamental concepts, developing of our framework and architecture of our thesis. In this section, we explain the literature related to Multi-Population Cultural Algorithm, Multilevel Selection/Group Selection and Evolution of Cooperation. The first section contains the Literature review of the related algorithms like Cultural Algorithm, Genetic Algorithm, and Multi-Population Cultural Algorithm. The second section of this chapter contains details of Multilevel Selection in Evolutionary Algorithms. The third section contains paper related to Evolution of Cooperation while the last section consists of papers related to Multi-Population Cultural Algorithms and ends with the conclusion.

2.1 Literature Review

This section consists of detailed explanation about the evolutionary algorithms (EA), different types of EA, Genetic Algorithm, Cultural Algorithm, Multi-Population Cultural Algorithm.

2.1.1 Evolutionary Algorithm

The Evolutionary algorithms (EAs) are a subset of those methods which has been successfully used in the past for optimization problems[39]. EAs are inspired by the biological model of evolution and the process of natural selection. EAs are generic population based meta-heuristic optimization algorithms. In EAs the population is randomly initialized over specific search space which is called the initial population. It incorporates evolutionary operators which include mutation and crossover. These operators creates new offspring's from the parent in the population. The selection operator selects the population with greater fitness from the parent and offspring which serves as population for next generation. The individuals left are discarded from the population. The process continuous until the termination criteria is fulfilled which can be either reaching a maximum number of predefined generations or CPU time. EA are based on the model of biological evolution. To solve a problem, a particular environment can be created where potential solutions can evolve. The parameters of the problem creates the environment which helps to evolve a good solution. EAs are a group of a probabilistic algorithm which is similar to the biological systems and artificial systems. There are many types of EA such as:

1. Genetic Algorithm
2. Cultural Algorithm
3. Multi-Population Cultural Algorithm

2.1.2 Genetic Algorithm

Genetic Algorithms (GA) are population-based evolutionary algorithms; subset of EAs. GA was first introduced by Holland [23] but became popular after the works of Goldberg [41]. GAs mainly is used to solve the search related problems and other optimization problems. When very less is known about the domain GA serves as very important algorithm. Genetic algorithm consist of a group of individuals known as population, and these individuals are used to find the optimal solution within the specified search space. An initial random population is generated over the search space and evolutionary operators like mutation, recombination and selection are applied to them. In GAs after each generation, the best individuals are selected for mutation, recombination, selection, and crossover. The individuals also exchange knowledge among them by making use of the operators. GAs are very simple to code, and the population is not initialized at one point. Instead, they are spread across the search space for exploration. GAs use mutation, crossover, and selection operator to achieve an optimal solution and enhance exploration and exploitation. The genetic algorithm operators are as follows:

1. Crossover :

This operator works similar to the biological model of reproduction. Two individuals are selected from the current generation (parents) on their fitness basis and are allowed to generate a new individual (offspring) in the next generation. This operator enhances the exploration in the search space.

2. Mutation :

This operator is used to change or flip the solution of the individual, and hence it is rarely used in GA.

3. Selection :

The selection operator behaves similarly to the natural selection that is found in biological systems. The selection operator selects the best individuals in the current generation based on their fitness. The fitter individuals are selected, and the weaker ones are discarded from the pool of individuals.

2.1.3 Cultural Algorithm

Cultural Algorithm (CA) is an Evolutionary Algorithm which is inspired by the model of the human evolution process. It incorporates knowledge which is used to direct the search spaces. The knowledge extracted by CA in belief space is incorporated for benefiting its search mechanism. The extracted knowledge helps the CA to find better quality solutions and also helps in improving the convergence rate.

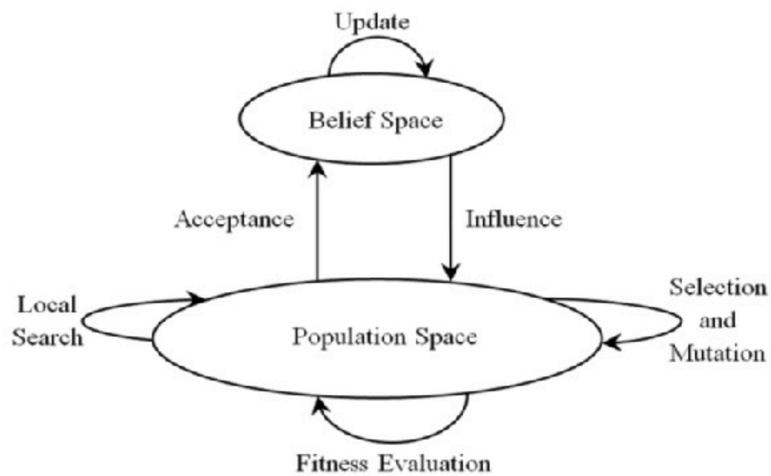

Figure 2.1: Architecture of CA

Figure 2.1 illustrates the architecture of CA. From the figure we can see, CA has population space, unlike any other EA where individuals reside. This space is managed by the EA like GA. CA has belief space which incorporates knowledge. This space stores and update the knowledge extracted over generations. Both of the space communicate with each other by using the acceptance and influence operators. The knowledge circulation is defined as below.

1. The belief space receives the top performing individuals in the generation g from the population space by making the use of acceptance function.
2. The belief space updates its knowledge.
3. The belief space sends the update knowledge to the population space using influence function in the next generation g+1.

4. The population uses the knowledge to generate offspring' for next generation g+1 from current generation g.
5. The top individuals from the next generation g+1 are sent to the belief space to update its knowledge.

This cycle continuous until the termination condition is reached. The CA works like other EAs, but instead of using the random operators it uses knowledge-based evolutionary operators. Cultural Algorithm consists of two components.

1. Belief Space
2. Population Space

Belief Space

Belief space consists of different kinds of knowledge which are helpful in solving the specific problem. Due to this belief space is divided into separate categories. These categories contain different kinds of knowledge depending on which the population poses in the search space. The belief space is a repository where the knowledge is stored and is used by the population to obtain an optimal result. The belief space is updated after every iteration by the best individual in the search space. This best individual helps the other individuals in the population to help them move towards better search space. Artificial belief space stores the knowledge which is gained during the execution of the algorithm and makes use of it in the next generation and for its generic evolution. There are different types of knowledge in the belief space that are as follows: [19].

1. Situational Knowledge
2. Normative Knowledge
3. Topological Knowledge

4. Historical Knowledge

5. Domain Knowledge

Population Space

Population space consists of the individuals in the population. The population component of CA is similar to that of GA. There are two function in CA which allows the individual to move from population space to belief space and vice versa. The acceptance function and the influence function. The acceptance function transfers the best individual from the population space into belief space. After that the belief space updates its knowledge. Then it updates the population space by making use of influence function. The individuals in the population space makes use of this knowledge to generate individuals for the next generation [20].

2.1.4 Multi-Population Cultural Algorithm

Multi-population Cultural Algorithm (MPCA) can be considered as an extension of cultural algorithms. They are used to solve the optimization problems similar to CA. MPCA is CA incorporating multiple populations. Digalakis et al. [21] were the first to introduce MPCA in their work to solve the electric generator scheduling problem. In the first model of MPCA, only the best solutions coming from each sub-populations were exchanged regarding migration rules. However, the best solutions only accounted for the current limited optimal information. MPCA has a number of parameters to optimize when they are compared with the traditional CAs. For example, the number of the subpopulations, the size of a subpopulation, the migration rules and a number of individuals migrating. There are various authors who have implemented MPCA in their work. Guo et. al [23] successfully implemented MPCA for the multi-modal optimization problem, Yi-nan et al. [22] for interactive optimization and constrained optimization problems. Alami et. al [42] also proposed a method of dividing the sub-population based on fuzzy clustering and introduced the concept of cultural exchange between the subpopulations. According to them, the cultural exchange meant to exchange information among belief space of sub-populations. Hylnka et. al [24] also implemented a method to migrate agent among sub-population for the optimization problem. Raessi et. al [18] introduced a new concept to solve the optimization problem in which the subpopulations remained the same. Instead, the optimization parameters were divided among them.

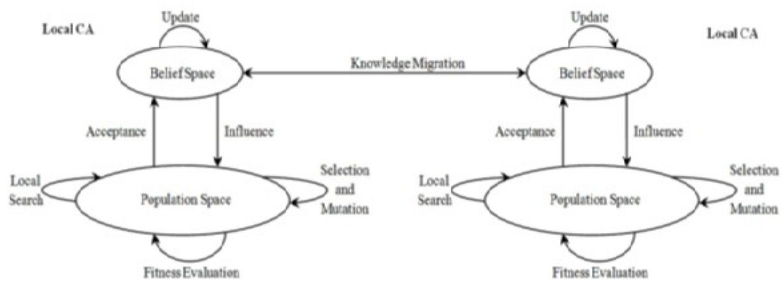

Figure 2.2: Architecture of MPCA

There are many versions of MPCA like Multi-Population Cultural Genetic Algorithms (MCGA), Multi-Population Cultural Differential Evolution (MCDE) and Multi Population Cooperative Particle Swarm Cultural Algorithm (MCPSCA). The architecture of MPCA is depicted in figure 4 [20].

2.2 Group Selection and Multilevel Selection

In nature, the success of cooperation is witnessed at all levels of biological organization. A growing number of biologists have come to believe that the theory of group selection is the explanation, even though this theory has been unpopular for some decades [2]. Group selection models divide individuals into groups, where they only get to interact with members of the same group. Selection operates within groups and between groups. Within-group selection equals natural selection as understood commonly; it selects individuals in a group proportionally to fitness. Individuals, therefore, compete against each other in the pursuit of their own interests. Between-group selection, in contrast, examines the total productivity of groups, and prefers the group with the best performance or the group whose individuals cooperate best. It forces individuals to co-adapt so that a cohesive group can be formed. It also resolves and reduces conflicts within groups, because conflicts would compromise group performance. In short, competition between groups encourages the emergence of cooperation within groups. In group selection models, individuals and groups are relative: groups can be regarded as individuals on a higher level, so that a new level of dynamics can act upon them. In this way, a hierarchical or nested structure can be constructed. This new perspective is now called multilevel selection (MLS) theory [6].

