

A Review on Genetic Algorithm and Its Applications in Data Mining

Divya Sindhu¹, Saurabh Sindhu²Lecturer, Department of Computer Science, CRM Jat College, Hisar, Haryana, India¹Lecturer, Department of Computer Science, CRM Jat College, Hisar, Haryana, India²

ABSTRACT: The generation of voluminous information due to recent technological advances all over the world, has necessitated the requirement of computerized databases to store, categorize and index the data. Further, the analysis of the data using specialized tools and algorithms, visualization of the results and evaluation of the produced knowledge are equally important steps in the knowledge discovery process. Genetic algorithm is a metaheuristic method inspired by the laws of genetics, which find useful solutions to complex problems. In this method, first some random solutions (individuals) are generated each containing several properties (chromosomes). Based on the laws of crossover and mutations, which occur in chromosomes, a second generation of individuals (population) is produced with more diverse properties. This novel and efficient computational algorithm is used to overcome the constraints posed by the traditional statistical methods. Genetic algorithm has applications in all the field of science including data mining, bioinformatics, phylogenetics, engineering, finance, business, agriculture as well as medical sciences.

KEYWORDS: Genetic algorithm, Data mining, Chromosomes, Population, Crossover, Mutation, Fitness function, Optimization

I. INTRODUCTION

Genetic algorithms (GAs) are adaptive and probabilistic search techniques used in computing to find exact or approximate solution to optimization and search problems. Genetic algorithms are categorized as global search heuristics and evolutionary optimization approach. These algorithms are inspired by Charles Darwin's theory about evolution in natural populations according to the principles of natural selection and "survival of the fittest" over many generations. After a number of new generations built with the help of the genetic algorithms, one obtains a solution that cannot be improved any further [1].

Genetic algorithm was developed by John Holland and his colleagues in Michigan University and it was based on the genetic processes of biological organisms [2]. In general, GA is a search algorithm based on the natural selection and genetics [3]. It uses a number of artificial individuals looking through a complex search space by using functions of selection, crossover and mutation (Fig. 1). The purpose to use GA is searching and finding optimal or good enough solution. This solution is hidden in a big search space and there is no guarantee to find any exact solutions when using GA. Genetic algorithms are different from the derivative-based, optimization algorithms. First of all, these algorithms search a population of points in the solution space in each iteration while classical derivative-based methods search only a single point. Moreover, GAs select the next population using probabilistic transition rules and random number generators while derivative-based algorithms use deterministic transition rules for selecting the next point in the sequence [4].

Genetic algorithm is one of the evolutionary algorithms (EAs) that implements optimization strategies by simulating evolution of species through the process of natural selection, survival of the fittest and reproduction [5, 6]. An evolutionary algorithm is a stochastic search for an optimal solution to a given problem [7] and it uses a population of individuals; where each individual is classified as a chromosome. Chromosome represents the characteristics of individuals in the population; these characteristics are thus referred as gene [8]. Solutions from one population are taken and used to form a new population. This is motivated by a hope, that the new population will be better than the

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

old one. Solutions which are selected to form new solutions (offspring) are selected according to their fitness: the more suitable they are, the more chances they have to reproduce. Genetic algorithms include evolutionary programming, evolutionary strategies [9], differential evolution [10], cultural evolution and coevolution [11].

