SYNOPSIS

1. INTRODUCTION

The increased assimilation of computing into day to day life has lead to the creation of colossal datasets in most realms of science and engineering, such as bioinformatics, Geographic Information Systems, Medical Records, Weather images, Intelligence Agency videos, e-commerce, online social network etc. The pace of data production is further expected to accelerate in the next few years. The volume of data is expanding in two folds: number of records and number of dimensions [1]. For example in a customer purchase behavior data set, along with the records of millions of users, there are hundreds of thousands of products each of which is mapped to a dimension. Similarly in movie ratings dataset, tens of thousands of movies are rated by millions of users where each user represents one dimension. Managing and processing such high dimensional data and attaining significant insights is one of the major challenges faced by the data scientists [8]. The two common challenges faced by the researchers and practitioners for analyzing high dimensional data include “curse of dimensionality” [2], [8], [9] and inherent sparsity in data space. The first challenge indicates that with the growth of dimensions, conventional data mining techniques become computationally expensive. Hence, their performance degrades with increase in dimensions and become inapplicable to most of the real world applications where dimensions are high. Second challenge is inherent sparsity in data space which signifies that data points in high dimensional data space are sparse and appear equidistant from each other. Consequently distance measure loses its significance in high dimensional space. This makes the data mining algorithms more vulnerable to noise. Researchers are constantly involved in finding efficient techniques for analyzing, representing and summarizing high dimensional data. Mainly three types of data mining models are given for handling large datasets [198]. These are frequent pattern mining, classification and clustering. Frequent pattern mining is based on market basket data analysis. It is used to determine collection of items appearing in large number of transactions. However, this approach produces redundant information regarding large number of frequent patterns. Classification is a supervised
technique which is used to predict class label for known item. This technique works on dataset with known classes only. Clustering is an unsupervised data mining technique as it analyzes data without prior class labels and does not produces any redundant information. Therefore this research work focuses on clustering for handling high dimensional data.

Clustering [3] is one of the data mining techniques which segments the large data into groups based on their similarity. The criteria for assembling the similar data points into one group and dissimilar in other group vary from algorithm to algorithm. There are variety of clustering algorithms developed by the research communities [25]. These are partition based (K-Means, K-Mediods, K-Modes, CLARA, PAM, fuzzy c means), density based (DBSCAN, OPTICS, DENCLUE), hierarchical based (BIRCH, CURE, ROCK), grid based (STING, OPTIGRID, CLIQUE) and model based (EM, COBWEB). Hierarchical clustering algorithms suits to the problems with point linkages but has inability to make decisions once splitting or merging is done. Density based algorithms finds clusters of arbitrary shapes however these are sensitive to its input parameters. Grid based clustering is though fast but shape of clusters depends on union of grids. All cluster boundaries are vertical or horizontal but not diagonal. Model based clustering approach tries to optimize the fit between the model and the cluster. However, this type of clustering technique is quite expensive. Partition based clustering algorithms are more prevalent as they are considered to be more suitable for solving scalability problem as compared to others. Hence, current research favors partition based clustering over other clustering algorithms.

K-Means clustering is one of the popular partition based clustering algorithm. This algorithm is heuristic in nature and determines the clusters in polynomial time. However, it is not guaranteed to obtain the optimal clusters as its efficacy depends upon the initial seeds (centroids) chosen. Due to this, it also has high probability of getting trapped in the local optima. Therefore, finding optimal clusters is a NP hard problem[31]. The objective of clustering is to find clusters with minimum intra cluster distance. The intra cluster distance can be expressed as:

\[
\sum_{i=1}^{k} \sum_{j=1}^{s} w_{ji} \sum_{m=1}^{d} ||x_{jm} - \text{Cen}_{im}||^2
\] (3.1)
This is sum of square of Euclidean distance of data points with its respective centroids. In Equation 3.1, \( k \) is the number of clusters, \( s \) is the total number of data points with \( D \) attributes, \( x_{jm} \) is the value of \( m^{th} \) attribute of \( j^{th} \) data point, \( Cen_{im} \) is the value of \( m^{th} \) attribute of centroid of \( i^{th} \) cluster, \( w_{ji} \) is a binary variable a takes value 1 if data point \( x_{jm} \) belongs to \( k^{th} \) cluster otherwise 0. Centroid matrix \( Cen \) holds \( k \times D \) values of centroids in our computational model. For measuring the distance between each pair of data point and centroid, \( Cen \) is scanned \( k \times D \) times (checking all attributes of data point with centroid). Since, the similarity/dissimilarity between the data points is computed based on their attributes, increase in number of attributes/dimensions of data, escalates the computational complexity of Equation 1 and becomes NP hard problem [13]. Hence efficacy of traditional clustering algorithm degrades with the increase in dimensions.