Competition between groups not only helps to construct hierarchies, but also accelerates evolution, as demonstrated by Banzhaf [7] through a series of experiments on a very simple artificial chemistry system.

2.2 Multilevel Selection in Evolutionary Algorithms

2.2.1 Investigations of Wilson's and Traulsen's Group Selection Models in Evolutionary Computation.

Evolving cooperation by evolutionary algorithms is impossible without introducing extra mechanisms. Group selection theory in biology is a good candidate as it explains the evolution of cooperation in nature. The authors referred the works of Wilson and Sober and the model by Traulsen and Nowak. The authors in this paper carried out investigations on the works of Wilson & Sober and Traulsen & Nowak. The investigations are conducted in the context of the n-player prisoner's dilemma (NPD). The NPD game offers a straightforward way of thinking about the tension between the individual and group level selection. Three evolutionary algorithms adapting the two models were designed and examined under different parameter settings; these parameters refer to group size, fraction of cooperators and selection pressure, and they directly affect the selection dynamics. In Trauslen's model, the group or individual to be eliminated is randomly selected. The third algorithm was the extension of Trauslen's group selection model where the group or individual to be eliminated is selected inversely proportional to its fitness. The results show that the algorithm which extends Traulsen's model is more robust towards parameter changes than the algorithms implementing the original Wilson and Traulsen models, because it is able to maintain high between-group variance which is able to ride individual selection arisen by the parameter changes.

2.2.2 Rethinking Multilevel Selection in Genetic Programming

The authors in this paper aimed to improve the capability of Genetic programming to tackle the evolution of cooperation: evolving multiple partial solutions that collaboratively solve structurally and functionally problems. The authors referred to the works of (Wu and Banzhaf, 2010, 2009)[8] and GP Teaming [28]. Genetic programming has proven to be an efficient and powerful problem-solving strategy. However, like all evolutionary algorithms, it is not a panacea; it has difficulty solving high dimensional, multimodal problems, which normally mean huge search spaces, or complicated fitness definitions, or expensive fitness evaluations. In this paper authors take a different approach to tackle the evolution of cooperation which is based on a computational multilevel selection framework [8]. This framework on one hand captures the gist of multilevel selection theory (MLS) [4, 6] in biology to encourage cooperation; on the other hand, it extends MLS to hierarchically create solutions for complex problems from simple subcomponents. The authors tested the applicability of this approach on 7 multi-class classification problems with different features, such as non-linearity, skewed data distribution and large feature space. The applicability of MLGP is verified on multi-class classification problems, in which 7 benchmark datasets with different data features, such as non-linearity, skewed data distribution, and large feature space were tested.

The authors claimed that the results, when compared to other cooperative evolutionary algorithms in the literature, demonstrate that this approach improves solution accuracy and consistency, and simplifies solution complexity. In the future, the authors plan to study the evolutionary transition by MLGP, and its potential applications.

2.2.3 A Hierarchical Cooperative Evolutionary Algorithm

The authors developed a novel cooperative evolutionary algorithm based on a new computational multilevel selection framework to successfully search multiple coadaptive subcomponents in a solution. The proposed algorithm constructs cooperative solutions hierarchically by implementing the idea of group selection. The authors referred to two well- known extensions of classic EAs i.e. Cooperative Co-Evolutionary Algorithms (CCEAs) [44] and Individual Evolutionary Algorithms (IEAs) [43]. Both algorithms are used for evolving cooperative solutions and both algorithms address, in different ways, the issues of problem decomposition, interdependencies between subcomponents, credit assignment, and the maintenance of diversity, which according to Potter and de Jong [45] are essential in cooperative EAs. However, the algorithm lack flexibility in determining the structure of solutions. In both cases, optimization is defined on individuals but not on collaborations. The author proposes a novel algorithm which constructs cooperative solutions hierarchically with the help of the group selection model proposed by Traulsen et al. [6]. In this work the authors investigate and compare the proposed algorithm on string covering problems whose fitness landscapes have multiple equal or unequal fitness peaks. Based on the experiments, the authors claimed that their algorithm improves both solution accuracy and evolutionary speed. In addition, the authors also said that the structure of a solution and the roles played by their subcomponents emerge as a result of evolution, rather than being designed by hand. In the future work the authors plan to study the evolutionary dynamics of this algorithm further to tackle real-world problems that require a substantial degree of cooperation.

2.2.4 Coevolution of cooperation and layer selection strategies in multiplex networks

The authors Hayashi et. al [21] in their work referred to the work of Gardens et. al [17], vWang et. al [50] and Buldyrev et. al [7]. Authors in their model have developed a co-evolutionary model of cooperation and layer selection strategies. Gardens et .al in their worked found that the evolution of cooperation was facilitated by the multiplex structure of networks only when the temptation to defect was large [17]. In the model of authors, each individual has a layer selection strategy and prisoner's dilemma game (PDG) strategy (cooperate or defect). Each individual plays PD with the neighbor in its layer or in another layer in which it wants to move. If the fitness of the neighbor is better than the individual, then the individual imitates its neighbor strategy. The imitation probability is linear to the difference between the fitness values. If the individual fitness is higher than its neighbor than the individual keeps its strategy or else imitates its neighbor strategy. Schematic image of the authors model is depicted in figure 2.6 [21].

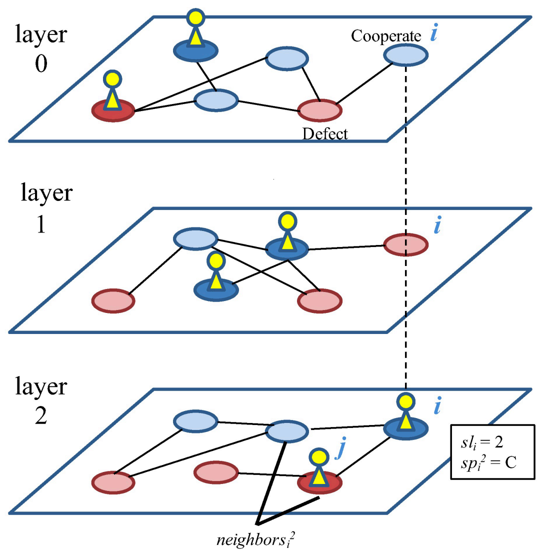

Figure 2.3: Schemetic image of model

The authors evaluated their work by having 100 individuals in the population, M =1, 3,., 19 layers and b=1.1, ., 2.1 which is the temptation to defect. The experiments were done for five trials for each combination of layers and the temptation to defect. The authors claimed from their experiment results that the proportion of cooperative strategies has increased with increasing the number of layers and is not dependent on the degree of the dilemma. Also, the increase in cooperative strategies which is caused due to the cyclic coevolution process of layer selection strategies and game theory strategies.

2.3 Evolution of Cooperation

2.3.1 Evolution of Cooperation by Multilevel Selection.

The success of cooperation is witnessed at all levels of biological organization. The authors have proposed a theory of group selection which has been unpopular for a long time. The works of Bowles, S & Ginitis, H (2004), Kerr. B & Godfrey-Smith, P. (2002), Fletcher, J & Zwick, M (2004) have been referred by the authors. The evolutionary biologists such as John Maynard Smith argued that natural selection acted primarily at the level of the individual. They followed the majority of biologists that group selection did not occur, other than in special situations such as the haplodiploid social insects like honeybees where kin selection was possible. The authors say that the competition between groups encourages cooperation. The population here is dispersed only once at the beginning of the process after that the groups are kept isolated. The entire evolutionary dynamics used in the model proposed by the authors are driven by individual fitness. Only the individuals are assigned the payoff values. Only the individuals reproduce. Groups can stay together or split/divide when reaching a certain size.

They derived a fundamental condition for the evolution of cooperation by group selection: if $b/c > 1 + n/m$, then group selection favors cooperation. The parameters *b* and *c* denote the benefit and cost of the altruistic act, whereas *n* and *m* denote the maximum group size and the number of groups. By proposing a minimalist model of multilevel selection they showed that the selection favors cooperators and opposes defectors. The model can be extended to more than two levels of selection and to include migration. This paper is cited 662 times. And the model proposed by the authors have been used by different authors in different fields.

2.4 Multi-Population Cultural Algorithms

2.4.1 A multi-population cultural algorithm for the electrical generator scheduling problem.

The authors were the first to introduce the Multi-Population Cultural Algorithm. They used MPCA to solve the electrical generator scheduling problem. The authors referred to the work of Mendes et al. and proposed a guided local search (GLS) based parallel cultural algorithm which is a hybrid algorithm of GA and GLS procedure. The authors proposed algorithm which is called Parallel Co-operating Cultural Algorithm (PARCA). They were the first to introduce multi-population cultural algorithm. The proposed algorithm is called Parallel Co-operating Cultural Algorithm (PARCA) in which the CAs were executed concurrently by the search programs. In this, the network of workstations was divided into two processors: a master processor and a slave processor. The master processor was in charge of initializing the population, managing the 23 population, performing the selection, mutation, and recombination. The slave processor was used to evaluate their simulations dispatched by the master processor. The population was divided into several sub-populations and were isolated from each other and managed their own local CA. The exchange of information between the populations allowed them to co-operate and to explore the promising areas of the search space, and also to reintroduce the previously lost genetic materials in the population. The populations also exchange their best individuals to enhance the search in the space. The architecture of PARCA is shown in figure below.