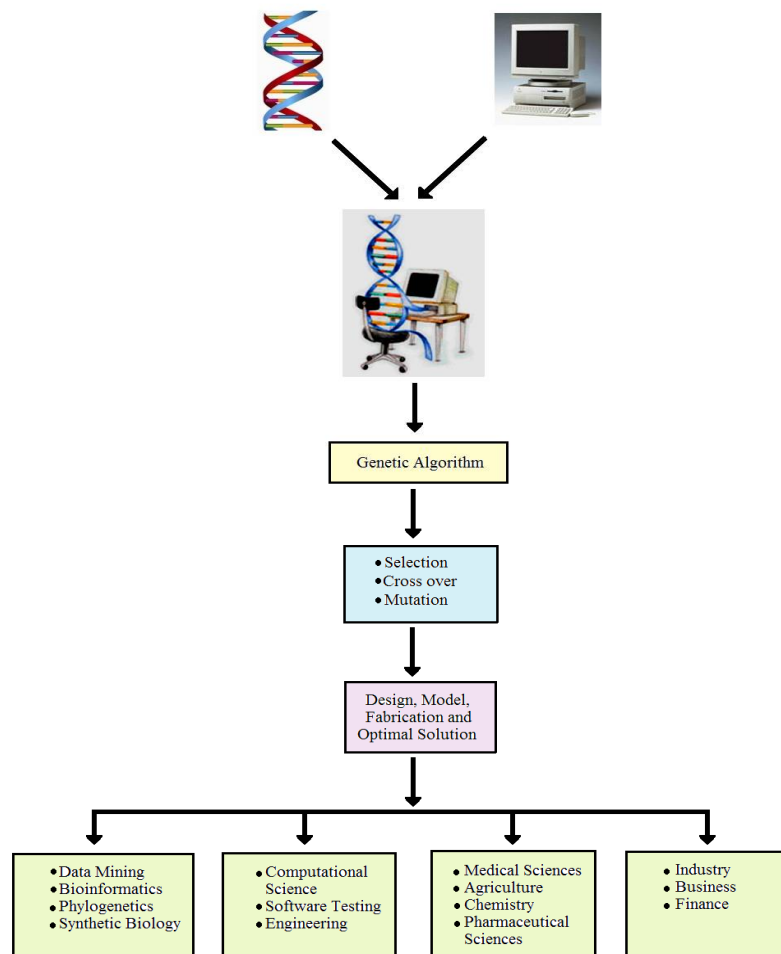


Figure 1. Genetic algorithm and its applications in various fields

II. DATA MINING

Data mining is an interdisciplinary area of research that has its roots in databases, machine learning and statistics, and has contributions from many other areas such as information retrieval, pattern recognition, visualization, parallel and distributed computing [12]. Data mining is the use of automated data analysis techniques to uncover previously undetected relationships among data items. It helps the scientists and the researches to extract the useful information from the huge amount of data at hand by providing sophisticated techniques [13]. This technique applied for the discovery of patterns hidden in large data sets, focusing on issues relating to their feasibility, usefulness, effectiveness and scalability. The applications of data mining include customer relationship management, market and industry characterization, medicine, engineering, agriculture and biology.

Various algorithms and techniques like classification, clustering, regression, artificial intelligence, neural networks, association rules, decision trees, genetic algorithm, nearest neighbor method etc., are used for knowledge discovery

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

from databases [14]. Genetic algorithms are used in various fields of data mining to get the optimized solutions for the better performance of the data that are required in decision making and to process the accurate result [15]. Initial population is generated by methodically eliminating the randomness through generalized uniform population method. In the subsequent generations, genetic diversity is ensured and premature convergence is prevented by the uniform operator.

III. METHODOLOGY USED IN GENETIC ALGORITHM

Genetic algorithm is a search based optimization technique [16] that evolves through a search space of candidate population to identify the best pool of population. To understand the concept and its application, some of the related terms used in genetic algorithm are as follows:

i. Chromosomes: These are the strings or numerical values which represent the candidate solution to the search problem [17, 18].

ii. Gene: The elements of the solution are known as the gene.

iii. Population: It refers to the total number of candidate solutions to be used in the optimization problem.

iv. Allele: The possible values of a gene are known as allele.

v. Genotype: It refers to the set of genes contained in a genome.

vi. Phenotype: These are the behavioral traits or characteristics of an individual [19].

The steps used in the methodology are as follows:-

- Learning the application domain to extract relevant prior knowledge and goals of application.
- Creating a target data set from raw data
- Data cleaning and preprocessing
- Data reduction and transformation
- Choosing data mining approach: Classification
- Choosing the mining algorithm(s): Genetic algorithm
- Patterns of interest found by genetic algorithm
- Evaluation of patterns and knowledge presentation, visualization and removing of redundant patterns, etc.
- Use of discovered knowledge

Evaluation of patterns and knowledge presentation are done after the data mining task. The significant patterns of population/masses are obtained after the execution of the genetic algorithm. These patterns are further evaluated and then knowledge is presented.

IV. STEPS INVOLVED IN IMPLEMENTATION OF GENETIC ALGORITHM

Genetic algorithms provide a comprehensive search methodology for machine learning and optimization. In genetic algorithm, the possibility of reproduction depends on the fitness of individuals. The better chromosomes they have (i.e., those with better characteristics), the more likely they are to be selected for breeding the next generation. There are several selection methods; however, the aim of all is to assign fitness values to individuals based on a fitness function and to select the fittest. Genetic alterations in chromosomes will happen via crossover and mutations to produce another generation. This iterative process will continue until the fittest individual (the optimal solution) is formed or the maximum number of generations is reached [20].

