Chapter 5

COMMUNITY DETECTION USING DIFFERENTIAL EVOLUTION ALGORITHM

5.1. Introduction

In this chapter, we explain our research work on the DE algorithm. In previous work are totally based on the Genetic Algorithm. After done the experiment on GA, we feel Genetic algorithm is most experimental evolutionary algorithm in the world of social network and many other research areas also. We think about the DE because that is also a very influential algorithm in the evolutionary category.

Evolutionary algorithms are well known optimization approaches to cope with non-linear complex problem. These population based algorithm, however suffer from a general weakness. They are computationally expensive due to slow nature of the evolutionary process. This experiment present some novel schemes to accelerate converge of evolutionary algorithm. In this experiment, we have done the best utilization of the DE algorithm for the community detection in social network on different conditions and choose the best DE version of that particular problem. In this experiment, we have modified the objective function. We replace the modularity function with the help of some other 7 objective functions.

After this experiment, we have done another work on the DE algorithm that is called the vertex similarity based DE algorithm. We have improved the DE with the help of node similarity concept in initialization of population phase. We choose improved population in the algorithm then surely improve our results.

After this experiment, we have done another research work based on the DE algorithm. We employed the opposition based learning concept in the initialization phase of the algorithm and tournament method for the selection of the population. We have created a three new combination of the DE algorithm and compare the simple DE algorithm. Modified DE version such as TDE (Tournament based DE), OBDE (Opposition concept based DE), TOBDE (Tournament and Opposition learning based DE) and compare with the SDE (Simple Differential evolution algorithm). Although the concept of opposition has an old history in other field and sciences, this is the first time that it contribute to enhance an optimization. The proposed opposition based DE (OBDE) employ opposition based optimization (OBO) for population initialization and also for generation jumping. In this work, appositives numbers have been utilized to improve the convergence rate of classical DE. A test with some datasets and benchmark function is employed for experiments verification. The contribution of the opposite numbers is empirically verified.

5.2. DE with Multiple Objective Functions

In this Chapter, different types of experiments are executed on Differential evolution algorithm and checked their performance. The algorithm is modified without changing the internal architecture. However, after amendment, the whole behavior of the algorithm is changed. Fitness function plays the key role in the every algorithm because fitness function is checked productivity and evaluates the every iteration in the whole procedure. In every domain and area or work, optimization techniques cannot be imagined without fitness function. It is concluded that some different work has been evaluated which is based on the objective function and fitness function. So we have chosen that the multiple fitness functions are used as a substitute of modularity in the differential evolutionary algorithm. There are some options i.e. conductance, internal density, Average degree, Normalized cut, Expansion, cut ratio as a fitness function. In the next section, the procedure of our algorithm is discussed.

In Differential evolution algorithm, input dataset in the form of adjacency matrix and some other input parameters are given below in Table 5.1, Encoding matrix representing the community partition of a given dataset. The algorithm runs through the maximum number of iteration N_{max}.

Parameter	Value	Description
μ	0.35	Threshold value for clean-up operation
P_n	100	Number of individuals in population
P_c	0.4	Ratio of crossover individuals to total no. of individuals of population
P_m	0.95	Ration of mutation individuals to total no. of individuals of population
N_{max}	100	Maximum number of iterations

Table 5.1: Parameter of experiments

5.2.1. Experimental Description

The algorithms discussed so far have been analyzed by a sequence of experiments. The experiment conducted on Microsoft Windows 7 Professional operating system using a MATLAB 11 programming platform with Intel (R) Core-i7 3.40 GHz processor and 4.0 GB RAM. The values of different parameters P_n, P_c, P_m, and Nmax like were fixed through a sequence of experiments for which proposed work in the given scenario work at best. Parameter values so set been shown in Table 5.1, the values of P_n, P_c, P_m, and N_{max} could be altered as appropriate for the situation. Performance of DE algorithm with multiple fitness function is tested on four real networks, and is compared with substitute fitness function with DE Algorithm (Jia, et al., 2012). While evolving to DE with multiple fitness function from simple DE algorithm, we have used the 6 fitness functions replacing the Modularity function and create the new name of algorithms with other fitness functions i.e. given below in Table 5.2. Four datasets used for experimentation are Strike (Michael, 1997), Zachary's karate club (Zachary, 1977), Dolphin sociality (Lusseau, et al., 2003) and American College Football (Evans, 2012). The specifications of each dataset have been summarized in Table 5.3.

Algorithm	Colour	Combinations
AG1	Green	DE with Conductance
AG2	Red	DE with Internal Density
AG3	Blue	DE with Average Degree
AG4	Magenta	DE with Normalize Cut
AG5	Cyan	DE with Expansion
AG6	Black	DE with Cut Ratio
AG7	Yellow	DE with Modularity

Table 5.2: Colour code of Algorithm for experiments

Datasets	Number of nodes	Number of edges	Number of communities
Strike	24	34	3
karate club	34	78	2
Dolphin sociality	62	159	2
American college football	115	613	12

Table 5.3: Details of real world datasets

5.3. Experiments Perform and Results Discussion

In this work, we have conducted a range of experiments for optimizing the results of community detection in social networks. In this proposed work, we have a tendency to use the differential evolutionary algorithmic program. The DE algorithmic program evolves from the genetic algorithmic program, wherever the modularity performs has been used as fitness performs. Main plan behind of those experiments is to judge completely different fitness function's performance. We have used the Average degree; Normalized cut, internal density, Expansion, Conductance, cut ratio as fitness functions.

We performed DE with multiple fitness function on four well known datasets i.e. Strike [, Zachary karate club, Dolphin sociality, American College Football. We assessed the performance of DE with multiple fitness based various algorithms on the basis of NMI because in that experiments modularity utilized as fitness function. Main purpose behind this experiment is whether we can check the nature and variation of the performance of Differential evolution algorithm. We can also be verified, which fitness function between 6 is replaced the standard modularity function and whether check which fitness function is given the better results.

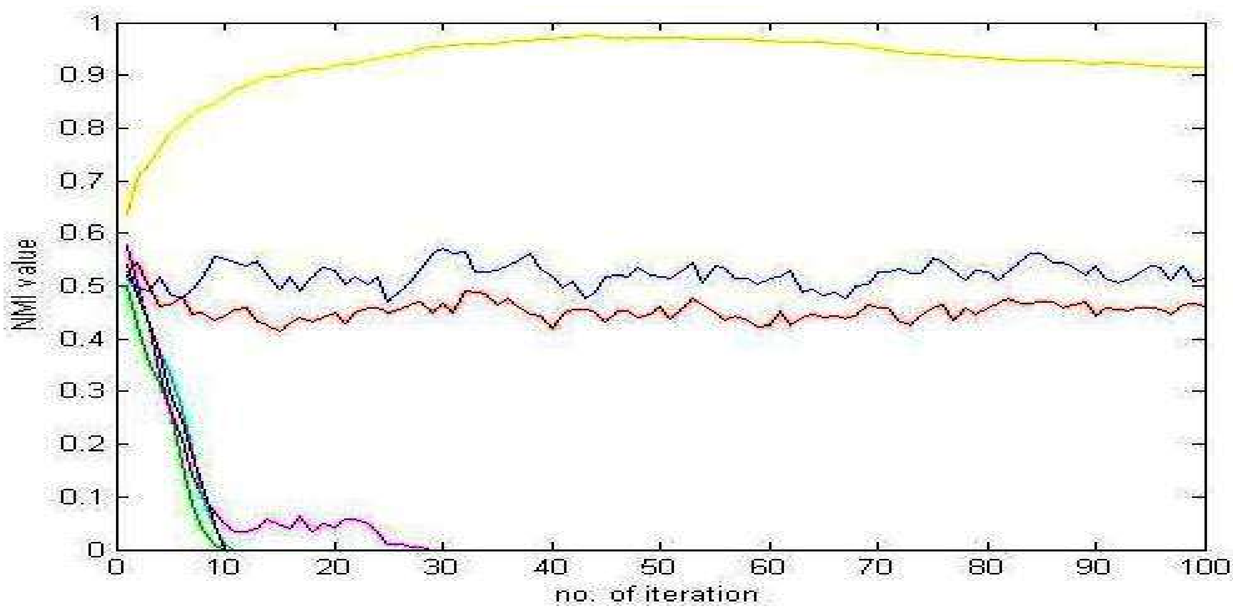

Figure 5.1 NMI values represent with number of iterations using Strike Dataset

Figure 1 represents the performance and nature of diverse version of DE with multiple fitness function algorithms. From Figure 5.1, It is observed that the AG-7 means DE with modularity combination raises the (McDaid, et al., 2011) NMI value = 0.9236. Then the AG-3 have got the NMI value =0.5324. Similarly, DE with internal density (AG-2) has got the NMI value=0.4654. These three variations of the DE have good performances as compared to other remaining concepts i.e. AG-1, AG-4, AG-5, AG-6. On the moderate/average platform, AG-5, AG-6, AG-1 and AG-4 have worst performance. The graph is represented for the strike datasets and whole analysis is based on this graph. So it can be said that the DE is good for very small dataset because DE with modularity performs best on this dataset. Similarly, others such as DE+ average degree, internal density are also suitable for small datasets. Expansion, cut ratio, conductance are very worst perform. Finally, it is concluded that (AG-7, AG-3, AG-2) fitness functions with DE algorithm can be used on small datasets.

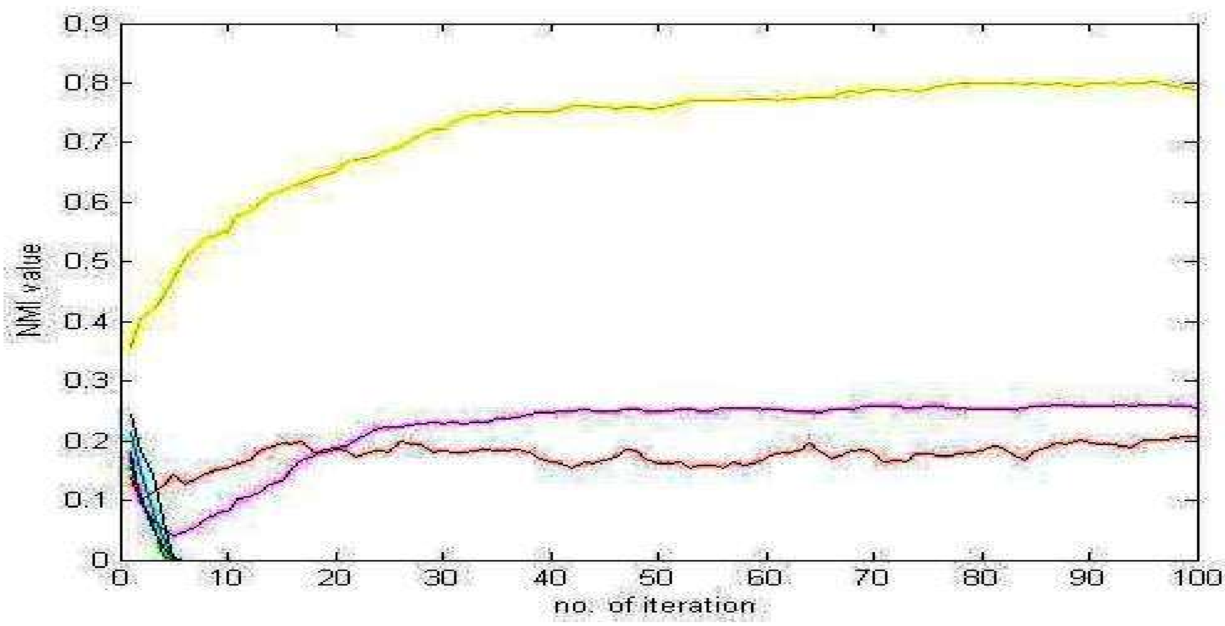

Figure 5.2 NMI values represent with number of iterations using Karate club Dataset

Figure 5.2 represents the performance and nature of various version of DE with various fitness function algorithm. Figure 5.2, it is observed that the AG-7 means DE with modularity combination raises the NMI value =0.7965. Then the AG-4 have got the NMI value =0.2675. Similarly, DE with internal density (AG-2) has got the NMI value =0.2134. These three variations of DE have better performance as compared to other variations for the karate club dataset. AG-6, AG-3, AG-5 and AG-1 have shown the worst performance for this dataset. After the analysis of above graph, it is said that the standard DE means the use of modularity as a fitness function which is suitable for the small datasets such as strike and karate. Another analysis is also available for DE with the internal density which is also suitable for the small datasets as after modularity it is the best for the small datasets. Finally, it is concluded that the DE with modularity shows the best performance for small datasets. Another DE (normalize cut and internal density) is also good for small datasets, but expansion, conductance, cut ratio and average degree shows worst performance for this dataset.