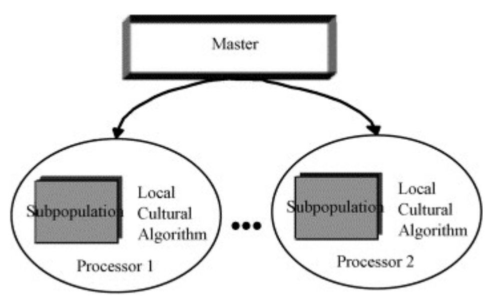

Figure 2.4: PARCA Model

The authors implemented the PARCA using the message passing interface (MPI) standard. The configurations of their system were: SGI Origin 200 and 6 Pentium (P5/100 MHz) cluster with interconnection through Ethernet (100 MB/s). According to the authors, the algorithm showed better results of optimization but the cost and execution time was slightly more than the existing algorithms at that time. Authors Digalakis and Margaritis were the first to introduce the Multi-Population Cultural Algorithm. They used MPCA to solve the electrical generator scheduling problem. The paper is cited 52 times.

2.4.2 Heterogeneous Multi-Population Cultural Algorithm.

According to the authors, the evolutionary algorithms were successfully applied to solve the optimization problems, but the issue with them was they had good chances of immature convergence and falling into local optima. The major reason behind this was they were not able to preserve diversity among the population over the course of generations. Author Raessi et. al stated that a group of sub-population which consists of different cultural algorithm do not directly communicate with each other so to overcome this problem the MPCA was introduced. Their work was inspired from the work of Digalakis et. al, Holland et al. , Koza et. al and Reynolds. The major reason behind this was they were not able to preserve diversity among the population over the course of generations.

The authors proposed a new framework of MPCA in which the subpopulations remained same, but the optimization parameters were divided among the subpopulations. Each sub-population optimized their parameters, and a set of the partial solution was generated. These partial solutions were combined to make the whole solution later. A detailed figure of the proposed architecture is depicted in figure. The Heterogeneous Multi-Population Cultural Algorithm was implemented using the JAVA platform by the authors. In their experiments, the population size was 1000 with 30 subpopulations, and each subpopulation had 33 individuals. The experiments were carried out for 10000 generations and 10 iterations. CEC 2012 benchmark problems were used to test the proposed algorithm, and the experiments were carried out on 8 functions. The authors were successful in getting minimum results on 7 out of the 8 functions. The minimum value of only one function was not found over the given time.

The authors explained that only one functions minimum value was not found over 10000 generations for five subpopulations, but they could have found it if the number of generations was increased. The authors claimed to find the minimum values of the numerical optimization functions and also their model was efficient in both time and space complexity. This paper is cited 12 times. The authors claimed to find the minimum values of the numerical optimization functions and also their model was efficient in both time and space complexity.

2.5 Conclusion

From works mentioned above, we can see that the Evolutionary Algorithms work efficiently for the optimization problems. MPCA, in particular, is effective for a dynamic population with multiple cultures. Incorporating Multilevel Selection strategy MPCA has shown better convergence rate than the traditional MPCA. By using the biological group selection theory the MPCA can provide us with better results when implemented to optimization problems.

Chapter 3

Proposed Approach

In this chapter, we will introduce the pseudo-code and framework of our proposed algorithm (multilevel selection framework). We will also discuss the design, belief space and population space used in our algorithm in detail. Later we will introduce multilevel selection theory and explain our proposed method.

3.1 Multilevel Selection Framework

The concept of multilevel selection is very simple, the different levels are like "Russian matryoshka dolls" (Wilson and Wilson, 2008) nested one within another. In our multilevel selection framework the population of individuals is subdivided into groups. The number of groups remain constant. The individuals of the group only interact with the members of the same group. Each group contains at least one individual. Each individual is assigned one strategy either cooperate or defect. Selection operates within groups and between groups. Within group selection favours individual with higher fitness in the group. Individuals therefore, compete within group. Between-group selection evaluates the performance of the group by seeing individuals of which group cooperate the best and selects that group.

The entire evolutionary dynamics is driven by individual fitness. Only the individuals in a group can reproduce. Groups can stay together or split or divide when reaching a maximum size. Groups that contain fitter individual reach maximum size faster and therefore they split more often. This concept leads to selection among groups, although only individuals reproduce in the population. These are the two levels of selection that we will study here.

3.2 Multilevel Selection in Multi-Population Cultural Algorithm (ML-MPCA)

Initialization : The line 1 and 2 of algorithm randomly initializing a Population *P* with *N* individuals. Each individual is assigned a unique ID. Individuals become the lowest level in the hierarchical structure. The population *P* is then divided into *m* groups. Here *m* groups can be seen as Local CAs with respect of MPCA. Individuals here are competitive with each other without being aware of collaborative goals.

Evolution on individual level: Individual and group fitness is evaluated of the generated population. Line 5-8: in every generation up to N' offspring's are reproduced. To select a parent individual for reproduction, a group has to be selected first based on fitness, from which an individual is selected with uniform probability. In our framework we are using roulette wheel selection in within group selection. Crossover exchanges randomly selected program segments between two parents, while mutation copies, deletes, adds, swaps, and changes instructions in an individual's program with predefined independent probabilities. The offspring is then added to group *gn.*

The pseudo-code of the algorithm is described below:

Algorithm 1: A Multi-Level Cooperative Multi-Population Cultural Algorithm

```
1.  P – Generate Initial Population (N,r);
2.  P' – Divide Population equally among groups (P,m);
3.  while population does not converge or max generation is not reached do
      4.  Evaluate Individual and group fitness;
      5.  for i=0 to N' do
            6.  gn ← Select_group(P');
            7.  Reproduce_offspring (gn);
            8.  put_back_to_group (idv, gn);
            9.  if group_size (gn) > n then
                  10.  split_group (gn);
                  11.  remove_group();
            12.  Update Local and Global Belief Space;
      13.  end
14.  end
```

Evolution on group level: Groups do not reproduce, they just split into two. In line 9-11 The algorithm will check if the group *gn* has reached its predefined size (n), if yes then the group *gn* will split into two daughter groups. We keep a constant number of individuals in the population, simply because individuals are the most basic building blocks. To maintain the constant number of groups in the population once the group *gn* splits, the group with the lowest fitness will be removed from the group. The best individuals from the population will be used to update the local and global belief space. The above steps will be repeated until a predefined termination criterion is reached, e.g. the maximum generation, a desired fitness or accuracy.

In summary, we proposed a novel ML-MPCA; this algorithm extends classic EAs by introducing group selection and the evolution on group levels. Group selection favors individuals who cooperate and contribute in a group. The evolution on group levels optimizes groups, which in turn should accelerate evolution on the individual level. We expect the algorithm will evolve faster and find better solutions.

Chapter 4

Experiments

In this chapter, we will firstly introduce the benchmark optimization functions used for evaluation of our algorithms. Then we describe the details of the experimental setup and fitness function. Later we will summarize the results and analyze it.

4.1 Benchmark Optimization Functions

Most commonly used benchmark optimization functions are used to evaluate our Algorithm and to compare it with the already existing algorithms. We have used CEC 2015 expensive benchmark functions which contain 15 functions. All the Functions used are minimal functions, so we are looking to find the minimum results. Some functions are non-convex, and some are convex. All the test functions are Dimension wise scalable. For our experiments, we have used different types of Functions like:

1. Unimodal functions
2. Simple multimodal functions

3. Hybrid functions

4. Composite functions

4.1.1 Unimodal Functions

The functions below are extension of the basic functions. Few functions are shifted and rotated.

$$o_{i1} = [o_{i1}, o_{i2}, o_{i3,..}, o_{iD}]^T \tag{4.1}$$

is the shifted global optimum, which is randomly distributed in [-80, 80]D. Each below function has shift data for CEC'15. All the test functions are shifted to o and scalable.

F1 (Rotated Bent Cigar Function): Rotated bent cigar function is extended from the bent cigar function. The function is featured as unimodal, non-separable and dimension-wise scalable. As seen from Figure 4.1 [31] it has smooth but narrow bridge.

$$f(x_1 \ldots x_n) = f_1(M(x - o_1)) + 100 \tag{4.2}$$

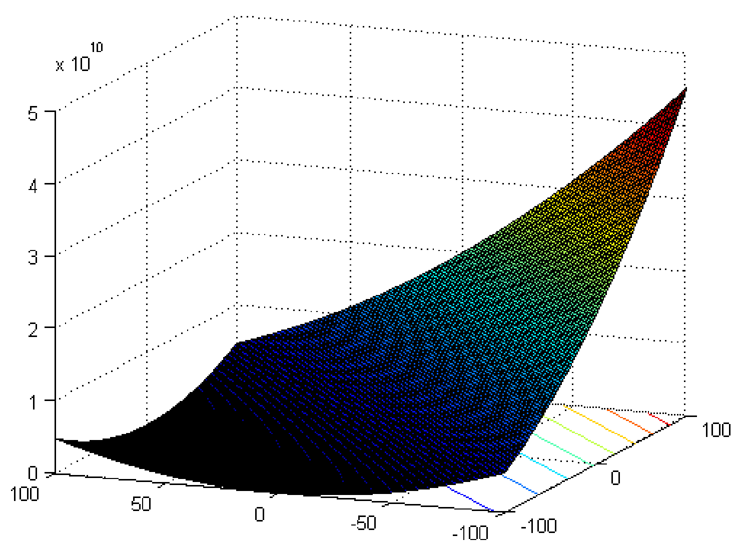

Figure 4.1: 3-D Map for Rotated Bent Cigar Function

F2 (Rotated Discus Function): Rotated discus function is extended from discus function. It featured as unimodal, non-separable and dimension-wise scalable. As depicted in Figure 4.2 [31] the function has one sensitive direction.