Genetic algorithm uses three main operators which are selection, crossover and mutation.

3.1. Selection: The selection process determines which solutions are to be preserved and allowed to reproduce, and which ones deserve to die out. The process normally involves identifying the good solutions in a population, making multiple copies of the good solutions and eliminating bad solutions from the population [21]. The final stage is applying the fitness function to evaluate the solutions. A fitness function represents the quality of the solution [22]. There are different types of selection methods used by GA, this include rank selection, proportionate selection, roulette wheel selection, tournament selection and steady selection [23].

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

3.2. Crossover operator: Based on the laws of crossover and mutations, which occur in chromosomes, a second generation of individuals (population) is produced with more diverse properties. Crossover and mutation are the two most central methods for diversifying individuals. In crossover, two chromosomes are chosen. Then a crossover point along each chromosome is chosen followed by the exchange of the values up to the crossover point between the two chromosomes. These two newly-generated chromosomes produce new offspring. The process of crossover will be iterated over and over until the desired diversity of individuals (i.e. solutions) is made.

The crossover operator is used to create new solutions from the existing solutions by recombination of the information from both parents [24]. It involves the exchange of genes randomly between the chromosomes of two parents to create new offspring [25]. Crossover can either be one point or two points.

3.1.2. Mutation: The final genetic operator is mutation which introduces new features to a given population or problem to obtain and maintain diversity in the population. It thus involves randomly changing individuals to allow diversity among the population. The various types of mutation are flip mutation, uniform mutation inversion mutation, scramble mutation, swap mutation, interchanging mutation and creep mutation [26]. The mutation also generates new configurations by applying random changes in different chromosomes [27].

The construction of genetic algorithm will include the following steps (Fig. 2):

- (i) **Start:** Generate initial random population of n chromosomes (suitable solutions for the problem).
- (ii) **Fitness:** Evaluate the fitness $f(x)$ of each chromosome x in the population.
- (iii) **New population:** Create a new population (offsprings) by repeating following steps until the new population is complete.
 - **Selection:** Select a pair of two parent chromosomes from the current population according to their fitness (the better fitness, the bigger chance to be selected).
 - **Crossover:** With a crossover probability, cross over the parents at randomly chosen points/sites to form new offsprings (children). If no crossover was performed, offspring is an exact copy of the parents.
 - **Mutation:** With a mutation probability, mutate new offsprings at each locus (position in chromosome) and add the resulting chromosomes to the population.
- (iv) **Accept:** Calculate the fitness of the resulting chromosomes and place new offsprings in a new population or
- **Replace:** Let the finest chromosomes survive to next generation. Use new generated population for a further run of algorithm
- (v) **Stop:** If the end condition is satisfied, stop and return the best solution in current population or go to step (ii).