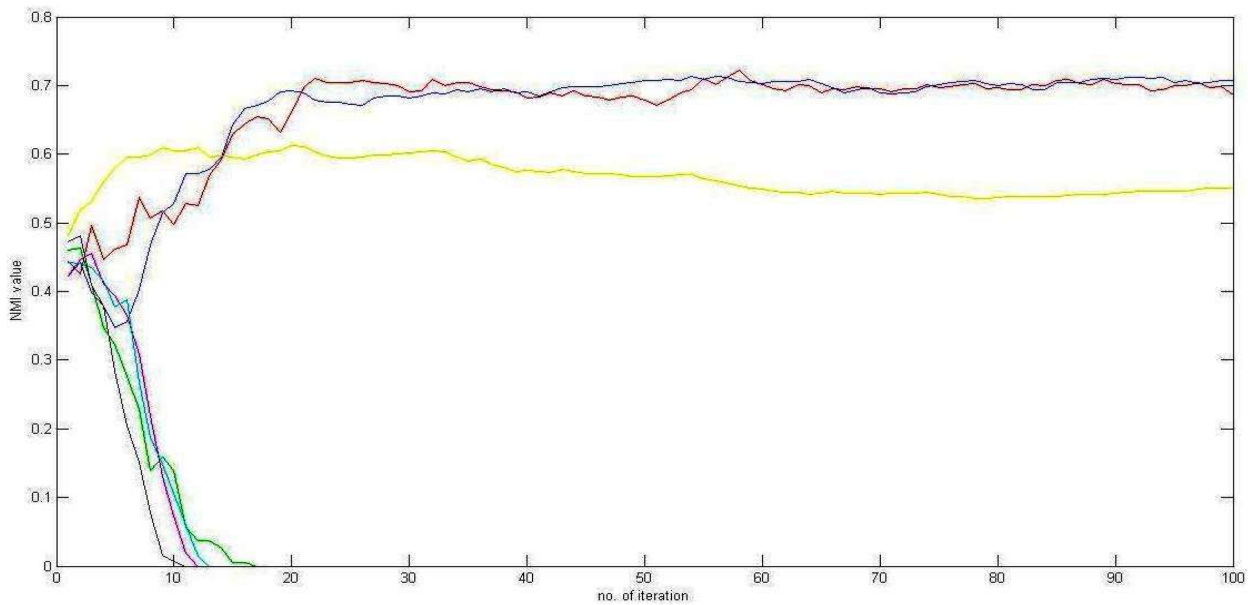

Figure 5.3 NMI values represent with number of iterations using Dolphin Dataset

Figure 5.3 represents the performance of the diverse version of DE. It is observed that the AG-3 means DE with the average degree has got the higher NMI value = 0.7122. Then the second higher value is AG-2 which has got the NMI value= 0.6985, this shows DE with internal density. The third one is AG-7 which has got the NMI value=0.5556. These three variations of the DE (AG-3, AG-2 & AG-7) perform better as compared to other variations. Remaining AG-1, AG-5, AG-4 and AG-6 show worst performance for the dolphin dataset. The result indicates that AG-7 is good but not the best for the average size data sets such as dolphin. AG-3 shows the best performance for the dolphin dataset and then AG-2 shows good performance.

Figure 5.4 represents the performance and nature of different version of DE algorithm. The graph represents that the AG-7 has got top position by performance and the NMI = 0.5452. Then the second one is AG-2 which means DE with internal density has got the NMI = 0.2916. Others have got the worst performance for this different version of DE for the football dataset.

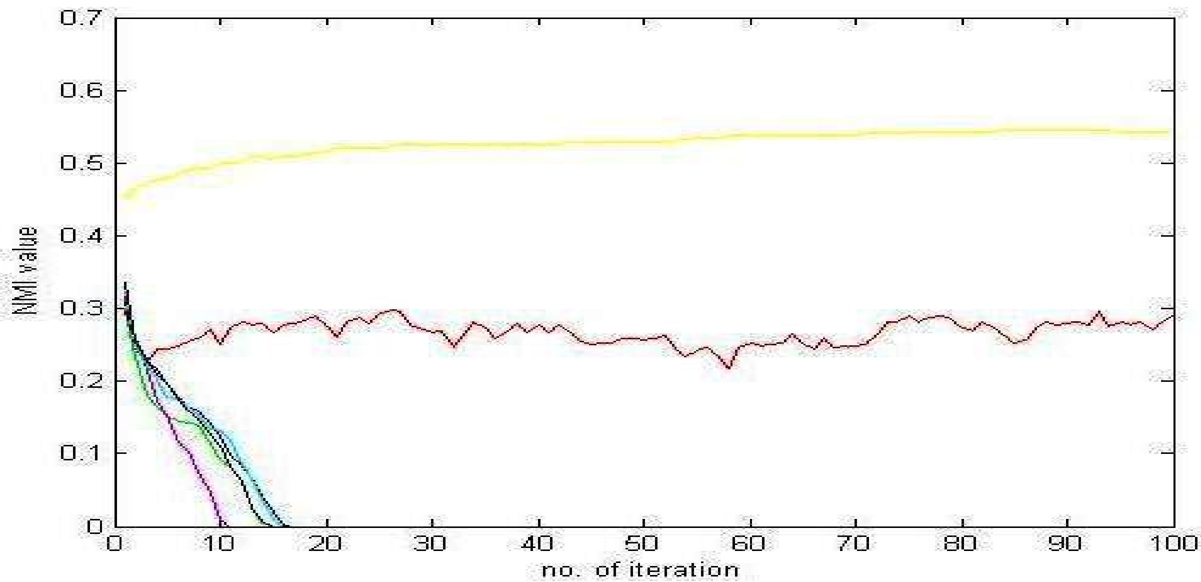

Figure 5.4 NMI values represent with number of iterations using Football team dataset

The performance of various versions of DE for four datasets is concluded. It is observed that AG-2 performs well for the whole dataset as a result represents that the DE with internal density always maintains the position in top 3 with every dataset. Similarly, AG-7 also maintains the position in all used datasets. NMI function measures the accuracy of algorithms. So the accuracy of DE can be shown with different variations. In the social network, quality and accuracy both are necessary metrics to measure the performance of algorithms. So the MCDM (Kou, et al., 2011) graph is used to check the performance of algorithm based on the quality and accuracy metrics. In this graph both quality and accuracy metrics are used, Discuss in detail in the next section.

5.3.1. Evaluation Method

Find the quality and accuracy of communities by any algorithm are very vital as important decisions depend upon these results. If incorrect communities are detected, then the applications which are connected to the community are affected. The accuracy of community is a lot of vital than quality connected problems. Hence, the analysis strategy is focused and inclined towards realization and the accuracy analysis of outcomes. We have got conjointly thought about quality measures to make sure about the degree of qualitative necessities. From these two measures, we have got thought about one more measure of qualitative accuracy. It has contained accuracy and quality at the similar time. For only either accuracy or quality measures, we have got evaluated the performance of algorithm through worth dependent comparative analysis. While the qualitative accuracy measure, we have got performed worth primarily dependent analysis likewise as a rank of algorithms. In general, the three methods are supported to calculate the performance of the proposed work. The impact and explanation of three evaluation methods are as given below:

5.3.1.1. Accuracy Measure

Community detection algorithmic rule aims to spot the variances inside the network and partitions network consequently. The division of a network is the combination of objects and associated links. Any inaccurate allocated node to some cluster can result in allot associated relations additionally to that cluster. Hence, those wrong allocated item can reason a forceful amendment in the framework read of communities and can guide to faulty explanations. An algorithmic rule should make sure that like as incorrect work will not happen or proportions of falsely assignments are less. Entire objects of the network have to imagine one by one for confirming the accuracy. Many accuracy measures are available which serve the same issue to judge foreseen communities with original communities. As delineated higher than ARI (Vinh, et al., 2010), NMI, Purity (Yang, et al., 2002), F-measure (Amigó, et al., 2009) and Entropy (Liu and Yu, 2005) area unit the measures that compare all objects of the network one by one with great solutions. In this approach calculated the accuracy of community detection in social network with the help of some well known real world network datasets.

5.3.1.2. Quality Measure

The quality measure concerns the cluster as an entire object within the cluster. It measures the practicability of clusters among the network structure. The possibility of a cluster is calculated by analyzing suitableness of vertexes belong the cluster based on their linkage with alternative group nodes and nodes outer the cluster. The effect on the number of individual nodes distribution to entirely different clusters in the network displayed with quality measures. The outcome of the community detection method based on the type of communities and organization of communities and find the quality measure value depend on it. The quality measure is incredibly useful in such cases so as to evaluate the performance of any algorithm. Four such quality metrics are considered such as Coverage (Cov) (Chockler, et al., 2003), Modularity (Q) (Newman, 2006), Number of communities (Com) (Tiebout, 1956) and Average Isolability (Isob) (Hu, et al., 2017). These are using datasets and cluster pattern for analysis of expected communities. All the data sets are considered which are employed of accuracy measure. As quality checker focuses additionally on the structure of the network and community structures instead of objects within the network thus need not bear in mind of original community structures.

5.3.1.3. Qualitative Measure

Quality and accuracy metrics are two different issues, and each has different significance. The Quality evaluation considers the network structure, whereas the accuracy compute considers exactness to calculate communities got with every algorithm. Our main aim is to focus on accuracy. In conjunction with this, a wise level of quality of the communities has to confirm. Hence, each these measures need to be balanced to get the combined essence of both. To realize such objective has to measure both accuracy and quality for the similar communities found with any algorithm. Most of the measures represented higher than for accuracy as well as quality may be considered for this purpose. However, those network data sets are solely found whose community combinations are well-known because accuracy measure needs real community structures. The quality measure does not have some limitation so any network data sets can be used.

All real world data sets considered for qualitative accuracy measure, which also employed in an accuracy measure. The analysis is focused in two directions. Primary, value primarily based analysis performed by considering all metrics. Secondary, we have a tendency to show Multiple Criterion decision making (MCDM) system is presented primarily based ranking accumulating all the accuracy metrics and quality metrics below one single score. The technique used for order preference by similarity to ideal answer (TOPSIS) developed by (Yoon and Hwang, 1995) for ranking the decision-making alternatives over multiple criteria. The TOPSIS method adopted which is demonstrated in Kou et al. (2014) for our ranking. TOPSIS has the rights to assign completely different weights to every criterion. Since we have to focus a lot of on accuracy, with this weight distribution mechanism, we will just apply much weight to accuracy metrics. The effectiveness of every algorithm is analyzed additionally with completely different accuracy levels.