$$f(x_1 \ldots x_n) = f_2(M(x - o_2)) + 200 \qquad (4.3)$$

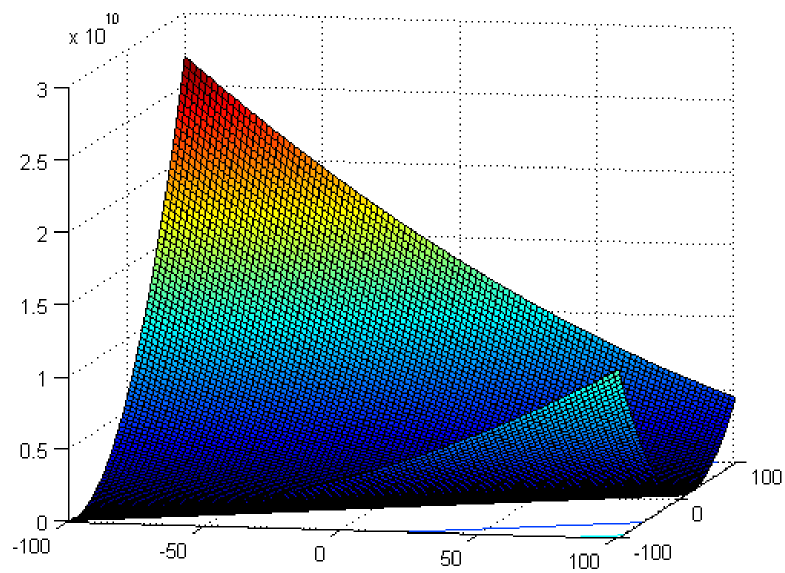

Figure 4.2: 3-D Map for Rotated Discuss Function

4.1.2 Simple Multimodal Functions

F3(Shifted and Rotated Weierstrass Function): The shifted and rotated weierstrass function is an extension of weierstrass function. It is featured as multi-modal, non-separable and dimension-wise scalable. As depicted in figure the function is continuous and differentiable only on a set of points**.**

$$f(x_1 \ldots x_n) = f_3\left(M\left(\frac{0.5(x - o_3)}{100}\right)\right) + 300$$

F4(Shifted and Rotated Schwefel's Function): The shifted and rotated schwefel's function is extension of schwefel's function. It is featured as multi-modal, non-separable and dimension-wise scalable. As seen from the figure 4.3 [31] the function has a lot of local optima and the second-best local optima is far from the global optima.

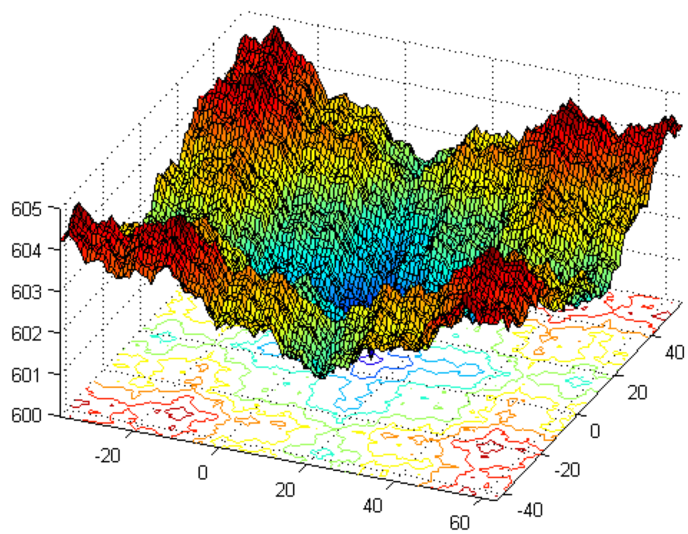

Figure 4.3: 3-D map for Shifted Rotated Schwefel's Function

$$f(x_1 \ldots x_n) = f_4\left(M\left(\frac{1000(x - o_4)}{100}\right)\right) + 400$$

F5(Shifted and Rotated Katsuura Function): The shifted and rotated katsuura function is an extension of katsuura function. It is featured as multi-modal, non-separable and dimension-wise scalable. It is seen in the figure 4.4 [31] that the function is continuous everywhere and it is not differentiable anywhere.

$$f(x_1 \ldots x_n) = f_5\left(M\left(\frac{5(x - o_5)}{100}\right)\right) + 500$$

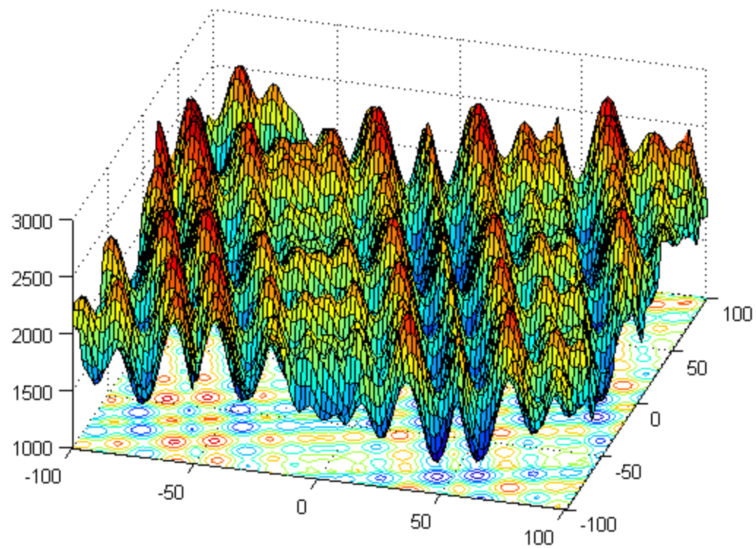

Figure 4.4: 3-D map for Shifted and Rotated Katsuura Function

F6(Shifted and Rotated HappyCat Function): The shifted and rotated happycat function is an extension of happycat function. It is featured as mutli-modal, separable and dimension-wise scalable.

$$f(x_1 \ldots x_n) = f_6\left(M\left(\frac{5(x - o_6)}{100}\right)\right) + 600$$

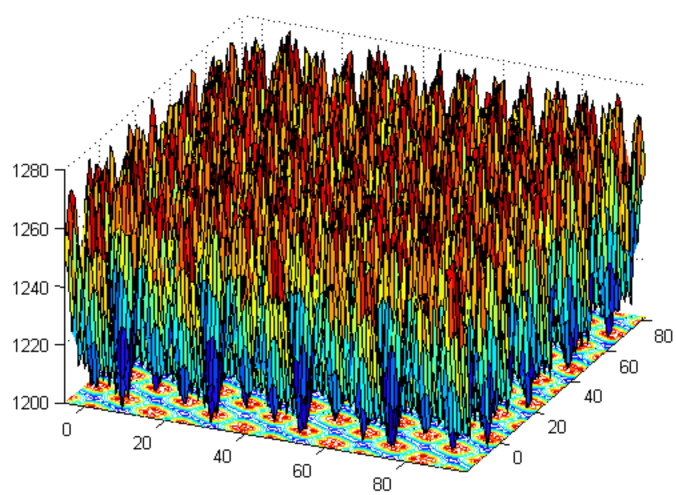

Figure 4.5: 3-D map for Shifted and Rotated HappyCat Function

F7(Shifted and Rotated HGBat Function): The shifted and rotated HGBat function is an extenion of HGBat function. It is featured as multi-modal, non-separable and dimension-wise scalable.

$$f\,(x_1 \ldots x_n) = f_7\,(M(\frac{5(x - o_7)}{100})) + 700$$

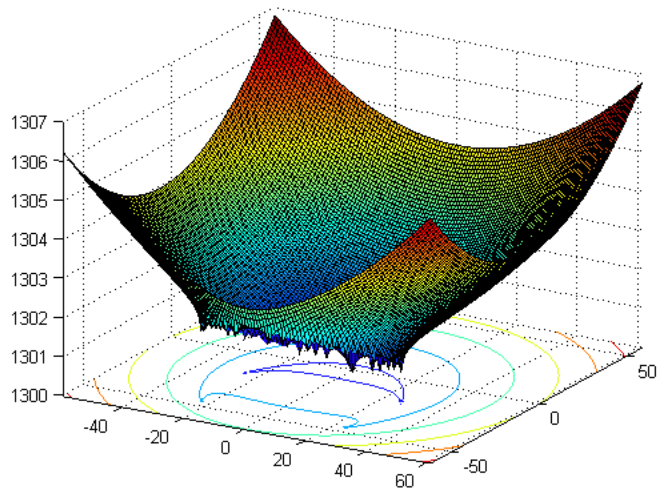

Figure 4.6: D map for Shifted and Rotated HGBat Function

F8(Shifted and Rotated Expanded Griewank's plus Rosenbrock's Function): The function is an extension and expanded version of two functions: Griewank's and Rosenbrock's function. The function is multi-modal, non-separable and dimension-wise scalable.