In the last few years, studies have proved the aptness of nature inspired algorithms to find near optimal solutions for NP hard problems. These are meta-heuristic algorithms inspired by the processes observed from nature. The nature may include biological (e.g. foraging behavior of bees, flocking of birds etc.) as well as non biological (e.g. physical and chemical process i.e. intelligent water drop, gravitation search etc.) processes [11]. Among these, algorithms inspired by the characteristics of biological processes termed as Bio-Inspired algorithms are more popular. Bio-inspired algorithms are further categorized as swarm intelligence algorithms and evolutionary algorithms. Swarm intelligence based algorithms are inspired by the collective behavior of natural swarms. In these algorithms, multiple self organized agents (swarms) work in unison to achieve the desired result. In the presented work, modified and hybrid variants of nature inspired algorithms are developed to find near optimal clusters in small as well as high dimensional data for simple and complex problems.

The report is divided into following subsections: section 2 represents the current status and research issues. Section 3 illustrates the description of research work. Section 4 describes significant contributions of research work and section 5 gives conclusion and future work followed by publications and references.

2. CURRENT STATUS AND RESEARCH ISSUES
Nature inspired algorithms (NIA) are widely used for solving the variety of optimization problems. These are capable of finding near optimal solution to NP-Hard problems. In order to understand the current status of nature inspired algorithms, literature survey has been performed. It is observed that importance of nature inspired algorithms has been realized worldwide. Research communities are involved in either designing and developing new NIA or modifying existing algorithms or applying them to solve various real world problems in diverse domains.

The various research gaps observed from literature review are as follows:

iii. There is tradeoff between dimensionality and complexity of the problem. Algorithms solve either complex problems with small dimensions or simple problems with high dimensions.

iv. Conventional NIA algorithms solve simple benchmark problems up to 1000 dimensions while clustering problems up to 101 dimensions only.

Thus, handling high dimensional problems with good performance efficacy is a challenging problem. It has been proved in literature that partition based clustering techniques are more suitable for solving scalability problem in high dimension data [12]. However, there are few inherent drawbacks of these techniques:

iii. Partition based clustering algorithms may get easily trapped in the local optima. Due to this they are not able to achieve global optimal solutions. Hence optimal clusters are not guaranteed.

iv. Performance of these algorithms depends on initial seed chosen. For example initial centroids chosen in K-Means algorithm decide the quality of clusters formed during algorithm’s execution.

The above defined problems become more intricate when number of dimensions of dataset is increased. Handling data with large number of attributes becomes a complex problem [2]. Hence high dimensional clustering poses number of challenges [9] which are as follows:

v. Though traditional clustering algorithms perform clustering in full dimension space, their performance degrades with the increase in dimensions. This problem is termed as “curse of dimensionality” [8], [9].

vi. Distance measure becomes meaningless as there is inherent sparsity in data space.
vii. All dimensions may not be important for all clusters. Clusters may exist in various subsets of dimensions i.e. subspaces. Hence finding clusters in different subspaces of high dimensional data is a hard problem.

Inherent challenges of clustering and research gaps identified in nature inspired algorithms forms the problem statement of our research work i.e. **“To Design and Develop Novel Variants of Nature Inspired Algorithms for Clustering High Dimensional Data”**.

3. DESCRIPTION OF RESEARCH WORK

On the basis of research gaps identified, the complete research work is divided into five approaches for addressing the challenges of partition based clustering and high dimensional clustering. Novel variants of nature inspired algorithms are developed in order to find algorithm that perform well on complex functions as well as optimally cluster high dimensional data.

**Approach 1:** In the first approach, a hybrid of K-Means and Flower Pollination algorithm (FPA) named KFPA is developed to overcome the inherent challenges of partition based clustering and improve clustering efficacy. FPA is a population based nature inspired algorithm inspired by the pollination behavior of natural flowering plants. The objective is to find most fitted flower in the population. In KFPA, each flower of the population is represented by centroids of clusters. There are \( n \) flowers where \( n \) is the population size. Initial flowers are obtained by performing a single iteration of K-Means algorithm. Hence this algorithm is not dependent on initial centroids chosen. Fitness of each flower is computed through an objective function i.e. minimizing intra cluster distance of clusters. FPA is an iterative algorithm where in every iteration each flower is updated by either local pollination or global pollination depending upon the switch probability. If updated flower gives better centroid positions than existing one i.e. better objective function value, then it is substituted with its constituent flower in the population else discarded. The process continues unless a termination criterion is satisfied. This way of hybridization improve clustering efficacy through compactness. The performance evaluation of KFPA reveals that:

- KFPA is suitable for clustering simple datasets with small dimensions. Datasets are simple when most of the clusters formed are spherical in shape.
• KFPA performs better than Bat algorithm and Firefly algorithm.
• Although KFPA performs better than its contemporary algorithm but global minima has not been attained. Hence there is scope for improvement in KFPA.