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

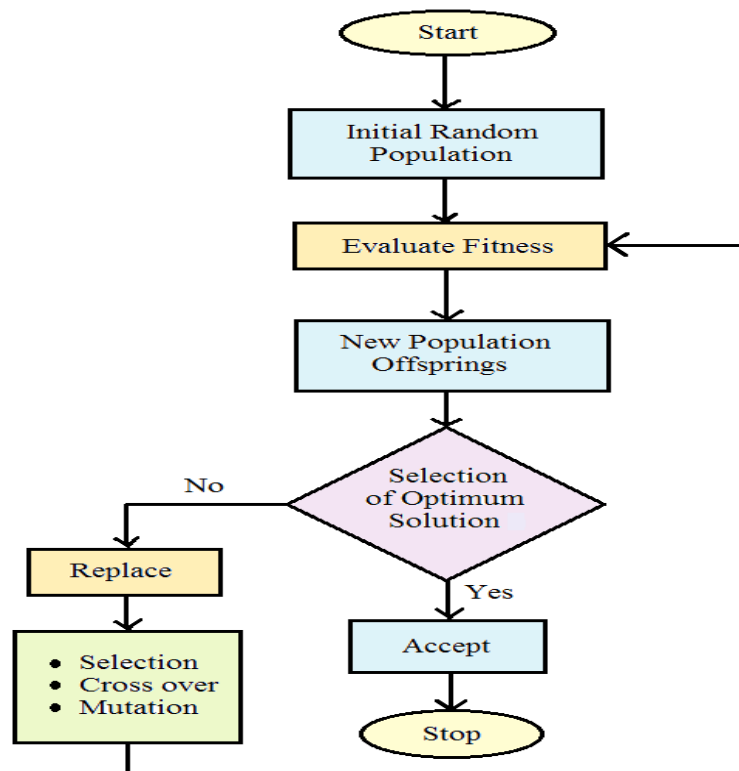


Figure 2. Flow chart for processing of genetic algorithm

V. APPLICATIONS OF GENETIC ALGORITHM

Computers have revolutionized our everyday life and they have benefited nearly all fields of science from aerospace and astronomy to biology, chemistry, physics, mathematics, geography, archaeology, engineering and social sciences. Notable research has focused on advances in genetic algorithms and related hybrid approaches, with application in various problem areas. Genetic algorithm-based approaches offer a more user-friendly, flexible and adaptable population-based approach than related algorithms. Given these advantages, genetic algorithms are being applied in several fields and some of their applications are described in this chapter under four categories.

5.1. Extraction of knowledge pattern and data mining in bioinformatics and phylogenetics

(i) Data mining: Data mining is one of the important application field of genetic algorithm [28, 29]. In data mining, genetic algorithm can be used either to optimize parameters for other kind of data mining algorithms or to discover knowledge by itself. The advantages of genetic algorithm become more obvious when the search space of a task is large. This technique is used in computing to find exact or approximate solution to optimization and search problems. The knowledge discovery process in databases utilizes the genetic algorithm [30].

(ii) Bioinformatics: Rapid developments in genomics and proteomics in recent years have generated a large amount of biological data [31]. Bioinformatics is the interdisciplinary science of interpreting and analysis of biological data using information technology and computational techniques [32]. It is the science of managing, mining, integrating and interpreting information from biological data at the genomic, proteomic, phylogenetic, cellular or whole organism levels [33, 34]. The need for bioinformatics tools and expertise has increased as genome sequencing projects have resulted in an exponential growth in complete and partial sequence databases. Analysis of large biological datasets

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

requires protein structure prediction, gene classification, cancer classification based on microarray data, clustering of gene expression data and statistical modeling of protein-protein interactions etc.

(iii) Phylogenetics: Recombination is a major evolutionary force in many organisms and can have a profound impact on evolutionary rates. Maximum likelihood methods of codon substitution, used to estimate selection pressures on sites in terms of the ratio of non-synonymous to synonymous substitution rates may generate many false-positive sites when recombination is not taken into account. Hence, screening of recombination should be an integral part of phylogenetic analyses and the use of phylogenetic incongruence among fragments of a sequence alignment to detect recombination is being used [35, 36]. Brauer et al. [37] tested parallel genetic algorithm with large (228 taxa) data sets of both empirically observed DNA sequence data (for angiosperms) as well as simulated DNA sequence data. For both observed and simulated data, search-time improvement was found nearly linear with respect to the number of processors, so the parallelization strategy appears to be highly effective at improving computation time for large phylogenetic problems using the genetic algorithm.

Pond et al. [38] proposed a model based framework that uses a genetic algorithm to search a multiple-sequence alignment for putative recombination break points, quantifies the level of support for their locations and identifies sequences or clades involved in putative recombination events. The software implementation can be run quickly and efficiently in a distributed computing environment and various components of the methods can be chosen for computational expediency or statistical rigour. The performance of the new method on simulated alignments was evaluated on an array of published benchmark data sets. Authors demonstrated that prescreening of alignments with this method allows the analysis of recombinant sequences for positive selection.

5.2. Applications in software testing, computer science and engineering

(i) Software testing: Manual and automated testing are the two main testing techniques that have been employed over the past decades. However, automated testing appears to be the most efficient and effective method, due partly to its efficiency and effectiveness in maximizing test coverage, detecting more errors, increasing test execution, decreasing the cost and improving the quality of the product. Based on the randomly generated test case from the population or input domain, a fitness function is used to evaluate the test cases to determine if it can be used for future generation of more test cases. The test case is executed and feedback information is collected. All test cases with high fitness value were selected and added to the best test pool. Test cases with low fitness value are discarded and not used for future generation or can be recombined and mutated to produce new test cases. The newly created offsprings are evaluated against the fitness value and this process continues until a stopping condition is achieved.