5.3.2. MCDM Ranking Related Setting

The communities got by entirely different algorithms are evaluated for accuracy and quality. For analysis of the clusters supported accuracy and quality, quite different metrics of list has taken into consideration. The accuracy of clusters is a lot of necessary than quality connected matters. Therefore, analysis ways utilized in this work are a lot of inclined in the direction of realization and analysis of the effect of accuracy. F-measure, NMI, ARI, and Acc_avg are metrics used for accuracy whereas Modularity, Coverage, Average insolubility and Average no. of communities are used to quality measure. After that Multiple Criterion decision making (MCDM) dependent ranking is performed. The benefit of MCDM rank is to accumulate all quality metrics and accuracy metrics below single score. TOPSIS Methodology (Kou, et al., 2014) is used for MCDM ranking. TOPSIS methodology will allocate weights to every of the metric wherever the summary of all weights allotted to entirely different metrics should be one. The weight age assigned to every metric depends on the priority of that metric. During this work, the concern is about to realize a lot of accuracy in communities. Therefore 75% weight is allotted to accuracy metrics and 25% weight is distributed to quality metrics. As in, a weight of each measure is being circulated equally between the metrics in that class. To measuring the accuracy, four metrics (NMI, ARI, average accuracy and F-measure) are considered and to compute quality, conjointly four metric (Coverage, Modularity, Average Isolability and Average no. of communities) are deployed. Therefore 75% weight allotted to accuracy are distributed among four metrics assignment every metric with 18.75% weight. The metric for quality is equally assigned 25% weight that has been an assignment every metric is 6.25% for quality capacity.

5.3.2.1. Measuring Accuracy

Results obtained on given datasets are presented in Table 5.4-5.7; Strike dataset values are available in Table 5.4. So AG7 has got the highest value for all the accuracy functions. Then the AG2 has got the higher F-measure value. Similarly, AG3 has got the higher values of NMI, ARI and average accuracy as compared to other algorithms. For Karate dataset, AG7 has got the highest values for all the accuracy measures which are shown Table 5.5. AG4 has got the higher values for NMI, Avg_acc, and ARI. Similarly, AG3 has got the higher F-measure as compared the other algorithms. For Dolphin dataset, AG3 has got the highest F-measure value as compared to other algorithms. Similarly, AG2 has got the highest ARI, NMI & Avg_acc values as compared to other given algorithms which are shown in Table 5.6. For Football dataset, AG7 has got the highest values for all the accuracy metrics. Then AG2 algorithm has got the second position which is shown in Table 5.7 and which is for all the accuracy metrics. All the given datasets used and compared the table values. AG7 algorithm has performed the best for accuracy metrics. But in Dolphin dataset, AG7 is not performing well as compared to AG2 and Ag3. So finally, it is said that different parameters and diverse datasets have changed the performance of Multiple DE algorithms.

5.3.2.2. Measuring Quality

The result obtained from all the utilized datasets in term of quality metrics are presented in Table 5.4 - 5.7; For Strike dataset, all the algorithms show the similar number of communities except AG2 and AG7. The average Isolability and number of communities' metrics are got the higher values as compared to other algorithms. Equality to this fact, a greater number of communities is resulting in better to the average Isolability is also an important point. Similarly, AG6 is got the higher modularity metrics, Coverage for AG3 & average Isolability got higher the AG7 algorithm for strike dataset. For Karate club dataset, the AG7 algorithm has got the good quality metric value except for modularity (Q). Again important fact, a higher number of communities is resulting in better quality metric value. AG1 has obtained greater modularity value for this dataset. For dolphin society dataset, all the algorithms have contained the same number of communities except AG7. AG6 has provided the highest modularity (Q) value. Similarly, AG3 has got the highest coverage value. Average Isolability is highest for the AG2 algorithm. For the football dataset, the entire algorithms have a similar number of communities except for AG2 & AG7. If again notable thing, a higher value of some communities resulting better quality metric like as average Isolability for both AG2 and AG7. Modularity quality metric contained the highest value for the AG6 algorithm. Similarly, AG2 has got the highest value of Coverage metric.

5.3.2.3. Value based Analysis

For strike dataset, AG7, AG4, AG3, AG2 have shown the better accuracy measure values. However, at the similar time, AG7 and AG3 have shown the better quality measure values; since both case AG7 and AG3 show better metric values we can say AG7 & Ag3 best-performed compare to other algorithms. Similarly, for karate club dataset, AG7 and AG2 have

Algorithms	Accuracy_avg	NMI_avg	Fmea_avg	Ari_avg	Com_avg	Q_avg	Cov_avg	Isob_avg
AG1	0.3653	0.0216	0.3403	0.0163	2	0.2951	0.3118	0.0229
AG2	0.5886	0.4541	0.5857	0.254	3	0.2934	0.1124	0.5739
AG3	0.6901	0.5214	0.4641	0.4196	2	0.296	0.3913	0.5027
AG4	0.3717	0.0335	0.3467	0.0239	2	0.3043	0.318	0.0367
AG5	0.3693	0.0276	0.3397	0.0219	2	0.2997	0.315	0.028
AG6	0.3694	0.0272	0.3409	0.0215	2	0.3123	0.3138	0.0288
AG7	0.9548	0.9267	0.872	0.8971	4	0.3065	0.0919	0.6881

Table 5.4: DE with Multiple variations using Strike Dataset

shown the better quality and accuracy metric values it means AG7 and AG2 excellent more than all other algorithms. Finally, we can say that AG7 have the best performer for the small datasets like as karate club and strike datasets and second choice for the solution is both algorithms AG2 and AG3. It means AG7 is not last and best option for the small datasets; another good choice is also available. For dolphin society dataset, AG2 and AG3 have shown the better accuracy metric values as compared to other algorithms. Similarly, AG2, AG3, and AG6 have performed better

Algorithms	Accuracy_avg	NMI_avg	Fmea_avg	Ari_avg	Com_avg	Q_avg	Cov_avg	Isob_avg
AG1	0.4878	0.0033	0.4985	0.0022	2	0.1626	0.4501	0.0075
AG2	0.5318	0.1752	0.3809	0.0774	2	0.1483	0.1966	0.5562
AG3	0.4879	0.0035	0.4989	0.0023	2	0.1244	0.4454	0.0092
AG4	0.565	0.2204	0.3805	0.1403	2	0.1414	0.1183	0.4198
AG5	0.4882	0.0048	0.4952	0.0026	2	0.1327	0.4335	0.0135
AG6	0.4893	0.0064	0.4983	0.0049	2	0.1372	0.4355	0.0118
AG7	0.8724	0.7232	0.715	0.74	3	0.1413	0.4819	0.5938

Table 5.5: DE with Multiple variations using Karate Club Dataset

for quality metrics. In the both cases, AG2 & AG3 have shown better metric values. So AG2 and

Algorithms	Accuracy_avg	NMI_avg	Fmea_avg	Ari_avg	Com_avg	Q_avg	Cov_avg	Isob_avg
AG1	0.5592	0.0312	0.5094	0.0337	2	0.2507	0.1276	0.0218
AG2	0.8783	0.6663	0.8901	0.7527	2	0.2513	0.0556	0.9443
AG3	0.8773	0.6649	0.9138	0.7501	2	0.2706	0.6545	0.0294
AG4	0.5593	0.0325	0.5059	0.0345	2	0.2753	0.123	0.0246
AG5	0.56	0.0328	0.5074	0.0358	2	0.2735	0.1243	0.0238
AG6	0.5564	0.0248	0.5073	0.0272	2	0.2791	0.1344	0.0202
AG7	0.7469	0.5665	0.3793	0.5088	4	0.2637	0.1114	0.7497

Table 5.6: DE with Multiple variations using Dolphin Dataset

AG3 show the best performance over all other algorithms. Similarly, in football dataset, AG2 and AG7 have shown the improved accuracy metric values. At the similar time AG2, AG6 &AG7 have demonstrated better quality metric values. So in both cases, AG2 and AG7 are common means both algorithms have performed best for quality-wise and accuracy-wise metrics

Algorithms	Accuracy_avg	NMI_avg	Fmea_avg	Ari_avg	Com_avg	Q_avg	Cov_avg	Isob_avg
AG1	0.1097	0.0184	0.0855	0.006	2	0.2799	0.0657	0.0337
AG2	0.4823	0.2667	0.1147	0.0873	3	0.2788	0.1954	0.7529
AG3	0.1175	0.0227	0.0858	0.0078	2	0.2782	0.0553	0.0367
AG4	0.103	0.0146	0.085	0.005	2	0.2824	0.0778	0.0291
AG5	0.1162	0.0226	0.0864	0.0078	2	0.283	0.0574	0.0347
AG6	0.1142	0.0213	0.0864	0.0073	2	0.2874	0.0599	0.0361
AG7	0.7934	0.6939	0.1877	0.289	5	0.2786	0.0621	0.786

Table 5.7: DE with Multiple variations using Football Dataset

compare to other algorithms. Whereas, entirely regarding accuracy we have obtained algorithm AG2 is best and regarding quality algorithm AG7 is best. Now, the difficulty is a way to decide that one is best. This drawback occurs because higher accuracy metric value does not involve higher quality metric value and the other way around. We can simply choose higher communities if an algorithm generates each quality and accuracy metric values also higher or lower. If not, we have to swap quality and accuracy metric values so as to make a decision on the most effective communities.

5.3.3. MCDM Ranking

Figure 5.5 - 5.8 and Table 5.8 illustrate MCDM rankings got depending to clusters predicted by various algorithms in each data set. We have allotted additional weights to accuracy

DATASET	ALGORITHM	MCDM RANK
STRIKE	AG1	0.3524
	AG2	0.4555
	AG3	**0.7052**
	AG4	0.3695
	AG5	0.3621
	AG6	0.3628
	AG7	**0.6675**
KARATE	AG1	0.3792
	AG2	0.4168
	AG3	0.3795
	AG4	0.4373
	AG5	0.3796
	AG6	0.3800
	AG7	**0.6182**
DOLPHIN	AG1	0.4623
	AG2	**0.6633**
	AG3	**0.7119**
	AG4	0.4618
	AG5	0.4625
	AG6	0.4572
	AG7	0.3371
FOOTBALL	AG1	0.5550
	AG2	0.4162
	AG3	**0.5606**
	AG4	0.5502
	AG5	**0.5599**
	AG6	0.5585
	AG7	0.4368

Table 5.8: MCDM ranking score obtained with 75% accuracy and 25% quality, higher score indicates more inclination of algorithm towards accuracy.

measures, so scores indicate algorithm's inclination towards accuracy. Clearly, AG7 display higher scores for small datasets such as karate club and strike, which indicates AG7 generates extremely inclined communities towards accuracy for the small datasets. In Dolphin data set, AG3 obtained the top score because both quality and accuracy measures were higher, which was also observed in the earlier analysis. AG2 gets a minor score than the AG3, but the score is more impressive than rest of the algorithms. Similarly, for football dataset, AG3 has got the highest ranking score as compared to another algorithm. AG5 and AG6 are just near to AG3 ranking score, but the score is more significant compared with the remaining algorithms. AG1 and AG7 show an indigent inclination towards accuracy in most of the datasets means AG1 is the poor ranking score for the small dataset and AG7 for the large datasets and it is evident since their accuracy, as well as quality metric values, were poor. At last AG3 illustrate higher scores for all the datasets except karate club dataset. It indicates that the AG3 originates extremely inclined clusters towards accuracy.

MCDM ranking with a difference of percentage of accuracy contribution in the collected score is displayed in Figure 5.5-5.8. The algorithms AG1, AG5, and AG6, show approximately similar nature for every used data set. Firstly, when accuracy involvement is given weight 25%, AG2 and AG7 illustrate higher scores than AG3, which is evident as both these algorithms generated communities with low accuracy and relatively higher quality than AG3. While accuracy is involvement 25% weight in the rank score, clearly quality measure involvement becomes 75% so that quality measure attribute will have more influence on the rank score.

In this Experiment, we mainly focus on the DE algorithm and produce the new version of DE algorithms for different conditions. Quality and accuracy of the communities are verified and also test the sensibilities with the help of objective functions in the social network. We have done experiments on Differential evolution algorithm using various types of different objective functions and optimize the results of community detection for the maintained range of datasets. After performed the operation resultant show that DE is surely additional expectable for expansion, internal density, and average degree as a fitness function. Apparently, we found another better option as fitness functions for the DE algorithm, and we will provide a better choice to the users according to categories of datasets. We will choose a fitness function according to requirement and find the optimized results. .