$$f\,(x_1 \ldots x_n) = f_8\,(M(\frac{5(x - o_8)}{100})+1) + 800$$

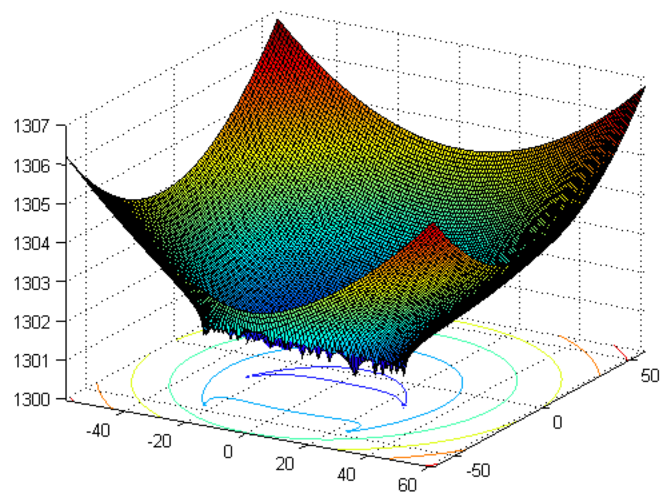

Figure 4.7: 3-D map for Shifted and Rotated Expanded Griewank's plus Rosenbrock's Function

F9(Shifted and Rotated Expanded Scaffer's F6 Function): Shifted and rotated expanded scaffer's F6 function is an extension of expanded scaffer's F6 function. It is featured as multi-modal, non-separable and dimension-wise scalable.

$$f\,(x_1 \ldots x_n) \;=\; f_9\,(M(x - o_9\,)+1) \,+\, 900$$

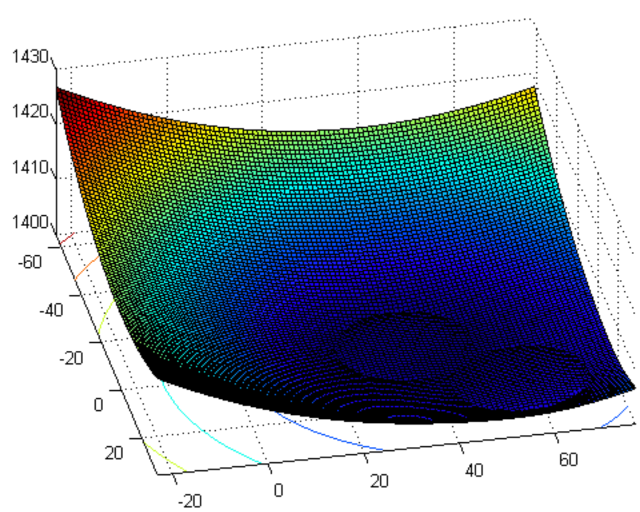

Figure 4.8: 3-D map for Shifted and Rotated Expanded Scaffer's F6 Function

4.1.3 Hybrid Functions

The hybrid functions are inspired from the real-world optimization problems. As in real-world optimization problems, different subset of variables possesses different properties. Similarly, in hybrid functions, the variables are divided randomly into some subsets and each subset will have different basic functions operating on them.

$$F(x) = g_1(M_1 z_1) + g_2(M_2 z_2) + \ldots + g_N(M_N z_N) + f^*(x)$$

$f^*(15) = 1000$

$f^*(16) = 1100$

$f^*(17) = 1200$

$F(x)$: *hybrid function*

$g_1(x)$: i^{th} *basic function used to construct the hybrid function*

N: number of functions

$$z = [z_1, z_2, \ldots, z_N];\ z1 = [y_{S_1}, y_{S_2}, \ldots, y_{S_m}],\ z2 = [y_{S_m+1}, y_{S_m+2}, \ldots, y_{S_m+n2}], \ldots,$$

$$z_N = [y_{S_{\sum_{i=1}^{N-1} n_i+1}} + y_{S_{\sum_{i=1}^{N-1} n_i+2}}, \ldots, y_{S_D}]$$

where, $y = x - o_i$ and S = *randperm(1:D)*

p_i : used to control the percentage of $g_i(x)$

n_i : dimension for each basic function $\sum_{i=1}^{N} n_i = D$

$$n_1 = [p_1 D],\ n_2 = [p_2 D],\ \ldots,\ n_{N-1} = [p_{N-1} D];\ nN = D - \sum_{i=1}^{N-1} n_i$$

F10(Hybrid Function 1) (N=3)

p=[0.3,0.3,0.4]

g_1 : Modified Schwefel's Function

g_2 : Rastrigin's Function

g_3 : High Conditioned Elliptic Function

F11(Hybrid Function 2) (N=4)

p=[0.2,0.2,0.3,0.3]

g_1 : Griewank's Function

g_2 : Weierstrass Function

g_3 : Rosenbrock's Function

g_4 : Scaffer's F6 Function

F12(Hybrid Function 3) (N=5)

p=[0.1,0.2,0.2,0.2,0.3]

g_1 : Katsuura Function

g_2 : HappyCat Function

g_3 : Expanded Griewank's plus Rosenbrock's Function

g_4 : Modified Schwefel's Function

g_5 : Ackley's Function

4.1.4 Composite Functions

$$F(x) = \sum_{i=1}^{N} w_i * [\lambda_i g_i(x) + bias_i] + f^*$$

f*(18) = 1300

f*(19) = 1400

f*(20) = 1500

F(x) : composition function

g_i: i^{th} basic function used to construct the composition function

N: number of basic function

o_i : new shifted optimum position for each $g_i(x)$, defines the global and local optima's position

$bias_i$: defines which optimum is global optimum

σ_i : used to control each $g_i(x)$'s coverage range, a small σ_i give a narrow range for that $g_i(x)$

λ_i : used to control each $g_i(x)$'s height

W_i : weight value for each $g_i(x)$, calculated as below:

$$w_i = \frac{1}{\sqrt{\Sigma_{J=1}^{D}}} exp\left(\frac{-\Sigma_{J=1}^{D}(x_j - o_{i_j})^2}{2D\sigma_i^2}\right)$$

Then normalize the weight $\omega_i = w_i / \sum_{i=1}^{n} w_i$

So when $x = o_i$, $\omega_j = \begin{Bmatrix} 1 & j = i \\ 0 & j \neq i \end{Bmatrix}$

for j= 1, 2, ..., N, $f(x) = bias_i + f^*$

The optimum which has the smallest bias value is the global optimum. The composition function merges the properties of the sub-function better and maintains continuity around the global/local optima.

F13(Composition Function 1) (N=5)

N = 5

σ = [10, 20, 30, 40, 50]

λ = [1, 1e-6, 1e-26, 1e-6, 1e-6]

bias = [0, 100, 200, 300, 400]

g_1 : Rotated Rosenbrock's Function

g_2 : High Conditioned Elliptic Function

g_3 : Rotated Bent Cigar Function

g_4 : Rotated Discus Function

g_5 : High Conditioned Elliptic Function

The function is featured as multi-modal, non-separable, asymmetrical and dimension-wise scalable. The function has different properties around different local optima.

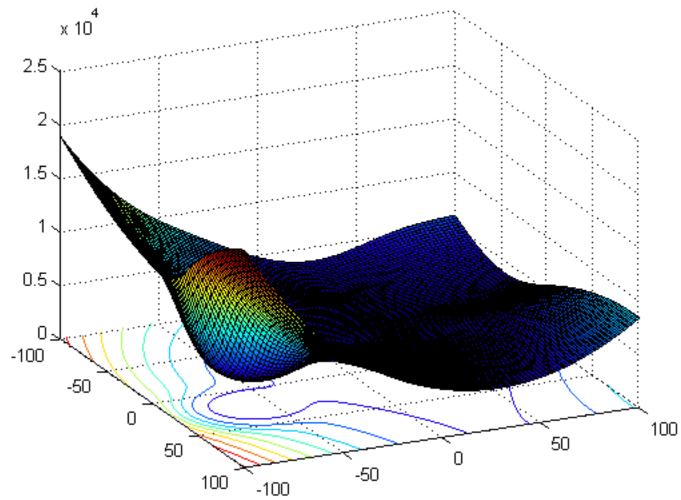

Figure 4.9: 3-D map for Composition Function

F14(Composition Function 2) (N=3)

N = 3

σ = [10, 20, 30]

λ = [0.25, 1, 1e-7]

bias = [0, 100, 200]

g_1 : Rotated Schwefel's Function

g_2 : Rotated Rastrigin's Function

g_3 : Rotated High Conditioned Elliptic Function

The function is featured as multi-modal, non-separable, asymmetrical and dimension-wise scalable. The function has different properties around different local optima.

F15(Composition Function 3) (N=5)

N = 5

σ = [10, 10, 30, 40, 50]

λ = [10, 10, 2.5, 2.5, 1e-6]

bias = [0, 100, 200, 300, 400]

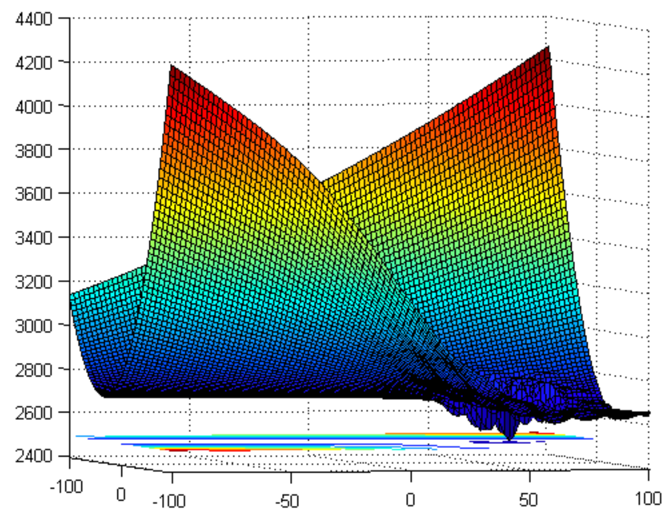

Figure 4.10: 3-D map for Composition Function

g_1 : Rotated HGBat Function

g_2 : Rotated Rastrigin's Function

g_3 : Rotated Schwefel's Function

g_4 : Rotated Weierstrass Function

g_5 : Rotated High Conditioned Elliptic Function

The function is featured as multi-modal, non-separable, asymmetrical and dimension-wise scalable. The function has different properties around different local optima.