Genetic algorithm can be used to automatically generate test case using the simple random testing technique that allows a test case to be selected randomly from the input domain [39, 40]. The application of genetic algorithm in software testing is an active area of scientific research and the application of evolutionary algorithm based on test case generation using the simple random testing technique was recently reviewed [41]. The findings showed that genetic algorithm performs better than other automated test generation techniques such as random testing and also increases the number of test cases, thereby improving efficiency. The use of genetic algorithm may also be investigated in regression testing as well as test case prioritization. Finally, a more robust way of improving the fitness function can be investigated.

Alander and Mantere [42] studied program of testing automation by using optimization via genetic algorithm and showed that genetic optimization performs better than simple random testing. Tripathy and Kanhar [43] proposed a heuristic algorithm with sampling techniques to optimize test suite. The genetic algorithm was used as an element to optimize the test suite. McCart et al. [44] used genetic algorithm to improve the testing technique process by reducing the execution time as well as improving the ability of the technique in detecting more errors. Wappler and Lammermann [45] proposed an evolutionary algorithm for the automatic generation of test cases for white-box testing of object-oriented programs. The experimental result of their approach out-performed random testing. Rauf et al. [46] proposed a graphical user interface (GUI) testing and coverage analysis technique using genetic algorithm to help generate the best possible test parameter based on a predefined testing goals. They used some of the modern tools and techniques of automating GUI testing such as control-flow graph, event-flow graph and event-flow model etc. to help reduce the challenges associated with testing of such systems.

(ii) Computer science: The Traveling Salesman Problem (TSP) is an optimization problem in which a salesman has to visit a given set of cities. The time complexity of the algorithm to find such a tour increases, as the number of cities

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

increases [47]. Many meta-heuristic techniques such as genetic algorithm, particle swarm optimization (PSO) and ant colony optimization (ACO) are used to solve TSP. Genetic algorithm depends upon its genetic operators such as selection, reproduction and mutation to solve any optimization problem. TSP is a minimization problem to find a route with minimum distance to visit all the cities. Genetic algorithm starts with an initial population of chromosomes. Chromosomes are generated randomly when the problem is represented in genetic form. Then, a bunch of chromosomes is selected from the initial population by using selection operator. This bunch of chromosomes participates in reproduction operation to generate new children. These newly generated children are added in the initial population and then mutation operation is performed to generate the next population. These operations i.e. selection, reproduction and crossover are repeated again and again for a given number of iteration count or till an optimal solution is not found.

An optimal and valid solution for Traveling Salesman Problem was proposed [48] and authors suggested that TSP can be used in many areas of engineering. They used Handle-C language to find improved and hybrid genetic algorithm to solve TSP. Authors implemented the handle C technique on various TSPLIB instances and found that the proposed algorithm outperforms as compared to some other existing algorithms. The improved genetic algorithm uses a stronger global search technique that gave higher rate of convergence of genetic algorithm. It means that genetic algorithm find tour with less path length in less time as compared to other existing algorithms.

Evolutionary computation techniques are commonly used to address problems with large search spaces, where an exhaustive search is not applicable in practice. Due to this fact, many machine learning and optimization tasks have been solved by using genetic algorithms or evolutionary algorithms. Giráldez et al. [49] briefly summarized some of the techniques for evolutionary algorithms to solve estimation and modelling problems and to provide decision rules using less computational resources. Guan and Zhu [50] explored an incremental approach to genetic-algorithms-based classification rather than neural networks to cope with learning tasks where the learning environment is ever changing or training samples become available over time.

(iii) Engineering: An integrated optimization of mechanisms with genetic algorithm was offered. The principle of which is to use a neural network as a global calculation program and to couple the network with stochastic methods of optimization [51]. Genetic algorithms rely on the natural laws of selection which let a living organism adapt to a given environment. From these principles, it seems judicious to apply genetic algorithms to the optimization of mechanical structures. Precise examples showed that genetic algorithms will allow the adaption of the mechanical object to its environment and to the specifications from the beginning of the design process.