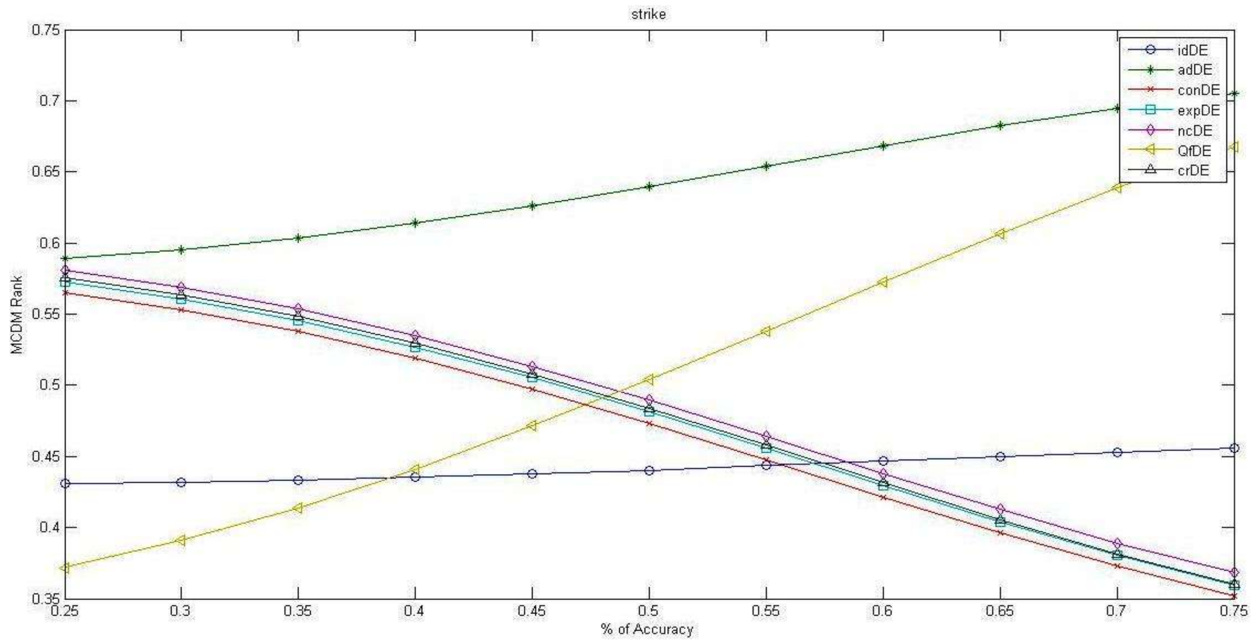

Figure 5.5 MCDM ranking graph for Strike dataset with Variation of accuracy involvement

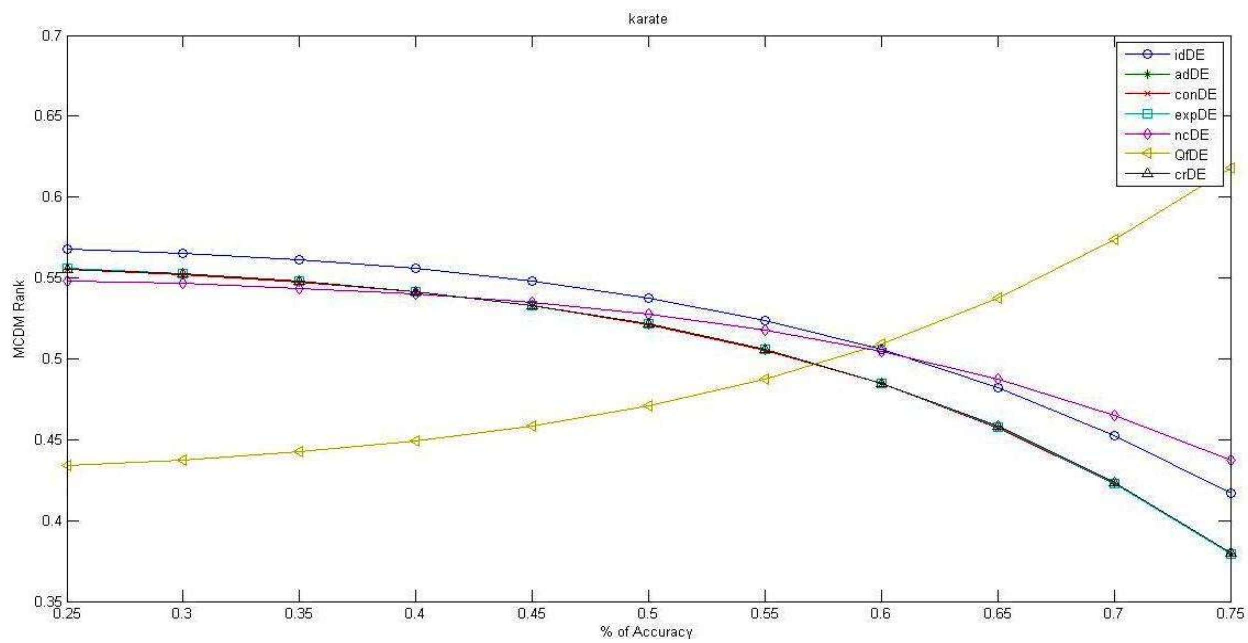

Figure 5.6 MCDM ranking graph for Karate club Dataset with Variation of accuracy involvement

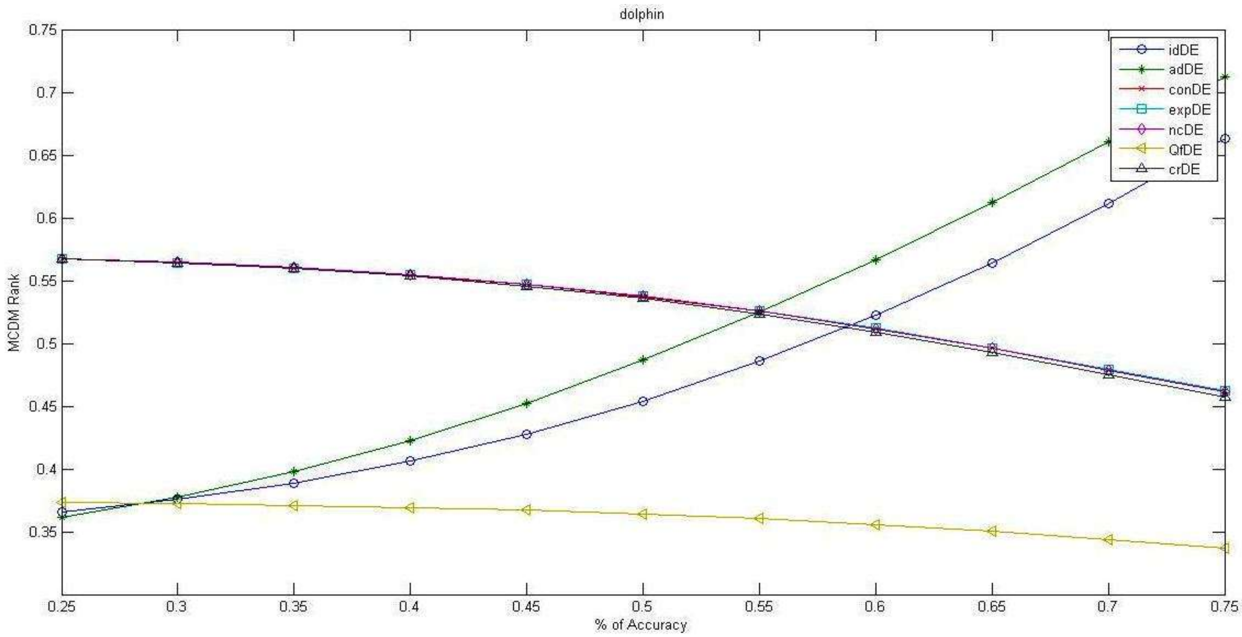

Figure 5.7 MCDM ranking graph for Dolphin Dataset with Variation of accuracy involvement

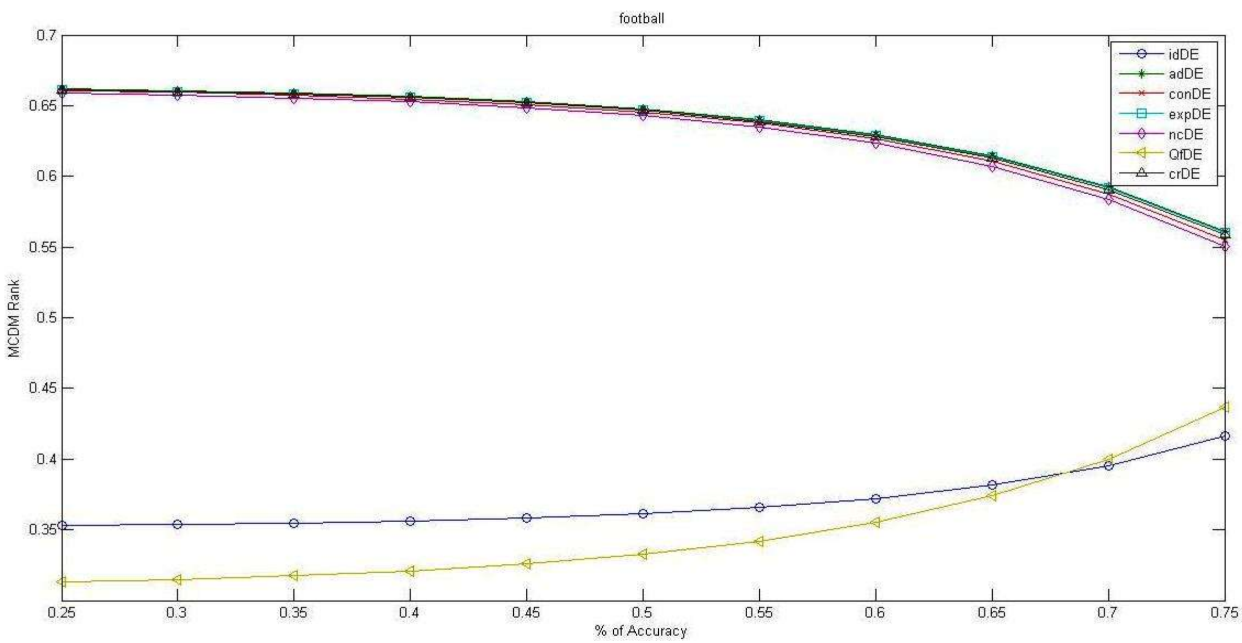

Figure 5.8 MCDM ranking graph for Football club Dataset with Variation of accuracy involvement

After this experiment, we have done another experiment with DE algorithm also. In this work we employed the node similarity concept for initialization phase for optimize the DE algorithm. We compare the result of VSDE with the DECD in this experiment.

5.4. Vertex Similarity Based Differential Evolution

Differential evolution (DE) is a simple and efficient stochastic population-based optimization algorithm proposed by R. Storn and K. Price in 1995. It comes under the class of evolutionary algorithms which also includes Genetic algorithms and others. Differential evolution has also been employed earlier for community detection in complex networks (Du, et al., 2008). We present an improvised DE based approach with biased initialization based on the concept of vertex similarity (Leicht, et al., 2006).

In DE, the initialization step starts with a population size of NP. In the mutation step, one individual, called the donor vector is selected to generate a mutant vector using the mutation operation. The mutation strategy used in VSDE is the "rand/2" strategy. It has been observed that, increasing either the population size or the number of pairs of solution to compute the mutation values the diversity of the possible movement increases, which in-turn promotes the exploration of the search space. The rand/2 strategy is described as:

$$V_i = X_{r1} + F_1(X_{r2} - X_{r3} + X_{r4} - X_{r5}) \quad \text{………….. (5.1)}$$

Where r1, r2, r3, r4 and r5 are randomly selected individuals from the population and satisfy r1 ≠ r2 ≠ r3 ≠ r4 ≠ r5.The mutation factor is usually a real number in the range [0, 1]. After this step, the mutant vector is checked if any of its components violate the boundary constraints. If they do, then they are corrected as in DECD .After mutation, recombination is done on the mutant vector v_i to generate trial vector u_i as:

$$U_{i,j} = \begin{cases} V_{i,j} & \text{if } rand_{i,j} \le CR \text{ or } j = irand \\ X_{i,j} & else \end{cases} \quad \text{……………. (5.2)}$$

Finally, the selection process is employed over the trial vector and the target vector and they are compared in terms of the objective function to select the better one to pass to the next generation.

$$X_i^{k+1} = \begin{cases} U_i^k & \text{if } f(U_i^k) < f(X_i^k) \\ X_i^k & \text{else} \end{cases} \quad \text{……………. (5.3)}$$

5.4.1. Proposed Work Description

Typically, the problem of identifying communities in a network is classified as NP-hard. As no effective deterministic and polynomial time algorithm exist for solving this problem, many nature inspired meta-heuristic approaches have been exploited in the recent past like GA and DE to detect communities in complex networks. As meta-heuristics such as DE do not guarantee optimal results, the steps involved such as the initialization, mutation etc. plays an important role in approaching the optimal result.