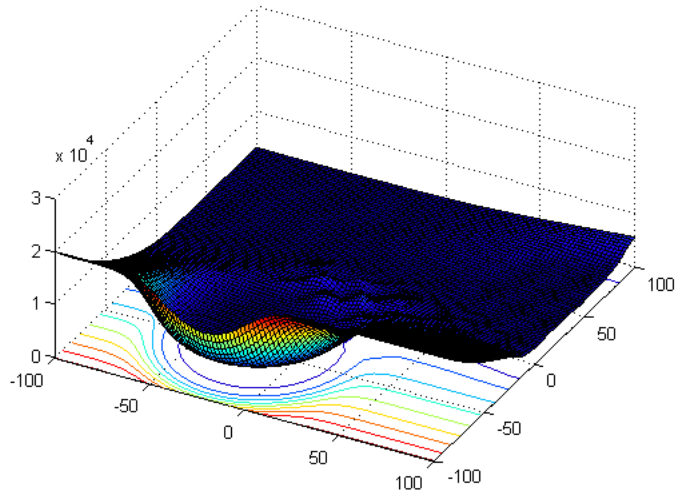

Figure 4.11: 3-D map for Composition Function 3

4.2 Study the Effects of Cooperation using (NPD) Problem

Our study aims to investigate the performance of proposed model in extending ML-MPCA to evolve cooperation. The investigation is conducted in the context of the n-player prisoner's dilemma (NPD). The NPD game offers a straightforward way of thinking about the tension between the individual and group level selection [26]; meanwhile it represents many cooperative situations in which fitness depends on both individual and group behavior. In this game, N individuals are randomly divided into m groups. Individuals in a group independently choose to be a cooperator or a defector without knowing the choice of others. The fitness function of cooperators (fC_i (x)) and defectors (fD_i (x)) in group i are specified by the following equations:[26]

$$fC_i\ (x) = base + \omega\ (\ \frac{b\ (\ n_i q_i - 1)}{n_i - 1}\ -c\ )$$

$$fd_i(x) = base + \omega\ \ \frac{b\ n_i q_i}{n_i\ - 1}$$

where base is the base fitness of cooperators and defectors; qi the fraction of cooperators in group i; ni the size of group i; b and c are the benefit and cost caused by the altruistic act, respectively; ω is a coefficient. Evidently, cooperators have a lower fitness than defectors, because they not only pay a direct cost, but also receive benefits from fewer cooperators than defectors do. The fitness of group i is defined as the average individual fitness. Although defectors dominate cooperators inside a group, groups with more cooperators have a higher group fitness. Hence, the dynamics between individual and group selection will drive the game in different directions.

The investigations focus on the effects caused by different group size n, initial fraction of cooperators r, and coefficient ω. Parameters n and r affect the as- sortment between cooperators and defectors in groups, and coefficient ω affects the individual and group fitness; both cause changes in selection dynamics.

To focus on the selection dynamics, we assume asexual reproduction without the interference of mutation[7]. A roulette wheel selection is adopted in the reproduction step for all algorithms. Parameters that are common to all experiments are set as follows: runs R = 20, generation gen = 500, population size N = 200, base fitness base = 10, benefit b = 5, cost c = 1.

For each algorithm, we measure:

Success Ratio by the number of runs whose population converges to cooperators to the number of total runs 20. The larger the ratio, the more likely an algorithm favors cooperation.

4.2.1 Effects of group size and initial fraction of cooperators

Here we study how our proposed algorithms behave under different group sizes. We set r (fraction of cooperators) = 0.5 and $\omega = 1$. Group size n is varied from {5, 10, 20, 50}. The success ratio and average variance ratio for each setting are listed in the tables below:

	r = 0.1	**r = 0.3**	**r = 0.5**	**r = 0.8**
n	**Success Ratio**	**Success Ratio**	**Success Ratio**	**Success Ratio**
5	1.00	1	1	1
10	1.00	1	1	1
20	0.35	0.65	1	1
50	0.15	0.25	0.30	0.40

Table 1 The effects of group size 'n' and initial fraction of cooperators 'r'

	r = 0.1	**r = 0.3**	**r = 0.5**	**r = 0.8**
n	**Success Ratio**	**Success Ratio**	**Success Ratio**	**Success Ratio**
5	1.00	1.00	1.00	1.00
10	1.00	1.00	1.00	1.00
20	0.25	0.55	0.80	0.90
50	0.00	0.00	0.00	0.10

Table 2 Wilson's group selection model [7]

	r = 0.1	r = 0.3	r = 0.5	r = 0.8
n	**Success Ratio**	**Success Ratio**	**Success Ratio**	**Success Ratio**
5	0.70	0.95	1.00	1.00
10	0.20	0.55	0.85	1.00
20	0.10	0.25	0.65	0.55
50	0.05	0.10	0.15	0.30

Table 3 Traulsen's group selection model [7]

It can be seen from the results as 'n' increases evolving cooperation becomes difficult. Our proposed algorithm converges to cooperators except n =50.

Further adjustments were made in the fraction of cooperators (r) from r = 0.1 to 0.3 to 0.5 and 0.8. When r drops the number of cooperators assigned to groups is smaller. Due to this influence of individual selection increases in the group.

The results on our proposed algorithm show that when n is small (5 or 10) due to strong group selection effects, the decrease of r does not affect the success ratio, but slows convergence speed towards cooperation.

For larger groups as n increases (group selection is weaker) and r decreases (individual selection is stronger), group selection can hardly dominate individual selection so it becomes difficult for our algorithm to preserve cooperation.

It can be seen from the results that our proposed model is robust to parameter changes than Wilson's and Trauslen's model[7].

4.2.3 Effect of coefficient ω to adjust the selection pressure

Here we use coefficient ω to adjust the selection pressure. If ω is small, the selection is called weak selection or else it is called strong selection. We will compare our proposed method (M1) with the Wilson's group selection model (M2).

We test our algorithm with r = 0.5 and ω set to {0.1,0.5,1,2,5,10} on all group sizes and note down the success ratio.

n	ω =0.1		ω =0.5		ω =1	
	M2	**M1**	**M2**	**M1**	**M2**	**M1**
5	1.00	1.00	1.00	1.00	1.00	1.00
10	1.00	1.00	1.00	1.00	1.00	1.00
20	0.90	0.95	1.00	1.00	0.80	1.00
50	0.40	0.45	0.80	0.65	0.10	0.80
n	**ω =2**		**ω =5**		**ω =10**	
	M2	**M1**	**M2**	**M1**	**M2**	**M1**
5	1.00	1.00	0.80	1.00	0.00	1.00
10	1.00	1.00	0.40	1.00	0.00	1.00
20	0.20	1.00	0.00	0.50	0.00	0.80
50	0.00	0.50	0.00	0.40	0.00	0.10

Table 4 The performance of algorithms under strong and weak selection

The above results show that the performance of our algorithm increases and then decreases as the ω value small to high or we can say when the selection pressure goes from weak to strong.

The results show that our proposed algorithm can successfully preserve cooperation under both strong and weak selection when the group size n is small (n = 5 or 10).

It can be noted from the experiments that smaller group sizes promote cooperation. And as the group size increase it is difficult to evolve cooperation. To evolve cooperation in larger groups extra mechanisms need to be developed. The results shows that our proposed algorithm can be used to evolve multiple partial solutions that can collaboratively solve structurally and functionally complex problems. And our proposed model shows better results than the Wilson's and Trauslen's models and quiet similar results on Traulsen's extended model[7].

4.3 Experimental Setup

Here, we have performed the experiments to compare the performance of MPCA, GA, CCGA[29] and ML-MPCA.

1. Genetic Algorithm
2. Cooperative Co-evolutionary Genetic Algorithm
3. Multi-population Cultural Algorithm
4. Multilevel cooperative Multi-population Cultural Algorithm

Simulations were performed to observe the performance of the proposed algorithm for finding optimal solutions. The 4 algorithms listed above are compared with each. We have used DE as our evolutionary algorithm in all algorithms with some changes in ML-MPCA. In ML-MPCA we are doing within group selection and between group selection as discussed in chapter 3 in our proposed method. To carry out a fair comparison, the parameters used for execution of all the algorithm are same. The values of the parameters are listed in the Table . All the algorithm are tested 20 times individual on all the fitness functions to get an accurate solution.

The performance of the algorithm was done using the following criteria:

Parameters	Values
Size of population	100
Number of subpopulation	10
Size of subpopulation	10
Maximum number of generations	100
Dimensions	10 & 30
Independent run times	20
C_r: Crossover probability	0.5
F: scaler factor	[0.5,2.5]

Figure 4.12: Algorithm Parameters

- Mean fitness value (Mean): mean value of the solutions got at the maximum generation in 100 runs.
- Standard deviation (Std.): standard deviation of the mean fitness.

4.3 Results and Analysis

In this section we will compare all the proposed strategy with MPCA,GA and CCGA. The comparisons are done on both low dimension (10D) and high dimension (30D) on all the benchmark problems mentioned in section 4.1.