5.3. Application of genetic algorithms in industry and business

(i) Industry: Data mining techniques can yield the benefits of automation on existing software and hardware platforms, and can be implemented on new systems as existing platforms are upgraded and new products developed. Faster processing and analysis of massive databases indicates that users can automatically experiment with more models to understand complex data. High speed makes it practical for users to analyze huge quantities of data. Thus, data mining technology can generate new business opportunities by providing these capabilities:

- **Automated prediction of trends and behaviours:** Data mining automates the process of finding predictive information in large databases. A typical example of a predictive problem is targeted marketing. Data mining uses data on past promotional mailings to identify the targets most likely to maximize return on investment in future mailings. Other predictive problems include forecasting bankruptcy and other forms of default.
- **Automated discovery of previously unknown patterns:** Data mining identifies previously hidden patterns through the analysis of retail sales data to identify seemingly unrelated products that are often purchased together. Other pattern discovery problems include detecting fraudulent credit card transactions and identifying anomalous data that could represent data entry keying errors.

(ii) Business: Commercial databases are growing at unprecedented rates. The accompanying need for improved computational engines can now be met in a cost-effective manner with parallel multiprocessor computer technology. Data mining algorithms have only recently been implemented as mature, reliable and understandable tools that consistently outperform older statistical methods. In the evolution from business data to business information, each new step has been built upon the previous one. For example, dynamic data access is critical for drill-through in data

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

navigation applications and the ability to store large databases is critical to data mining. Data mining tools predict future trends and behaviours, allowing businesses to make proactive, knowledge-driven decisions.

Comprehensive data warehouses that integrate operational data with customer, supplier and market information have resulted in an explosion of information. Data mining techniques can be implemented rapidly on existing software and hardware platforms to enhance the value of existing information resources and can be integrated with new products and systems as they are brought on-line. When implemented on high performance client/server or parallel processing computers and data mining tools can analyze massive databases. This type of finding helps business persons to make certain decisions, such as catalogue design, cross marketing and customer shopping behavior analysis. Association rule algorithms need to generate rules with confidence values less than one. However the number of possible association rules for a given dataset is generally very large. Quantifiable business benefits have been proven through the integration of data mining with current information systems. A wide range of companies have deployed successful applications of data mining. Some successful application areas include:

- A pharmaceutical company can analyze its recent sales force activity and their results to improve targeting of high-value physicians and determine which marketing activities will have the greatest impact in the next few months. The data needs to include competitor market activity as well as information about the local health care systems.
- A credit card company can leverage its vast warehouse of customer transaction data to identify customers most likely to be interested in a new credit product. Using a small test mailing, the attributes of customers with an affinity for the product can be identified.
- A diversified transportation company with a large direct sales force can apply data mining to identify the best prospects for its services. Using data mining to analyze its own customer experience, this company can build a unique segmentation identifying the attributes of high-value prospects.
- A large consumer package goods company can apply data mining to improve its sales process to retailers. Data from consumer panels, shipments and competitor activity can be applied to understand the reasons for brand and store switching. Through this analysis, the manufacturer can select promotional strategies that best reach their target customer segments.
- Companies may leverage the knowledge about customers implicit in a data warehouse to reduce costs and improve the value of customer relationships. These organizations can now focus their efforts on the most important (profitable) customers and prospects, and design targeted marketing strategies to best reach them.

5.4. Genetic algorithm applications in agriculture, chemistry, drug design and medical sciences

(i) **Agriculture:** Agriculture plays a crucial role in the development of the Indian economy. The major aspect for improving agriculture productivity is on embedding agriculture information system with computer technologies to extend the services that will provide better farming facilities to the farmers who are residing in distant locations and not aware of new technology of nowadays. The transfer of agriculture related information helps in collection of relevant data about the farming techniques, crop, type of soil and so that farmers are provided with the best farming methods. The aim of genetic algorithm is to provide the optimal solution to the farmer and this will help in improving quick decision making capabilities [52]. Using various data mining techniques, the problems of farmer may be solved by giving them the most optimal solution to their problem [53].

Nowadays, large companies use genetic algorithm to optimize and design products ranging from tiny chip to a agriculture system. In GA, there is a parallel computing which make it more appealing because it helps to reduce the problem of lack of speed in computing. It allows more number of iterations which will increase solution and provide ways for better solutions. The combining of genetic algorithm with other evolutionary computation algorithms such as fuzzy and neural network is going on. The results achieved with the help of genetic algorithms will lead towards a development strategy in the rural sector through agriculture. In a country like India whose rural economy is mostly agriculture based, a sustainable development could be in the context of globalization and environmental strategies by reorganizing land system for various agricultural activities.