In this experiment, we have proposed a new biased initialization using the notion of vertex similarity. To the best of our knowledge vertex similarity has never been used with DE for community detection in complex networks.

- **Initialization**

Generally, DE uses random initialization to produce the initial population. Here we employ a new initialization process in DE based on the concept of vertex similarity and neighborhood principle which exploits the structural similarity of nodes in the network. We abbreviate this algorithm of population initialization as VSDE.

A vertex j is similar to vertex i if i has a network neighbor v that is itself similar to j. In this case, it can be concluded that vertices are said to be similar if they are similar to vertices which are they similar.

$$\mathrm{DSD} = \frac{\alpha}{\lambda_{\max}} \mathrm{A}(\mathrm{DSD}) + \mathrm{I} \qquad \text{……………… (5.4)}$$

Where, A is the adjacency matrix of the network, λ_{max} is the maximum Eigen-value of A; D is the nth order diagonal matrix whose elements are

$$\mathrm{d_{ii}} = \sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{a_{ij}}\,, \mathrm{i}, \mathrm{j} = 1,2, \dots, \mathrm{n} \qquad \text{……………….. (5.5)}$$

representing the degree of nodes, I is the identity matrix and S is the similarity matrix. Alpha is in the range [0, 1] and is taken to 0.97.

The initial population is generated using the similarity matrix “S” obtained above. Now as it is evident that nodes which are immediate neighbor should preferably lie in the same community, we extend the initial population matrix obtained with the neighborhood principle i.e. randomly selecting some nodes and assigning their commIDs to their immediate neighbors. Through this process the initial population P0 is generated.

- **Fitness function**

VSDE uses Network modularity as fitness function which was proposed by Newman and Girvan. Due to its ability to detect good communities it has been used by many community detection algorithms. Higher value of modularity indicates more stable and fit community structure, therefore our aim is to maximize the fitness value.

- **Mutation**

Our VSDE used rand/2 mutation strategy for mutation of individuals in the population. It has been observed that, increasing either the population size or the number of pairs of solution to compute the mutation values, the diversity of the possible movement increases which in-turn promotes the exploration of the search space. For mutating according to rand/2 strategy, five random individuals are chosen from the population NP and are mutated according to the equation (1).

- **Crossover**

VSDE follows the crossover strategy just as the DECD in Reference.

- **Clean-up step**

It is possible that some nodes may be put into a wrong community in the process of evolution Reference. The clean-up operation is based on community variance and is employed as in.

5.4.2. Experimental Result

VSDE is tested against an artificial random network (Zhou, 2003) and two real world networks (Kaiser and Hilgetag, 2004).VSDE has been implemented in Matlab and on Windows 7 Home Premium with Intel Core i5 2.40GHz processor and 4.0GB RAM. The parameters used in experiments are given in Table 5.9.

As metric based on network modularity may not be reliable so we have also employed accuracy as another quantitative metric for evaluation.

Parameter	Value	Description
NP	200	Population Size
F	0.9	Mutation Factor
CR	0.3	Crossover Parameter
η	0.35	Threshold Value
N_{max}	200	Number of generations

Table 5.9: Parameters of VSDE

➢ Artificial Network

To test the performance of VSDE, artificial network proposed by Girvan and Newman was used (Li, et al., 2008) . The network has 128 nodes divided into 4 communities of 32 vertices each. For each node, Z_{in} is defined as the average edges connecting it to the members of the same community and Z_{out} as the average edges connecting it to members of other communities.

In Figure 5.9 -5.10, the horizontal axis represents the number of inter−community edges Z_{out} and the vertical axis represents the fraction of vertices classified correctly. It is clear that our VSDE performs significantly better than DECD in the range $Z_{out} > 7$ while it performs almost same when $Z_{out} <= 7$.

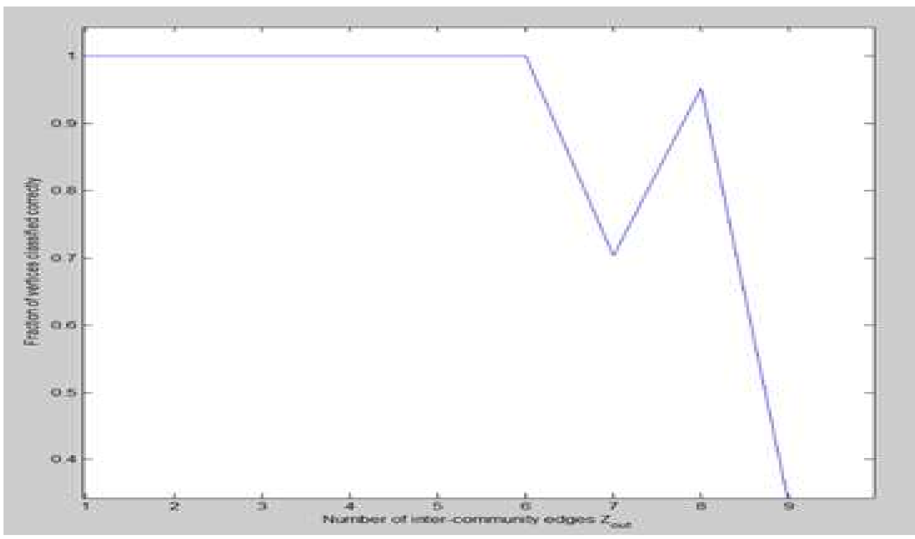

Figure 5.9 fraction of vertices correctly classified by VSDE

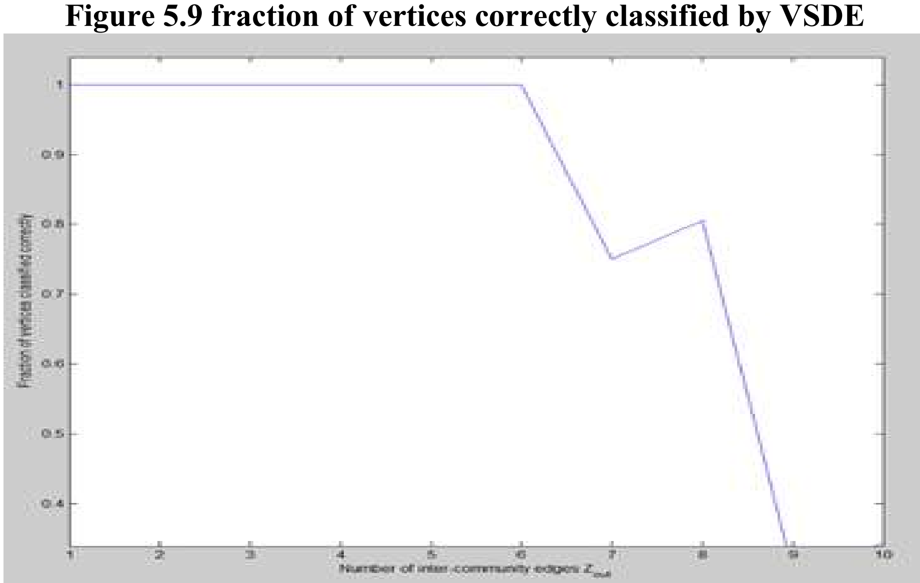

Figure 5.10 fraction of vertices correctly classified by DECD

- **Real World Social networks**

The performance of VSDE was further tested on two real-world social networks, Zachary's karate club network and American college football network.

- Zachary's karate club network

  This network shows the social network of friendships between members of a karate club in an American university. Here, nodes represent club members and edges indicate social communication between them. There are a total of 34 nodes and 78 edges in the network which splits into two independent clubs due to internal divergence.

- American college football network

  This network is a representation of the schedule of Division I games for the 2000 games. In this football network, there are 115 nodes and 616 edges divided into 12 communities representing the teams and the regular season games between the two teams they connect, respectively. In the regular season, teams attend 12 conferences of different sizes. The majority of matches are played between teams within the same conference, thus the 12 conferences constitute the network's 12 real communities.

As discussed above the metric accuracy is defined as-

$$\text{Accuracy} = \frac{\sum_{k=1}^{n} equal(t_k, p_k)}{n} \quad \ldots\ldots\ldots\ldots\ldots\ldots (5.6)$$

Where,

$$\text{Equal (x, y)} = \begin{cases} 1, & if commID(x) = commID(y) \\ 0, & otherwise \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots (5.7)$$

From Table 5.10, it is clear that VSDE performs better than DECD on both karate and football networks. It is observed that for a larger football network the performance of VSDE is significantly better as it detects eleven communities which is more close to the real world football network. In the Zachary's network the modularity and the accuracy both have increased considerably depicting a better community formation when compared to DECD.

We have presented VSDE as an improvisation of existing differential evolution technique to identify community in complex networks. This algorithm requires no prior information about the networks which makes VSDE suitable for real world applications. It is clear that VSDE outperforms DECD and other previously proposed approaches and can compete with any community detection algorithm. There is a lot of scope for improvement of this algorithm by employing different fitness functions or by using other properties of graphs for initialization and selection etc.

Network	Algorithm	N_{pr}	Q_{avg}	Q_{bst}	Acc_{avg}	Acc_{bst}
Karate	VSDE	4	0.417±0.002	0.419	0.976±0.006	1
	DECD	4	0.415±0.001	0.416	0.970±0.001	0.970
Football	VSDE	11	0.603±0.0	0.603	0.956±0.0	0.956
	DECD	10	0.604±0.0	0.604	0.947±0.0	0.947

Table 5.10: Experimental results of the Zachary's karate club network and the American college football network, N_{pr} is the average number of communities; Q_{avg} and Q_{bst} are the average and best values of modularity Q, respectively; and Acc_{avg} and Acc_{bst} are the average and best accuracy, respectively.

After this VSDE, we have done another experiment on DE algorithm. We employed the opposition based learning and tournament method with DE algorithm into different steps such as initialization and selection, both are very important role play in that algorithm. So that we created the three new version of DE like as TDE (Tournament based DE), OBDE (Opposition learning based DE) and TOBDE (both tournament and opposition based DE) and compare the SDE (simple DE).

5.5. Tournament & Opposition Learning Based DE

In this Section the detailed Explanation of our Algorithm is given. The Algorithm is mainly based on Opposition learning and Tournament Selection method which helps in fast Convergence of the solution members. Pseudo code of this proposed algorithm is given below:

Algorithm pseudo-code of TOBDE

For j=1:No. Of iteration

1.) Choose random population with total number of communities equal to Nmax collectively forming a 2D matrix, say MAT_A

2.) Calculate fitness value of each population and sort the population according to the increasing level of fitness value and maintain same order matrix size.

3.) Using Xi = A+B-Xi, calculate the opposite community of each node where A is the lowest community value and B is the highest community value in the individual containing the particular node, let it be the MAT_B

4.) For i=1: N_{max}

a.) if MAT_A[i] q-value <MAT_B[i] q-value

MAT_A[i] = MAT_B[i]

5.) Take MAT_A as resultant matrix

6.) For i=1: P_n*P_m

a.) Take 3 random individuals

b.) Apply tournament method, sort in the order A, B, C according to the Q-value with A being the best

c.) Make new individual vector, V_i= A+ f*(B-C) in the maintained space.

7.) For i=1: P_n*P_c

a.) Apply binomial crossover and keep in the maintained space.