**Approach 2:** In this approach, KFPA is further enhanced to develop modified flower pollination algorithm (MFPA). In the evolutionary process of KFPA, the solution is retained in the population if fitness is not improved through local or global pollination. Due to this algorithm may not achieve global optima. Therefore in MFPA, the most infertile flower in the population is replaced with a random flower to enhance the performance efficacy and diversification capability of the algorithm. This might iterate the solutions towards optimal clusters. This step could improve the exploration property of algorithm with some probability and increases the search space of finding optimal solution. The modified version of flower pollination algorithm is integrated with K-Means algorithm (MFPA-C) and it has been found to perform better than contemporary nature inspired algorithms.

The following outcomes are retrieved from performance evaluation of MFPA-C:

1) MFPA-C provides superior performance than flower pollination algorithm, bat algorithm and firefly algorithm by 50%, 97% and 99% respectively.
2) MFPA-C is suitable for clustering complex datasets with small dimensions. Datasets are complex when clusters formed are arbitrary shape with complex distribution of data.
3) Since MFPA is a new modified bio-inspired algorithm and has been evaluated on clustering application for only small dimensions. So, the next objective is to evaluate the efficacy of MFPA on standard benchmark functions for higher dimensions.

**Approach 3:** It has been observed from literature that efficacy of nature inspired algorithms is evaluated on benchmark test functions and applications. Since MFPA has already been tested on clustering, so in this approach it is assessed on standard benchmark functions. Here CEC2014 [137] complex benchmark functions are chosen. These test functions are composed of 30 functions that are subdivided into four subclasses named as: unimodal functions, simple multimodal functions, hybrid functions and composition functions. Algorithms evaluate every benchmark function on 10, 30, 50 and 100 dimensions.
MFPA is evaluated against artificial bee colony (ABC) algorithm [53], cuckoo search algorithm (CSA) [133], bat algorithm (BA) [124], firefly algorithm (FFA) [54] and flower pollination algorithm (FPA) [47]. The empirical study revealed that MFPA though perform well for clustering complex small dimension dataset, it does not give considerable performance on high dimensional complex benchmark functions. In order to find the best algorithm on such benchmark functions, an in-depth comparative analysis of five existing nature inspired algorithms i.e. ABC, CSA, BA, FFA and FPA is made. For fair comparison between algorithms, following parameters are used:

1) Self adaptive version of each algorithm is used. That means, parameter values are self tuned during algorithm’s execution.
2) Initial random solution at the starting of each algorithm is taken same.
3) Total number of fitness function evaluations for all algorithms is taken same.
4) Same number of independent runs is considered for all algorithms.

The results are compared on the basis of best, worst, mean, median and standard deviation of error values for 50 independent runs. Best algorithm for each type of function is identified. In order to substantiate the results following analysis are made on results obtained by different algorithms:

i) Computational time complexities of algorithms are analyzed.
ii) Statistical significance of results obtained is established through Wilcoxon rank sum test.

From overall comparison and analysis of results it has been inferred that ABC emerged as winning algorithm on high dimensional problems. Second rank is occupied by FPA and CSA as they depict comparable performances. However, optimal values of CEC2014 benchmark functions have not been achieved. Hence there is a scope of improvement.

**Approach 4:** Based on the observations of previous approach, a hybrid of artificial bee colony (ABC), differential evolution (DE) and flower pollination algorithm (FPA) is developed named ABC_DE_FP. Since, ABC emerged as best algorithm while FPA is the second best algorithm, FPA and Differential evolution (DE) algorithm are incorporated in ABC algorithm. Differential evolution algorithm (DE) is an evolutionary algorithm, known for its good convergence speed [46]. Proposed algorithm ABC_DE_FP is divided into three phases via different kinds of bees:
employed bee phase, onlooker bee phase and scout bee phase. The objective of algorithm is to find flower with maximum nectar amount. Food sources in artificial bee colony algorithm become flowers. Onlooker bees of ABC act as pollinators in flower pollination and pollination is global in nature. Exploitation is improved through mutation and crossover strategies of DE algorithm. FPA helps in escalating the exploration capability of ABC algorithm. In the employed bee phase, updation of solution is made through mutation and crossover strategies of differential evolution algorithm while in the onlooker bee phase, evolutionary mechanism of flower pollination is applied. On the basis of switch probability, exploration is performed through global pollination of FPA while exploitation is performed through mutation and crossover strategies of DE. Scout bee phase remains same as in ABC algorithm. Thus, hybrid algorithm ABC_DE_FP maintains proper balance between exploration and exploitation processes.