(ii) **Chemistry:** Chemical engineering systems are widely and instinctively complicated. The vision and aim of a designer should be to unravel the hidden truths and the latent scientific vision and scientific understanding. Science and engineering will see a new dawn with the new and innovative ventures in the field of non-traditional optimization techniques. The evolution of genetic algorithm and the vast application areas surely overcome wide and versatile

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

visionary frontiers [54]. Vision of the application areas of genetic algorithm will soon surpass wide frontiers of science and engineering of petroleum refining. Lower computational time and improved efficiency of the processes will evolve a new dimension of hope and scientific justification.

In the present age of energy sustainability, the question of depletion of fossil fuel sources needs to be vehemently addressed with immense vigour and vision. Petroleum refining and energy sustainability in today's world have an umbilical cord whose scientific vision needs to be reassessed. Application of genetic algorithm and revamping of petroleum refining process needs to be addressed with immense scientific vision, caution and urgency. Fluidized catalytic cracking is a vital and primordial issues in the entire gamut of petroleum refining [55]. Palit et al. [56] reviewed a deep insight into newer technologies and subsequent application of multi-objective optimization in designing a fluidized bed catalytic cracking unit with the application of genetic algorithm.

(iii) Pharmaceutical sciences and drug design: Designing a drug is the process of finding or creating a molecule which has a specific activity on a biological organism. Drug design is difficult since there are only few molecules that are both effective against a certain disease and exhibit other necessary physiological properties, such as absorption by the body and safety of use. The main problem of drug design is therefore how to explore the chemical space of many possible molecules to find the few suitable ones. Computational methods can be used to assist and speed up the drug discovery process. Lameijer et al. [57] discussed the applications of evolutionary algorithms in drug design, in which evolutionary algorithms are used both to create new molecules and to construct methods for predicting the properties of real or yet unexisting molecules.

Genetic algorithms and neural networks are versatile methods for a variety of tasks in rational drug design, including analysis of structure–activity data, establishment of quantitative structure–activity relationships (QSAR), gene prediction, locating protein coding regions in DNA sequences, 3D structure alignment, pharmacophore perception, docking of ligands to receptors, automated generation of small organic compounds, and the design of combinatorial libraries [58]. Terfloth and Gasteiger [59] gave a brief overview of these applications of neural networks and genetic algorithms in drug design and provided an insight into the underlying principles of such methods. In contrast to classical statistical methods such as regression analysis or partial least squares analysis (PLS), artificial neural networks are ideally suited for use in drug design and QSAR. Moreover, genetic algorithms are utilized in the structure alignment, variable selection in QSAR studies, structure generation, design of combinatorial libraries, and docking of ligands to proteins.

(iv) Medical sciences: A great wealth of information is hidden amid medical research data that in some cases cannot be easily analyzed, if at all, using classical statistical methods. In medicine, electronic chips and computers are the backbones of a lot of imaging, diagnostic, monitoring and therapeutic devices. These devices are composed of several different hardware components and are managed and controlled by software, which in turn are based on algorithms. Algorithms run on a set of well-described rules and instructions that define a sequence of operations. Metaheuristic methods use algorithms that can more quickly solve complex problems, or they can find an approximate solution when classical methods are not able to find an exact one. Metaheuristic approaches have been frequently used in other fields of science where complex problems need to be solved and optimal decisions should be made. In medicine, valuable work has been done on these potent algorithms for offering solutions to the countless complex problems, which physicians encounter every day.

Ghaheri et al. [60] introduced genetic algorithm as one of these metaheuristics and reviewed some of its applications in medicine. Metaheuristic algorithms have been developed to offer optimal or near-optimal solutions to complex data analysis and decision-making tasks in a reasonable time. The use of genetic algorithm has promising implications in various medical specialties including radiology, radiotherapy, oncology, cardiology, endocrinology, surgery, obstetrics and gynecology, infectious diseases, rehabilitation medicine, neurology, pharmacotherapy and health care management. Thus, applications of the genetic algorithms in disease screening, diagnosis, treatment planning, pharmaco-vigilance, prognosis and health care management enabled the physicians to envision possible applications of this metaheuristic method in their medical career.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

V. CONCLUSION

Current research trends tend to show that there is a high potential for remarkable advances in genetic algorithms and its variants, specifically in grouping genetic algorithms. Genetic algorithm-based approaches offer a more user-friendly, flexible, and adaptable population-based approach than related algorithms. Given these advantages, further developments and advances in grouping genetic algorithms are quite promising.