8.) Select fittest P_n individuals according to the fitness value among total of P_n+ (P_n*P_c) + (P_n*P_m) vectors.

Parameter	Value	Description
N_{max}	100	Total number of iterations.
P_n	100	Population Size
P_m	0.2	Mutation rate
P_c	0.8	Cross over rate
F	0.5	Differential Amplification Factor

Table 5.11: Description of the Parameters

5.5.1. Experimental Analysis & Result Discussion

Our present algorithm is tested on some real life data sets which are Zachary's karate club, Dolphin sociality, American college football, Strike Data set and compared with the algorithms SDE (Jia, et al., 2012), TOBDE and TDE (Qu and Suganthan, 2010).This experiment was carried on Microsoft Windows 10 (X64) operation system using a R programming platform; Intel (R) Core (TM) i5-7200U CPU T8300 @ 2.50GHz 2.71 GHz processor, 4.00 GB memory and 1024 GB hard disk. Detailed analysis of our experiments is described below in the following sections of this paper.

5.5.1.1. Modularity with Number of Iteration graph Analysis

- **Strike Data set**

In Figure 5.11, The change of average modularity functional value (Q) which determining the fitness (Quality) of candidate solutions obtained through our algorithms (OBDE, TDE, TOBDE) with no of iterations made is compared with SDE. The data is graphically analyzed and the results obtained is all the four algorithms show exponential increase in Q value up to first 20 iterations and for next 80 iterations q values are almost same .comparing with SDE our algorithms(OBDE, TDE , TOBDE) are giving better quality values(Q) i.e. OBDE, TDE , TOBDE are reaching the Q value up to 0.55 whereas SDE giving Q value up to 0.5 after 100 iterations.

- **Karate club dataset**

In Figure 5.12, For this network also all the four algorithms show exponential increase in Q value up to first 20 iterations and for next 80 iterations q values are increasing but growth of the curve is less .comparing with SDE our algorithms (OBDE, TDE, TOBDE) are giving better quality values (Q) i.e. OBDE, TDE , TOBDE are reaching the Q value up to 0.4 whereas SDE giving Q value up to 0.38 after 100 iterations

- **Dolphin Sociality**

In Figure 5.13, For this network also all the four algorithms show exponential increase in Q value up to first 20 iterations and for next 80 iterations q values are increasing, but growth of the curve is less. Through the result we obtained highest Q value of 0.42 after 100 iterations for TOBDE approach.

- **American Football club**

In Figure 5.14, For this network also all the four algorithms show exponential increase in Q value up to first 20 iterations and for next 80 iterations q values are increasing, but growth of the curve is less. Through the result we obtained highest Q value of 0.53 after 100 iterations for TOBDE approach.

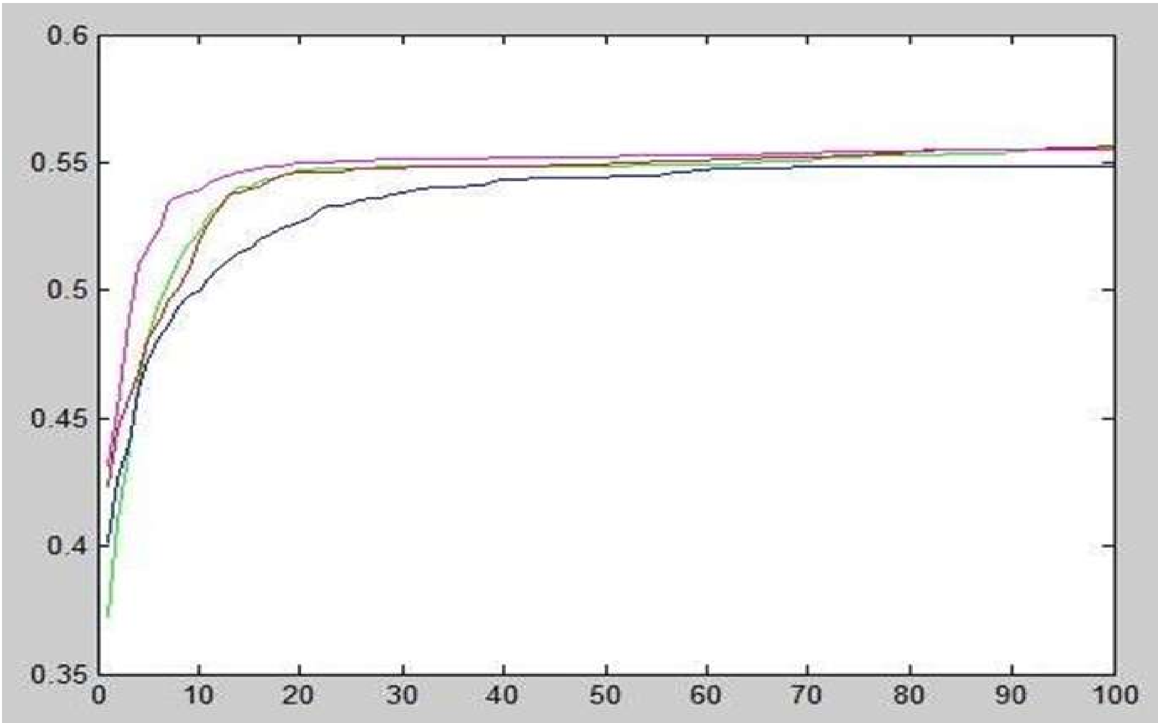

Figure 5.11 Q-fun value (on Y axis) vs. no. of iteration (on X axis) for comparison between SDE, OBDE, TDE and TOBDE for Strike dataset.

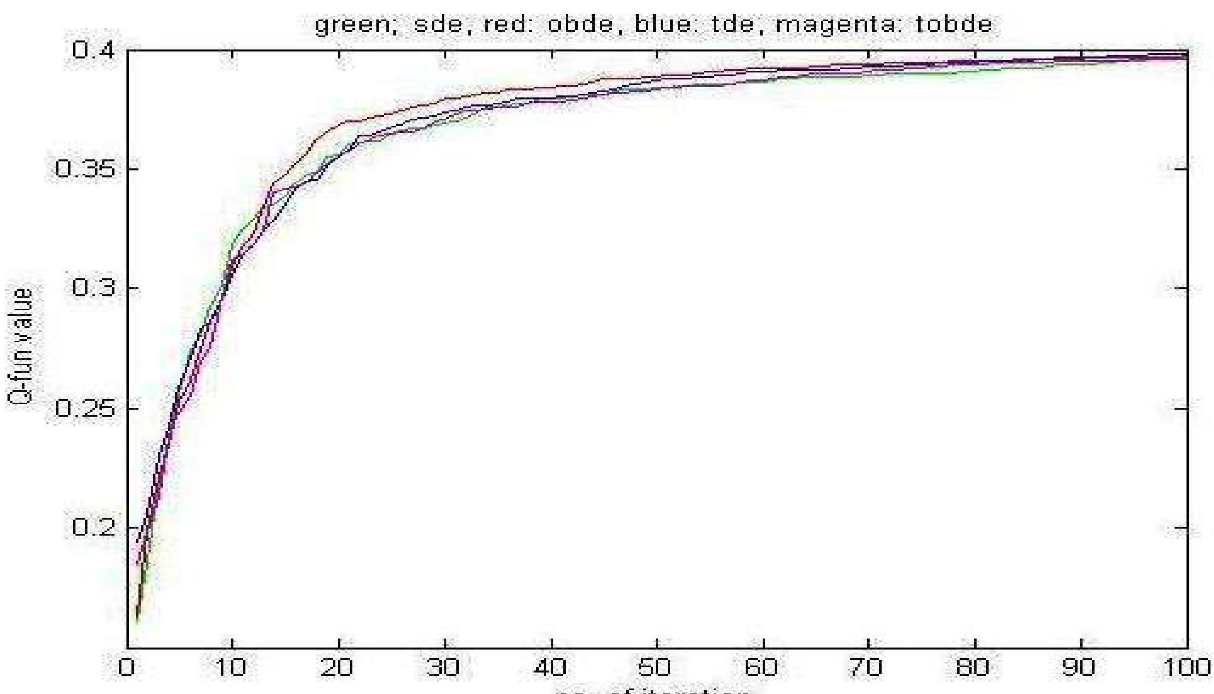

Figure 5.12 Q-fun value (on Y axis) vs. no. of iteration (on X axis) for comparison between SDE, OBDE, TDE and TOBDE for Karate club dataset.

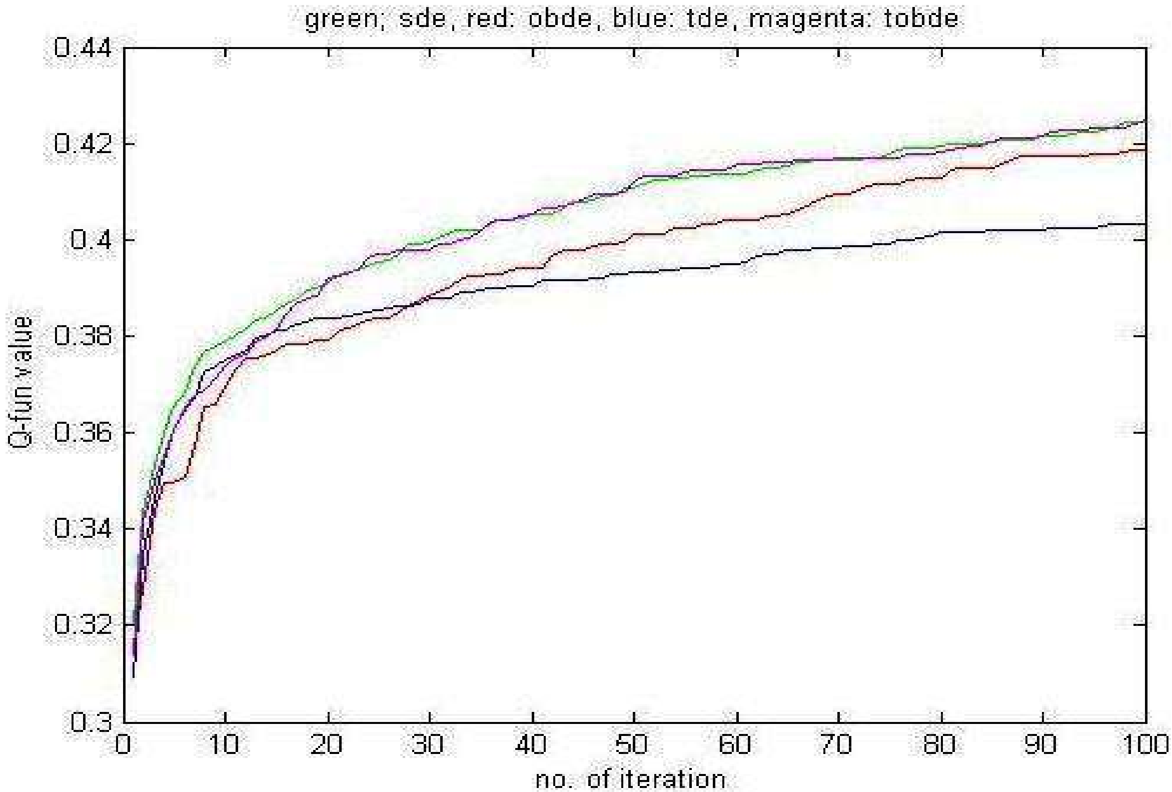

Figure 5.13 Q-fun value (on Y axis) vs. no. of iteration (on X axis) for comparison between SDE, OBDE, TDE and TOBDE for Dolphin dataset.

5.5.1.2. Accuracy & Quality metric values Analysis

We considered several parameters (ENT_avg, ACC_avg, NMI_avg, FMEA_avg, Q _avg) where ACC, NMI, FMEA are the measures of accuracy and purity, ENT and Q measures of Quality of partitioning.