The efficacy of ABC_DE_FP is tested on continuous CEC2014 benchmark functions against state of art ABC variants. Number of fitness function evaluations is taken as termination criteria for ABC_DE_FP and other compared algorithms. Each algorithm is executed 50 times independently and experimental results are compared in the form of mean error value of objective functions. Further analysis of experimental results of ABC_DE_FP versus ABC variants is made on basis of:

i) Convergence speed of algorithms in the form of run length distribution graphs over the number of fitness functions evaluated.

ii) Statistical significance of results obtained by ABC_DE_FP and other compared algorithms is made via Wilcoxon rank sum test.

It has been deduced from the experimental results and analysis that proposed algorithm ABC_DE_FP scales well with increasing dimensions of problem. It outperform all ABC variants on high dimensional (100 dimensions) complex benchmark functions while it gives comparable result on small dimensions (10, 30 and 50 dimension). The algorithm maintains proper balance between exploitation and exploration as it depicts good convergence speed. Thus, the algorithm ABC_DE_FP successfully performs well on continuous high dimensional complex benchmark functions. The next objective is to evaluate ABC_DE_FP for clustering high dimensional data.

**Approach 5:** In this approach a hybrid of ABC_DE_FP and subspace clustering algorithm is developed to perform clustering in high dimensional data. Subspace clustering plays an
important role as in high dimensional data, clusters exists in various subspaces. Subspaces are subsets of dimensions and dimensions may overlap in different subspaces. There are two desirable properties of subspace clustering: firstly, algorithms should find all clusters in which a point participates and secondly, algorithms should form maximal subspace clusters without any redundant information. Existing subspace clustering algorithms satisfy either of the properties but not both. Also these algorithms become computationally expensive in high dimensional data. Hence, a nature inspired subspace clustering named S_FAD algorithm is proposed which satisfies both the desirable properties of subspace clustering and scales well on high dimensional data. In this approach, initially each data point is assigned a unique large random integer called signature. S_FAD uses bottom up subspace search method, thus it starts clustering from one dimensional data through self adaptive DBSCAN algorithm. Since DBSCAN is sensitive to input parameters hence we developed its self adaptive version by extracting the parameters i.e. minpts and epsilon using binary ABC_DE_FP algorithm. Once clusters in each dimension are formed, cluster signature is generated. A cluster signature is sum of signatures of its respective data points. Each cluster entry is made into hash table with signature, data points and dimensions. The same process is repeated for all dimensions of dataset and cluster information is noted in hash table. Thereafter, the rows with same signatures and same dimensions (subspaces) are merged in hash table. In this way, maximal subspaces are obtained and no redundant subspace is left. In order to obtain cluster in each subspace, self adaptive DBSCAN clustering is applied on each row of hash table using its respective subspace dimensions.

S_FAD algorithm is evaluated against state of art subspace clustering algorithms on 6 artificial and 11 real datasets. Evaluation parameters considered for comparing the algorithms are F1_Measure and accuracy. On the basis of these evaluation parameters, analysis of S_FAD versus other subspace clustering algorithms is made in terms of following parameters:

i) Average ranking – In this ranking method, accuracy and F1_measure values defined for each dataset by every algorithm is sorted and assigned the ranks. Thereafter, average ranks of each algorithm on all datasets are calculated.

ii) Success rate ratio ranking- SRR is a ranking method where ratio of success rates is computed between the pairs of algorithms. This method aids in determining the significant differences in algorithms.
iii) Statistical difference in results obtained by S_FAD versus other subspace clustering algorithms is made via Wilcoxon signed rank test.

iv) Scalability - In order to analyze the performance of algorithms with increasing dimensions, scalability graphs on dimensions are plotted and analyzed.

Experimental results and analysis reveals that S_FAD gives comparable performance to the existing subspace clustering algorithms on small dimensional datasets. While it gives high accuracy and F1_Measure on high dimensional datasets. Thus following outcomes are revealed in this approach:

a) S_FAD is successful in forming subspace clusters in high dimensional data where other existing subspace clustering algorithms fails. It scales well up to 6400 dimensional real dataset.

b) S_FAD can finds overlapping subspace clusters of arbitrary shape.

c) It removes redundant subspaces by determining maximal subspaces through hashing.

d) No normalization in dataset is required as parameters of DBSCAN are determined from binary ABC_DE_FP.

e) S_FAD can form subspace clusters of varied densities.