Table 5 Results on 10D & 30D benchmark functions

Function	Dimension	GA	CCGA	MPCA	ML-MPCA
F1	**10**	1.12E+11 (7.12E+09)	2.33E+10 (1.04E+10)	1.44E+09 (3.15E+08)	**3.56E+05** (2.67E+04)
	30	3.12E+11 1.05E+10	4.14E+10 2.14E+09	3.18E+10 4.99E+09	**1.87E+06** 3.18E+08
F2	**10**	6.76E+10 7.65E+09	3.46E+09 2.32E+08	1.22E+05 9.47E+04	**3.58E+04** 7.75E+05
	30	7.67E+10 1.29E+10	4.21E+08 2.23E+08	**2.16E+05** 7.50E+04	4.21E+06 7.75E+05
F3	**10**	3.19E+02 0.743	3.15E+02 0.765	**3.10E+02** 0.712	4.78E+02 0.572
	30	3.59E+02 1.359	3.49E+02 1.564	3.43E+02 1.277	**1.02E+02** 1.222
F4	**10**	5.06E+03 2.01E+02	4.34E+03 1.03E+03	**3.13E+02** 0.793	1.03E+03 1.45E+02
	30	1.23E+04 5.32E+02	2.23E+04 1.45E+04	5.57E+03 2.53E+02	**4.45E+03** 3.33E+02
F5	**10**	5.15E+02 0.654	4.54E+02 0.324	5.02E+02 0.218	**1.02E+02** 1.326
	30	5.15E+02	5.27E+02	5.04E+02	**2.24E+02**

		0.396	0.689	0.457	0.257
F6	**10**	6.16E+02 2.077	6.34E+02 2.182	**6.01E+02** 0.192	1.46E+03 1.11
	30	6.11E+02 0.946	8.32E+02 1.385	6.04E+02 6.342	**2.34E+02** 5.453
F7	**10**	1.48E+03 5.60E+01	1.45E+03 3.37E+02	**7.11E+02** 2.97E+01	2.29E+03 1.20E+02
	30	1.28E+03 3.55E+01	2.24E+03 3.32E+03	**7.74E+02** 1.10E+01	3.56E+03 1.02E+02
F8	**10**	7.79E+09 2.43E+09	6.54E+08 2.54E+08	4.83E+03 3.67E+03	**2.25E+02** 1.57E+02
	30	1.11E+12 4.67E+11	3.45E+11 2.56E+10	1.41E+07 6.34E+06	**1.10E+04** 2.45E+03
F9	**10**	9.05E+02 0.029	8.09E+02 0.209	9.04E+02 0.113	**1.455E+01** 0.158
	30	9.15E+02 0.020	1.03E+03 0.028	**9.15E+02** 0.155	9.93E+02 0.112
F10	**10**	4.83E+10 4.63E+09	2.67E+09 1.29E+09	**1.01E+06** 4.11E+05	2.43E+07 1.36E+07
	30	7.54E+10 8.84E+09	2.42E+09 1.29E+09	7.06E+07 1.98E+07	**2.21E+06** 1.25E+06

F11	**10**	4.70E+03 2.62E+02	2.34E+03 3.43E+02	1.11E+03 2.73E+03	**1.45E+01** 1.24E+01
	30	1.20E+04 6.37E+02	1.04E+04 2.34E+04	1.33E+03 4.52E+01	**1.24E+03** 2.87E+02
F12	**10**	1.88E+10 1.07E+10	1.22E+09 1.04E+08	1.48E+03 5.73E+01	**8.02E+02** 1.77E+01
	30	7.51E+11 3.58E+11	2.67E+10 2.31E+10	9.48E+03 7.65E+03	**1.05E+03** 6.25E+02
F13	**10**	2.46E+04 1.62E+03	1.34E+04 1.03E+04	**1.66E+03** 1.53E+01	2.21E+03 1.01E+01
	30	4.11E+04 3.52E+03	2.54E+04 2.03E+04	**2.12E+03** 9.97E+01	3.42E+03 4.51E+01
F14	**10**	1.87E+04 1.16E+04	2.34E+04 2.02E+04	1.61E+03 1.61E+03	**1.23E+02** 2.77E+00
	30	8.22E+04 4.70E+03	2.33E+04 1.03E+04	1.78E+03 2.71E+01	**1.56E+02** 2.23E+02
F15	**10**	6.43E+04 4.28E+03	2.30E+04 2.32E+03	**1.98E+03** 1.12E+02	1.81E+04 1.77E+02
	30	1.60E+06 8.85E+03	2.34E+05 1.20E+04	2.91E+03 2.15E+01	**1.03E+03** 1.78E+02

Table 6 Computational complexity observed in ML-MPCA

Function	D = 10		D = 30		T_B/T_A
	T_1	$T_A=T_1/T_0$	T_1	$T_B=T_1/T_0$	
F1	0.0088	0.0767	0.0277	0.2415	3.148
F2	0.0103	0.0897	0.0281	0.2449	2.730
F3	0.0429	0.3740	0.3337	2.9090	**7.778**
F4	0.0106	0.0924	0.0307	0.2676	2.896
F5	0.0286	0.2493	0.2114	1.8431	**7.393**
F6	0.0077	0.0671	0.0248	0.2162	3.222
F7	0.0086	0.0749	0.0257	0.2240	2.990
F8	0.0099	0.0863	0.0286	0.2493	2.888
F9	0.0097	0.0845	0.0292	0.2546	3.013
F10	0.0091	0.0793	0.0294	0.2563	3.232
F11	0.0169	0.1473	0.0911	0.7942	**5.391**
F12	0.0120	0.1046	0.0492	0.4289	4.100
F13	0.0122	0.1064	0.0643	0.5606	5.268
F14	0.0113	0.0985	0.0550	0.4795	4.868
F15	0.0488	0.4254	0.3844	3.3573	**7.892**

The performance of an algorithm determines if its solution has better quality than the solution produced by the other methods for the same problem instance or not. Hence, the performance is an important measure in optimization problems. It determines the success probability of an algorithm. One would like to use the algorithm which produces the best solution. TABLE 5 presents mean and standard deviation measures for three algorithms on the 15 test functions. The mean and standard deviation show quality of the results obtained by each algorithm. To ease of observation, the best results obtained by the algorithms are shown in bold.

For 10 dimensional problems the proposed method have 8 out 15 better results and for 30 dimensional problems we have 11 out of 15 better results when compared to the existing algorithms. This results show that our proposed algorithm improves solution accuracy and consistency.

Furthermore, complexity of ML-MPCA on both 10D & 30D is evaluated following the guidelines provided in CEC'15 [13]. The value for T_0 has been calculated using the test program provided in the guidelines. The calculated computing time for the test program is $T_0 = 0.1147s$. The average complete computing time T_1 for all the benchmark functions is calculated. Finally the algorithm complexity T_1/T_0 has been measured. And the ML-MPCA complexity has been given in Table 6.

From Table 6, value of T_B/T_A equal to one shows the zero complexity from dimension 10 to 30 for the reported computationally expensive problems. The values which are greater than one represent the complexity of computational time using the ML-MPCA. F3 and F5 (multi-modal functions) have demonstrated higher complexity in terms of computational time and as expected, for the hybrid functions (F11 and F12) and composition functions (F13 –F15) have shown higher values due to their complication, especially for F15. Relative computationally expensive problems are highlighted in bold in Table 6. For the rest of functions the average T_B/T_A is almost equal to 3.0 which means the complexity for dimension 10 to 30 is increased for almost three times.

Chapter 6

Discussion

In our thesis we have demonstrated how to implement multilevel selection framework in MPCA to study the effects of cooperation, to evolve cooperating agents and also tested our model using the CEC 2015 expensive benchmark problems.

While studying the effects of cooperation with our proposed approach it can be noted from the experiments that smaller group sizes promote cooperation. And as the group size increase it is difficult to evolve cooperation. To evolve cooperation in larger groups extra mechanisms need to be developed. The results shows that our proposed algorithm can be used to evolve multiple partial solutions that can collaboratively solve structurally and functionally complex problems. And our proposed model shows better results than the Wilson's and Trauslen's models[7].

The optimization results obtained by the ML-MPCA are compared with the Genetic Algorithm, a variant of genetic algorithm i.e. cooperative co-evolutionary genetic algorithm and multi-population cultural algorithm. The results in Table 5 shows that our model gives better results on all the 30 Dimension hybrid and composition function. By observing Table 5, we can see that ML-MPCA could find the minimum average error in 8 and 11 out of 15 cases for dimensions 10 and 30, respectively. It shows that ML-MPCA provides better results on 30D computationally expensive problems.

However the following issues need to be given consideration before ML-MPCA is applied to new problems.

Evolutionary transition: Even though not mentioned in this paper, ML-MPCA has the potential to be extended to an evolutionary transition model, in which groups, depending on their levels, become a new complex organism functioning differently from their components. This model, we believe, will be useful to solve problems whose subcomponents have more complicated interactions, such as agents in multi-agent systems.

Group Fitness definition: Average individual performance and over- all group performance is necessary for the group fitness. Missing either of them will cause evolution to drift to suboptimal solutions[16].

Parameterization: The framework extends evolution to group levels; therefore, one needs to specify values for new parameters namely the cooperation, crossover and mutation rates for reproducing groups. And generally speaking, EAs with structured population need more parameters.

Chapter 7

Conclusion And Future Work

The primary focus here was to study the effect of cooperation using multilevel selection model in MPCA. The approach used in this paper is based on biological group selection theory. The results show that our proposed algorithm can successfully preserve cooperation under both strong and weak selection when the group size n is small (n = 5 or 10) compared to other models that are studied here. It can be noted from the experiments that smaller group sizes promote cooperation. And, as the group size increase it is difficult to evolve cooperation. It can be said that compared to previous multilevel selection models[3][6] our proposed algorithm is robust in response to parameter changes such as group size (n), fraction of cooperators (r) and selection pressure (ω).

We have used CEC 2015 expensive benchmark problems to evaluate the performance of our algorithm and compared them with the existing algorithms. The results depict that the proposed algorithm performs better with higher dimension than the lower dimension problems. For 10 dimensional problems the proposed method have 8 out 15 better results and for 30 dimensional problems we have 11 out of 15 better results when compared to the existing algorithms

The developed algorithm needs further improvements. Studying the reproduction on the group level by multilevel selection theory. By applying operators on the group level to select groups. In the future work the proposed ML-MPCA model can be extended to more than two levels of selection and can also include migration. The proposed model can also be designed to study the cultural group selection theory.