- **Strike Data set**

In Table 5.12, By running our algorithms(OBDE, TDE , TOBDE) on this data set we got the partitioning of communities with highest Quality measures as we got highest Q avg (0.5455) and also with highest accuracy measures ACC_avg(0.9837), NMI_avg(0.971), FMEA_avg(0.9503), ARI_avg(0.9641) for TOBDE approach as compared with SDE.

- **Karate club dataset**

In Table 5.13, Running our algorithms(OBDE, TDE , TOBDE) on this data set we got the partitioning of communities with highest Quality measures as we got ENT_avg (0.5229) and also with highest accuracy measures ACC _avg(0.8732), FMEA _avg(0.7368) for TOBDE approach as compared with SDE.

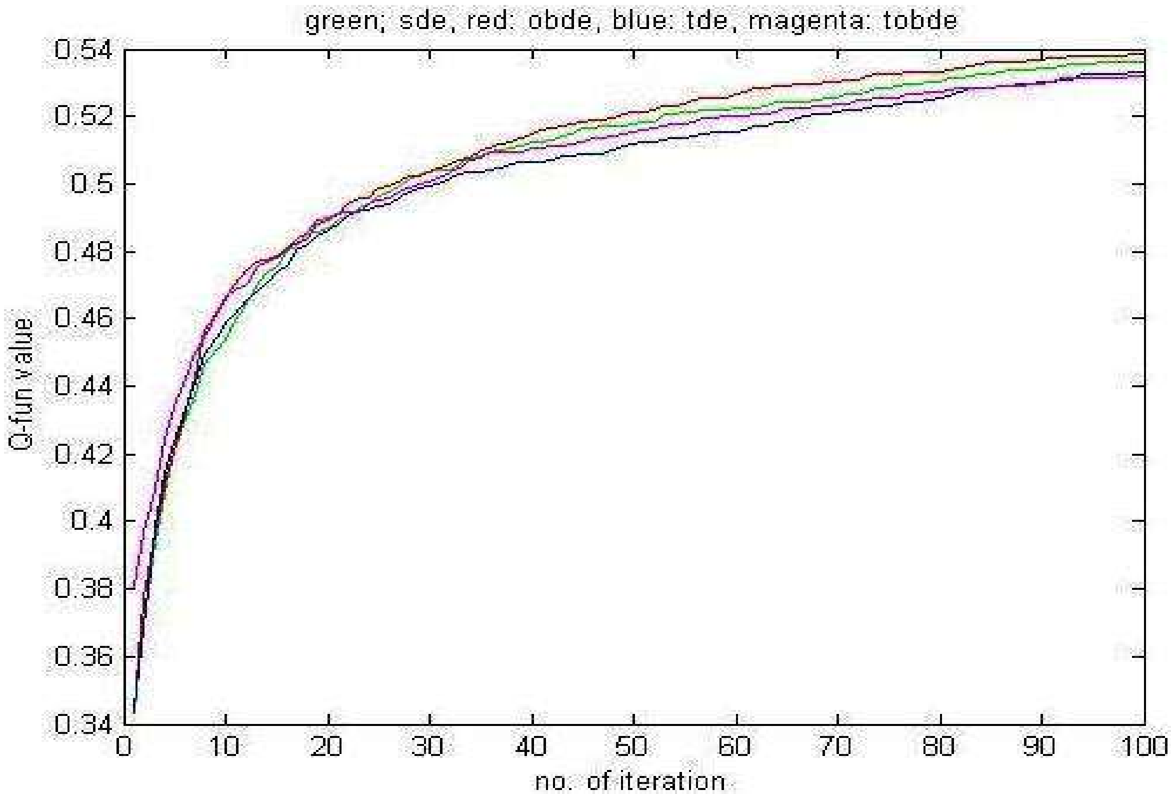

Figure 5.14 Q-fun value (on Y axis) vs. no. of iteration (on X axis) for comparison between SDE, OBDE, TDE and TOBDE for American Football dataset.

Algorithm	ENT avg	ACC avg	NMI avg	FMEA avg	ARI avg	Q avg
SDE	0.0809	0.9569	0.9254	0.8907	0.9028	0.5426
OBDE	0.0756	0.9598	0.9317	0.885	0.9087	0.5438
TDE	0.0398	0.9767	0.9573	0.9495	0.9486	0.5438
TOBDE	0.0267	0.9837	0.971	0.9503	0.9641	0.5455

Table 5.12: Accuracy & Quality metric values for Strike data set whose ground truth communities are known

Algorithm	ENT avg	ACC avg	NMI avg	FMEA avg	ARI avg	Q avg
SDE	0.0809	0.8681	1	0.7279	1	0.551
OBDE	0.0756	0.8521	1	0.703	1	0.4676
TDE	0.0398	0.8629	1	0.7205	1	0.5171
TOBDE	0.3047	0.8732	1	0.7364	1	0.5229

Table 5.13: Accuracy & Quality metric values for Dolphin Dataset whose ground truth communities are known

➢ Dolphin Sociality

In Table 5.14, Running our algorithms (OBDE, TDE , TOBDE) on this data set we got the partitioning of communities with highest Quality measures as we got highest ENT_avg (0.5455) and highest Q_avg for OBDE and also with highest accuracy measures ACC_avg(0.8042), NMI _avg(0.6389), FMEA_avg(0.4761), ARI_avg(0.4761) for TOBDE approach as compared with SDE.

Algorithm	ENTavg	ACC avg	NMI avg	FMEA avg	ARI avg	Q avg
SDE	0.4231	0.7826	0.6123	0.4626	0.5641	0.4058
OBDE	0.3981	0.7981	0.6266	0.4751	0.6053	0.4068
TDE	0.4299	0.7746	0.6082	0.4423	0.5618	0.408
TOBDE	0.3812	0.8042	0.6389	0.4761	0.6176	0.4045

Table 5.14: Accuracy & Quality metric values for Dolphin Dataset whose ground truth communities are known

➢ American Football club

In Table 5.15, Running our algorithms(OBDE, TDE , TOBDE) on this data set we got the partitioning of communities with highest Quality measures as we got highest Q avg (0.5093) and also with highest accuracy measures ACC avg(0.8327), NMI avg(0.66), FMEA avg(0.2558), ARI avg(0.3943) for TOBDE approach as compared with SDE.

Algorithm	ENT avg	ACC avg	NMI avg	FMEA avg	ARI avg	Q avg
SDE	0.1374	0.8225	0.645	0.2524	0.3783	0.5093
OBDE	0.1334	0.8295	0.6632	0.2607	0.3927	0.5117
TDE	0.1381	0.8189	0.6469	0.2504	0.3735	0.5081
TOBDE	0.1354	0.8327	0.66	0.2558	0.3943	0.5138

Table 5.15: Accuracy & Quality metric values for American Football club Dataset whose ground truth communities are known

5.5.1.3. MCDM Rank Analysis

By considering accuracy and quality of partitioning of communities multi criteria decision making (MCDM) rank is given to algorithms tested on real life networks and change of MCDM ranks with the percentage accuracy of our algorithms is described below.

In Figure 5.15 is decreasing with increasing level of accuracy for TDE and TOBDE and the graph is increasing for SDE and OBDE algorithms. Up to 75% level of accuracy MCDM ranks of these algorithms are plotted. The maximum rank (0.65) with 25% accuracy is obtained for TDE approach.

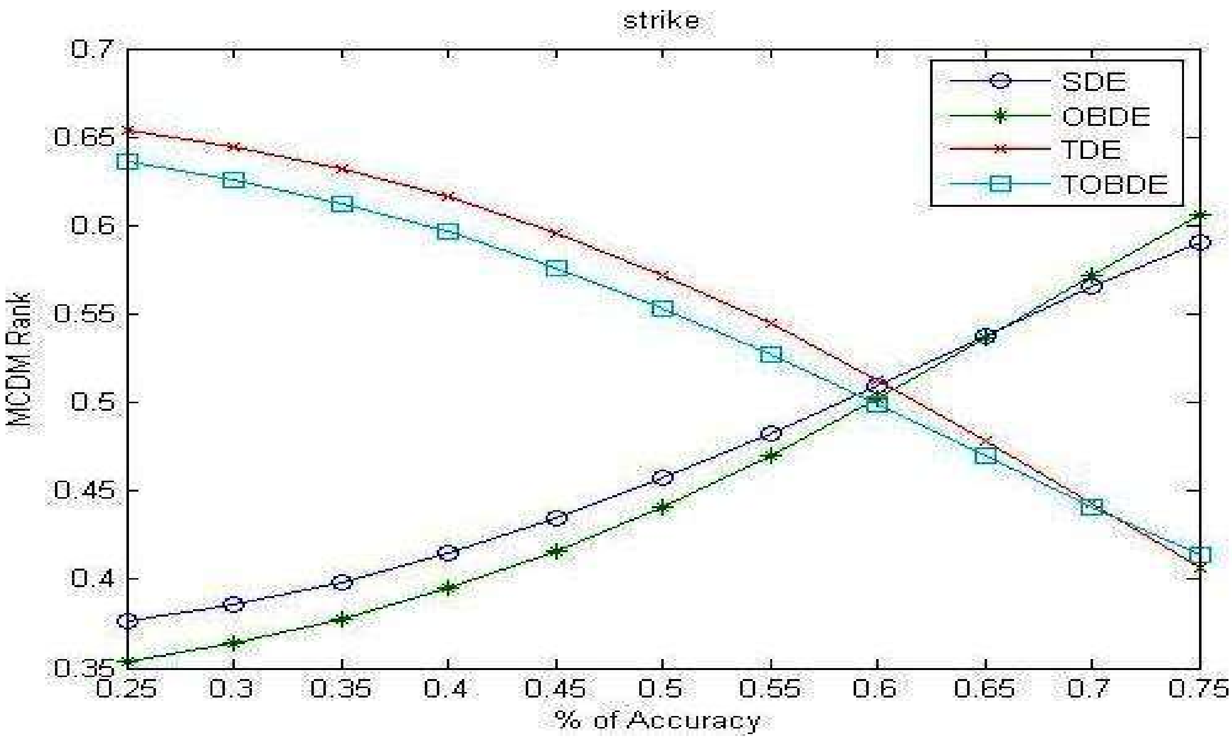

Figure 5.15 MCDM ranking acquired by each algorithm in Real world known network for Strike dataset with Variation of accuracy contribution

In Figure 5.16 is decreasing with increasing level of accuracy for TDE , TOBDE and SDE algorithms, the graph is increasing for OBDE algorithms. Up to 75% level of accuracy MCDM ranks of these algorithms are plotted. The maximum rank (0.55) with 75% accuracy is obtained for OBDE approach.