4. SIGNIFICANT CONTRIBUTIONS

Current research provides the algorithms for clustering small as well as high dimensional data. The work focuses on nature inspired algorithms whose exploration and exploitation capabilities aid in enhancing the scalability of the clustering techniques. The algorithms developed in this research work are useful in clustering small dimensional simple and complex datasets. Also, proposed algorithms scales well on high dimensional complex benchmark functions and clustering datasets. This research may be useful in number of fields where data is complex and high dimensional clustering is required e.g. recommender system, image processing, computer vision etc.

Main contributions of research work are as follows:
• In order to overcome challenges of partition based clustering, a hybrid of K-Means and flower pollination algorithm (KFPA) is developed. KFPA is well suited to the simple problems with small dimensions.

• A new modified version of FPA (MFPA) is developed. It performs well on complex small dimensional clustering datasets against contemporary nature inspired algorithms.

• An empirical analysis of five existing nature inspired algorithms is made on real parameter optimization. Results reveal that artificial bee colony algorithm (ABC) is the best algorithm when analyzed on high dimensional complex benchmark functions. Cuckoo search gives better results on small dimension problems. However, its efficacy degrades with increase in dimensions. Results also reveal that Bat algorithm takes least time as compared to other algorithms but it could not attain global optimal value. Hence, on the basis of overall observations, ABC is better algorithm as compared to others and is followed by flower pollination and cuckoo search algorithm as they depict comparable performances.

• A new hybrid algorithm ABC_DE_FP (combination of artificial bee colony, differential evolution and flower pollination algorithm) is developed. This algorithm outperforms state of art ABC variants on complex high dimensional benchmark functions.

• In order to perform clustering in high dimensional data and find all clusters existing in various subspaces, a nature inspired subspace clustering algorithm (S_FAD) is developed. S_FAD uses binary version of ABC_DE_FP and finds overlapping subspace clusters of varied densities. This algorithm gives number of advantages over existing subspace clustering algorithms and scales upto 6400 dimensional dataset.

6. CONCLUSION AND FUTURE WORK

The research work presents novel variants of nature inspired algorithm that targets the challenges of partition based clustering and high dimensional clustering. Objective also focuses to develop the algorithms that can successfully handle complex high dimensional problems. The complete research work is divided in five approaches. In first approach, flower pollination algorithm
(FPA), a contemporary nature inspired algorithm is integrated with K-means (KFPA) for improving the clustering efficacy of K-Means for simple problems with small dimension. A variant of KFPA that is modified FPA (MFPA) is developed to enhance the clustering efficiency against contemporary nature inspired algorithms. Experimental results established that MFPA perform well on small dimensional complex clustering datasets. To further substantiate the observations, MFPA is tested on CEC2014 complex numerical benchmark problems upto 100 dimensions. However, contrary to our expectations, MFPA did not perform well on these problems. Thus, a study was performed to find best algorithm for solving simple and complex numerical benchmark problems. Five well known existing nature inspired algorithms are explored and analyzed. On the basis of observations, a new hybrid nature inspired algorithm is developed which is termed as ABC_DE_FP. It is a hybrid of artificial bee colony (ABC), FPA and differential evolution (DE) algorithm. ABC_DE_FP outperformed the state of art ABC variants on high dimensional complex benchmark functions. In the next approach ABC_DE_FP is adapted to cluster high dimensional data. In order to handle high dimensional clustering challenges, binary version of ABC_DE_FPA is integrated with subspace clustering and a new algorithm is developed named S_FAD. The algorithm is evaluated against state of art subspace clustering algorithms and found to perform better on high dimensional datasets. Also S_FAD is successful in finding overlapped subspace clusters of varied densities of up to 6400 dimensional real dataset. The algorithm provides number of benefits as compared to existing algorithms. In S_FAD no normalization of original dataset is required. The algorithm finds maximal subspaces and overlapping clusters of arbitrary shapes. Thus, S_FAD can be applied in any high dimensional application where accuracy is main concern.

Future plan of research work can be as follows:

i. Algorithms developed in research work can be applied in various applications such as recommender system, text mining, social network analysis etc.

ii. In distributed databases, the algorithms can be used for vertical fragmentation of data.

iii. Algorithms may be enhanced to solve combinatorial problems like TSP, 0-1 knapsack problem, etc.

iv. Parallel version of algorithms can be developed to handle the big data.