Bibliography

[1] Sarker, R., Mohammadian, M., & Yao, X. (Eds.). (2002). Evolutionary optimization (Vol. 48). Springer Science & Business Media.

[2] Nowak, M. A. (2006). Five rules for the evolution of cooperation. science, 314(5805), 1560-1563.

[3] Sober, E., & Wilson, D. S. (1999). Unto others: The evolution and psychology of unselfish behavior (No. 218). Harvard University Press.

[4] Wilson, D. S. (1975). A theory of group selection. Proceedings of the national academy of sciences, 72(1), 143-146.

[5] Wilson, D. S., & Wilson, E. O. (2008). Evolution" for the Good of the Group": The process known as group selection was once accepted unthinkingly, then was widely discredited; it's time for a more discriminating assessment. American Scientist, 96(5), 380-389.

[6] Traulsen, A., & Nowak, M. A. (2006). Evolution of cooperation by multilevel selection. Proceedings of the National Academy of Sciences, 103(29), 10952-10955.

[7] Wu, S. X., & Banzhaf, W. (2009, September). Investigations of Wilson’s and Traulsen’s group selection models in evolutionary computation. In European Conference on Artificial Life (pp. 1-9). Springer, Berlin, Heidelberg.

[8] Wu, S. X., & Banzhaf, W. (2010, July). A hierarchical cooperative evolutionary algorithm. In Proceedings of the 12th annual conference on Genetic and evolutionary computation (pp. 233-240). ACM.

[9] Wu, S. X., & Banzhaf, W. (2011). Evolutionary transition through a new multilevel selection model. In ECAL (pp. 874-881).

[10] Markvoort, A. J., Sinai, S., & Nowak, M. A. (2014). Computer simulations of cellular group selection reveal mechanism for sustaining cooperation. Journal of theoretical biology, 357, 123-133.

[11] P. A. Grudniewski and A. J. Sobey, “Multi-level selection genetic algorithm applied to cec’09 test instances,” in Evolutionary Computation(CEC), 2017 IEEE Congress on. IEEE, 2017, pp. 1613–1620.

[12] S. X. Wu and W. Banzhaf, “The use of computational intelligence in intrusion detection systems: A review,” Applied soft computing, vol. 10,no. 1, pp. 1–35, 2010.

[13] Q. Chen, B. Liu, Q. Zhang, J. J. Liang, P. N. Suganthan, B. Y. Qu, "Problem Definition and Evaluation Criteria for CEC 2015 Special Session and Competition on Bound Constrained Single-Objective Computationally Expensive Numerical Optimization", Technical Report, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Technical Report, Nanyang Technological University, Singapore, Nov 2014.

[14] P. Parikh, “Knowledge migration strategies for optimization of multi-population cultural algorithm,” Ph.D. dissertation, University of Windsor(Canada), 2017.

[15] B. Simon, J. A. Fletcher, and M. Doebeli, “Towards a general theory of group selection,” Evolution, vol. 67, no. 6, pp. 1561–1572, 2013.

[16] S. X. Wu and W. Banzhaf, “Rethinking multilevel selection in genetic programming,” in Proceedings of the 13th annual conference on Genetic and evolutionary computation. ACM, 2011, pp. 1403–1410.

[17] Reynolds, R. G. (1994, February). An introduction to cultural algorithms. In *Proceedings of the third annual conference on evolutionary programming* (pp. 131-139). River Edge, NJ: World Scientific.

[18] Ziad Kobti et al. Heterogeneous multi-population cultural algorithm. In Evolutionary Computation (CEC), 2013 IEEE Congress on, pages 292{299. IEEE, 2013.

[19] Andrew William Hlynka and Ziad Kobti. Knowledge sharing through agent migration with multi-population cultural algorithm. In FLAIRS Conference, 2013.

[20] Ziad Kobti, RG Reynold’s, and Tim Kohler. A multi-agent simulation using cultural algorithms: The effect of culture on the resilience of social systems. In Evolutionary Computation, 2003. CEC'03. The 2003 Congress on, volume 3, pages 1988{1995. IEEE, 2003.

[21] Jason G Digalakis and Konstantinos G Margaritis. A multi-population cultural algorithm for the electrical generator scheduling problem. Mathematics and Computers in Simulation, 60(3):293{301, 2002.

[22] Yi Nan Guo, Yuan Yuan Cao, and Dan Dan Liu. Multi-population multi-objective cultural algorithm. In Advanced Materials Research, volume 156, pages 52{55. Trans Tech Publ, 2011.

[23] Yi-nan Guo, Jian Cheng, Yuan-yuan Cao, and Yong Lin. A novel multi-population cultural algorithm adopting knowledge migration. Soft computing, 15(5):897{905, 2011.

[24] Andrew William Hlynka and Ziad Kobti. Knowledge sharing through agent migration with multi-population cultural algorithm. In FLAIRS Conference, 2013.

[25] Katsuki Hayashi, Reiji Suzuki, and Takaya Arita. Coevolution of cooperation and layer selection strategy in multiplex networks. Games, 7(4):34, 2016.

[26] J. A. Fletcher and M. Zwick. N-player prisoner's dilemma in multiple groups : A model of multilevel selection. In E. Boudreau and C. Maley, editors, Proceedings of the Artificial Life VII Workshops, 2000.

[27] J. A. Fletcher and M. Zwick. The evolution of altruism: Game theory in multilevel selection and inclusive fitness. Journal of Theoretical Biology, 245:26–36, 2007.

[28] M. Brameier and W. Banzhaf. Evolving teams of predictors with linear genetic programming. Genetic Programming and Evolvable Machines, 2(4):381–407, December 2001.

[29] Potter, M. A., & De Jong, K. A. (1994, October). A cooperative coevolutionary approach to function optimization. In International Conference on Parallel Problem Solving from Nature (pp. 249-257). Springer, Berlin, Heidelberg.

[30] Santosh Upadhyayula. Dominance in multi-population cultural algorithms. 2015.

[31] Robert G Reynolds and SM Saleem. The impact of environmental dynamics on cultural emergence. Perspectives on Adaptions in Natural and Artificial Systems, pages 253{280, 2005.

[32] Zhen Wang, Attila Szolnoki, and Matjaz Perc. Evolution of public cooperation on interdependent networks: The impact of biased utility functions. EPL (Europhysics Letters), 97(4):48001, 2012.

[33] Chang, M., Ohkura, K., Ueda, K., & Sugiyama, M. (2003, December). Group selection and its application to constrained evolutionary optimization. In Evolutionary Computation, 2003. CEC'03. The 2003 Congress on (Vol. 1, pp. 684-691). IEEE.

[34] Scheuring, I. (2009). Evolution of generous cooperative norms by cultural group selection. Journal of theoretical biology, 257(3), 397-407.

[35] Sober, E., & Wilson, D. S. (2000). Summary of: ‘Unto others. The evolution and psychology of unselfish behavior’. Journal of Consciousness Studies, 7(1-2), 185-206.

[36] Lichocki, P., Floreano, D., & Keller, L. (2014). Selection methods regulate evolution of cooperation in digital evolution. Journal of The Royal Society Interface, 11(90), 20130743.

[37] Akbari, R., & Ziarati, K. (2011). A multilevel evolutionary algorithm for optimizing numerical functions. International Journal of Industrial Engineering Computations, 2(2), 419-430.

[38] Yang, Z., Tang, K., & Yao, X. (2008, June). Multilevel cooperative coevolution for large scale optimization. In Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence). IEEE Congress on (pp. 1663-1670). IEEE.

[39] Panth Parikh. Knowledge migration strategies for optimization of multi-population cultural algorithm. 2017.

[40] Agoston E Eiben, James E Smith, et al. Introduction to evolutionary computing, volume 53. Springer, 2003.

[41] JH Holland. Adaptation in natural and artificial systems. 1 edigao. Ann Arbor, USA: The University of Michigan Press, 1975.

[42] J Alami, A El Imrani, and A Bouroumi. A multipopulation cultural algorithm using fuzzy clustering. Applied Soft Computing, 7(2):506{519, 2007.

[43] P. Collet, E. Lutton, F. Raynal, and M. Schoenauer. Individual GP: an alternative viewpoint for the resolution of complex problems. In W. Banzhaf, J. Daida, and et al., editors, Proceedings of the 1st Genetic and Evolutionary Computation Conference (GECCO '99), volume 2, pages 974–981, Orlando, Florida, USA, 1999. Morgan Kaufmann.

[44] M. A. Potter and K. A. de Jong. A cooperative coevolutionary approach to function optimization. In Y. Davidor, H.-P. Schwefel, and R. Ma¨nner, editors, Proceedings of the 3rd International Conference on Parallel Problem Solving from Nature (PPSN III), Jerusalem, Israel, volume 866/1994 of LNCS, pages 249–257. Springer Berlin/Heidelberg, 1994.

[45] M. A. Potter and K. A. de Jong. Cooperative coevolution: An architecture for evolving coadapted subcomponents. Evolutionary Computation, 8(1):1–29, 2000.

Vita Auctoris

NAME	:	Dilpreet Singh
PLACE OF BIRTH	:	Bilaspur, U.P, India
YEAR OF BIRTH	:	1993
EDUCATION	:	New Horizon College(VTU), Bangalore, India
		Bachelor of Engineering, Computer Science
		2011-2015
		University of Windsor, Windsor ON, Canada
		Master of Science, Computer Science 2017-2018