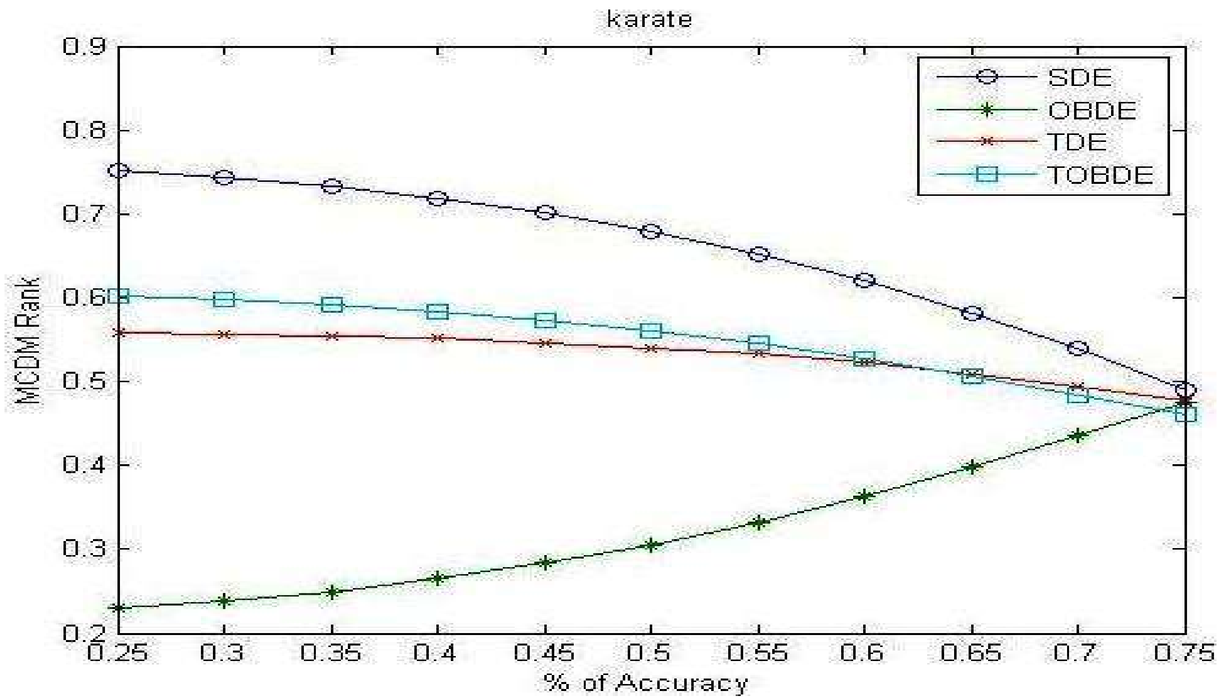

Figure 5.16 MCDM ranking acquired by each algorithm in Real world known network for Karate Club dataset with Variation of accuracy contribution

In Figure 5.17 is decreasing with increasing level of accuracy for OBDE and TOBDE and the graph is increasing for SDE and TDE algorithms. Up to 75% level of accuracy MCDM ranks of these algorithms are plotted. The maximum rank (0.68) with 25% accuracy and with 75% accuracy highest rank (0.55) is obtained for TOBDE approach.

In Figure 5.18 is decreasing with increasing level of accuracy for SDE and the graph is increasing for TDE algorithms. For OBDE and TOBDE it is almost parallel. Up to 75% level of accuracy MCDM ranks of these algorithms are plotted. The maximum rank (0.62) with 75% accuracy is obtained for TDE approach.

By taking accuracy and quality as two multiple criteria, MCDM rank is calculated for our algorithms (OBDE, TOBDE, TDE) for our input data sets (Strike Data set, Zachary's karate club, Dolphin network, American Foot ball Network).for Strikes network we got highest rank (0.618) through OBDE approach, for Dolphin network we achieved highest rank (0.5117) through TOBDE algorithm and for American Foot ball network we achieved highest rank(0.5889) through TDE algorithm.

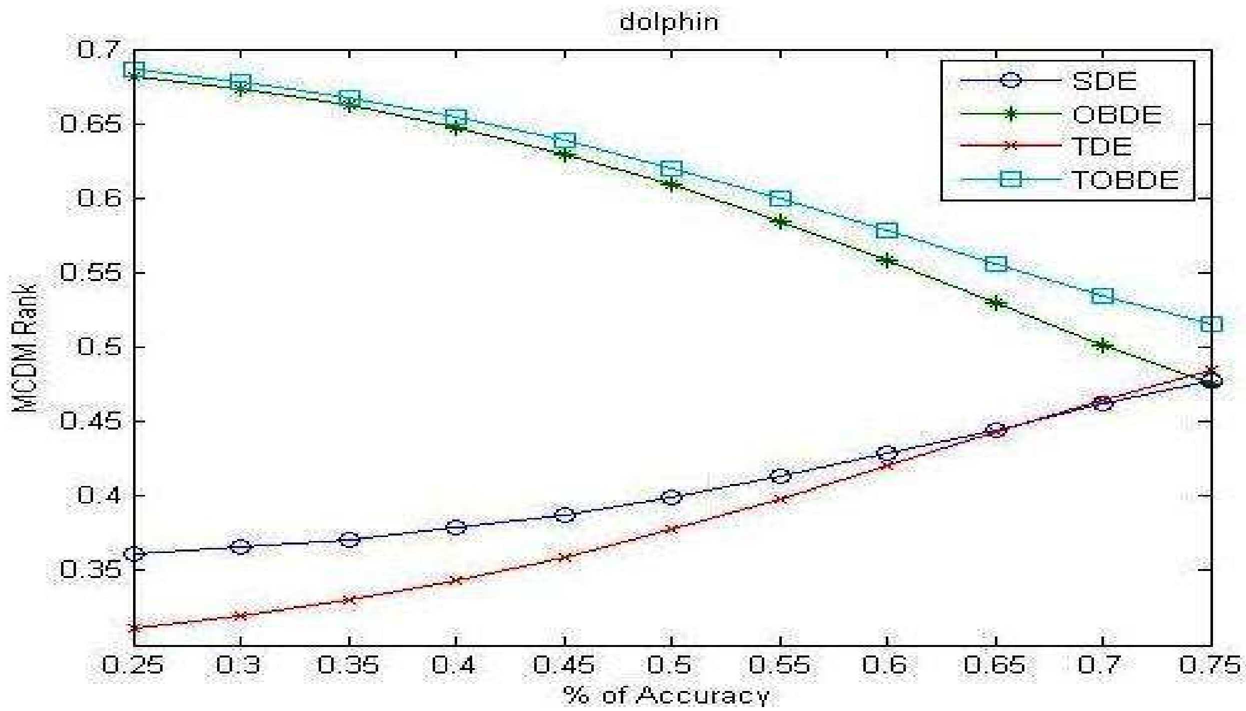

Figure 5.17 MCDM ranking acquired by each algorithm in Real world known network for Dolphin dataset with Variation of accuracy contribution

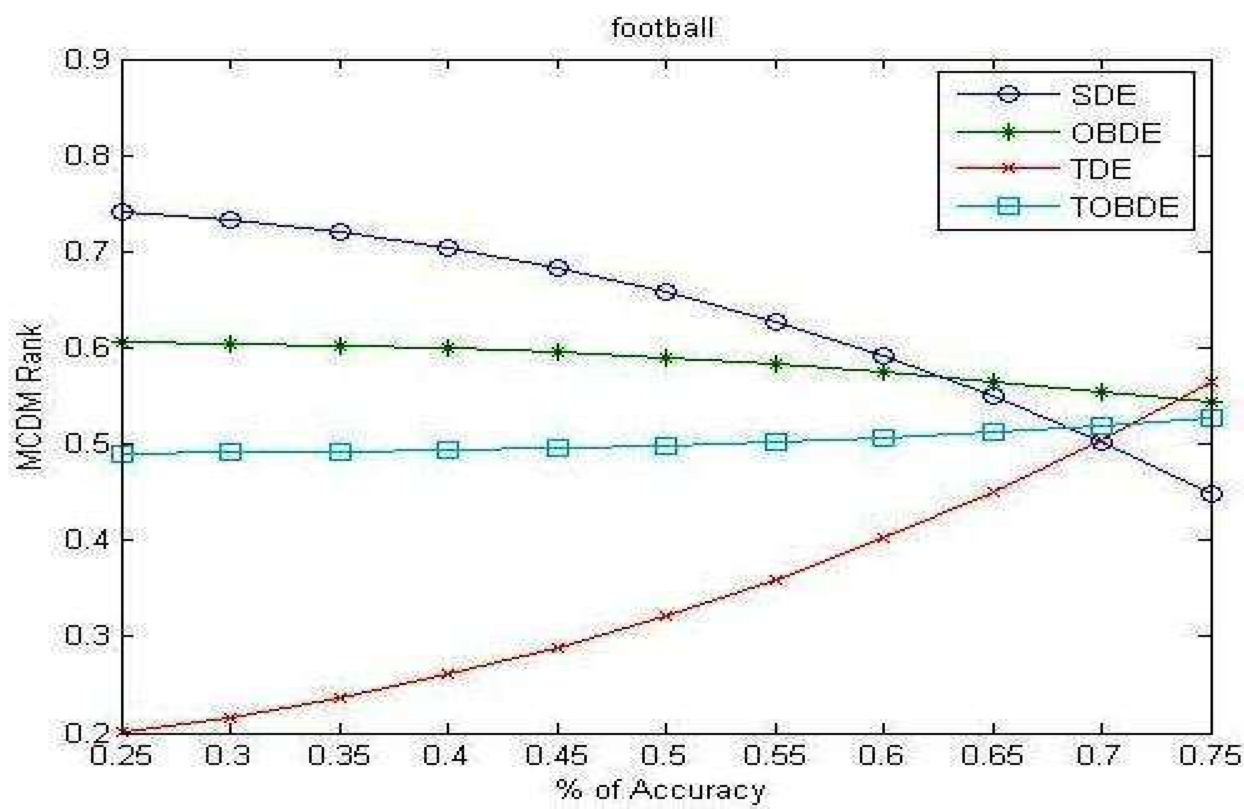

Figure 5.18 MCDM ranking acquired by each algorithm in Real world known network for American Football Dataset with Variation of accuracy contribution

DATASET	ALGORITHM	MCDM RANK
STRIKE	SDE	0.5921
	OBDE	**0.6118**
	TDE	0.4109
	TOBDE	0.4238
KARATE	SDE	**0.4866**
	OBDE	0.4805
	TDE	0.4756
	TOBDE	0.4619
DOLPHIN	SDE	0.4472
	OBDE	0.4716
	TDE	0.4835
	TOBDE	**0.5117**
FOOTBALL	SDE	0.4519
	OBDE	0.5789
	TDE	**0.5889**
	TOBDE	0.5623

Table 5.16: MCDM ranking score obtained with 75% Accuracy and 25% Quality. Higher score indicate more inclination of algorithm towards accuracy.

In this experiment, we totally focused on the DE with Tournament and opposition learning based concept for the community detection in social networks. After done this experiment and result analysis we found that the new DE is very much useful for optimize the resultant. In DE algorithm, very much hope for the further research.

According to the MCDM rank Table 5.16, we confirmed that the proposed OBDE , TDE and TOBDE is better performance compare to the SDE for all the employed standard datasets. In future we utilized that concept on the swarm techniques just like a ant colony optimization and social spider algorithm and etc. I hope that experiments are also successful for the social network analysis. Both concepts are successful for enhance the convergence rate and reduce the slow performance of the population based algorithms mostly evolutionary algorithms. Genetic Algorithm and Differential Evolution is the most used algorithm in the evolutionary category.

5.6. Conclusion of the Chapter

In this Chapter, we employed the differential evolution algorithm for the some useful experiment. We mainly focus on DE algorithm and fitness functions because objective functions have played a significant role in optimization in a different type of research area. We used the DE with multiple objective functions; in this experiment we replaced the fitness function with the help of some other functions like as conductance, normalized cut, internal density, average degree, expansion, Cut ratio. Objective functions have checked the outcomes of the algorithms for community detection in social networks. Quality and accuracy of the communities are verified and also test the sensibilities with the help of objective functions in the social network. We have done experiments on Differential evolution algorithm using various types of different objective functions and optimize the results of community detection for the maintained range of datasets. After performed the operation resultant show that DE is surely additional expectable for expansion, internal density, and average degree as a fitness function. Apparently, we found another better option as fitness functions for the DE algorithm, and we will provide a better choice to the users according to categories of datasets. We will choose a fitness function according to requirement and find the optimized results.

After this experiment, we have done another research work depend on DE algorithm. Its name is VSDE means vertex similarity based DE, in this work we used the vertex similarity concept for the population initialization because it's a very important role play in this algorithm. After result we have show our algorithm strength with the help of positive results. It is also a very useful and successful experiment in the domain of social network analysis.

We have tried to another research on the DE algorithm; in this experiment we used the opposition learning concept and tournament method for the different phase of the DE algorithm. We used the differential evolution algorithm for community detection in complex network. DE is evolutionary technique i.e. swarm techniques. We improved the classical version of Differential evolution for optimization. In this work, we modified the DE with the help of some other concepts i.e. initialization process modified by opposition based concept and selection process done by the tournament method. A new DE algorithm employed the real world and artificial datasets for community detection in social networks. In recent scenario, DE is a very well known algorithm for optimization after my experiment gain the new height in social network analysis and community detection. In this experiment, DE is good for accuracy-wise and quality-wise for community identification. According to results new DE is better performance compare to other evolutionary algorithm.