REFERENCES

- [1] Hermadi, I. and Ahmed, M. A., "Genetic algorithm based test data generator," in: Evolutionary Computation Congress, 2003. pp. 85-91.
- [2] Holland, J. H., "Genetic algorithms," Scientific American, vol. 267, pp. 66-72, 1992.
- [3] Kosindrdech, N. and Daengdej, J., "A test case generation process and technique," Journal of Software Engineering, Vol. 4, pp. 265-287, 2010.
- [4] Krishnakumar, K. Goldberg, D. E., "Control-system optimization using genetic algorithms". Journal of Guide Control and Dynamics, Vol. 15, pp. 735-740, 1992.
- [5] Suman, S., "A genetic algorithm for regression test sequence optimization," International Journal of Advance Research in Computer and Communication Engineering, Vol. 1, 2012.
- [6] Goodman, E.D., "Introduction to genetic algorithms," In: Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation, pp. 205-226, 2014.
- [7] Geem, Z.W., Kim, J.H. and Loganathan, G., "A new heuristic optimization algorithm: harmony search," Simulation, Vol. 76, pp. 60-68, 2001.
- [8] Beasley, D., Bull, D. R. and Martin, R. R., "An overview of genetic algorithms: Part 1, fundamentals," University Computing, Vol. 15, pp. 56-69, 1993.
- [9] Beyer, H. G. and Schwefel, H. P., "Evolution strategies—A comprehensive introduction," Natural Computing, Vol. 1, pp. 43-52, 2002.
- [10] Qin, A. K., Huang, V. L. and Suganthan, P. N., "Differential evolution algorithm with strategy adaptation for global numerical optimization," IEEE transactions on Evolutionary Computation, Vol. 13, pp. 398-417, 2009.
- [11] Shennan, S., "Genes, memes, and human history: Darwinian archaeology and cultural evolution," 2002.
- [12] Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R., "Advances in knowledge discovery and data mining". AAAI Press/MIT Press, Menlo Park, California, USA, 1996.
- [13] Vishwakarma, P., Kumar, Y. and Nath, R. K., "Data mining using genetic algorithm", International Journal of Engineering Research and Development, Vol. 5, Issue 2, pp. 88-94, 2012.
- [14] Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H. and McLachlan, G. J., "Top 10 algorithms in data mining. Knowledge and Information System, Vol. 14, Issue 1, pp 1-37, 2008. doi:10.1007/s10115-007-0114-2.
- [15] Agarwal, R., "Genetic algorithm in data mining", International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 5, Issue 9, pp. 631-634, 2015.
- [16] Falkenauer, E., "Genetic algorithms and grouping problems", John Wiley & Sons, Inc., 1998.
- [17] Mitchell, M., "Genetic algorithms: An overview," Complexity, Vol. 1, pp. 31-39, 1995.
- [18] Sastry, K., Goldberg, D.E. and Kendall, G., "Genetic algorithms," In: Search methodologies, ed: Springer, pp. 93-117, 2014.
- [19] Engelbrecht, A. P., "Fundamentals of computational swarm intelligence", John Wiley and Sons, 2006.
- [20] Reeves, C., "Genetic algorithms. Handbook of metaheuristics", Springer, pp. 55-82, 2003.
- [21] Sivanandam, S. and Deepa, S., "Introduction to genetic algorithms", Springer Science and Business Media, 2007.
- [22] Baresel, A., Sthamer, H. and Schmidt, M., "Fitness function design to improve evolutionary structural testing," In: Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation, pp. 1329-1336, 2002.
- [23] Goldberg, D. E. and Deb, K., "A comparative analysis of selection schemes used in genetic algorithms," Foundations of Genetic Algorithms, Vol. 1, pp. 69-93, 1991.
- [24] Srinivas, M. and Patnaik, L.M., "Adaptive probabilities of crossover and mutation in genetic algorithms," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 24, pp. 656-667, 1994.
- [25] Bodenhofer, U., "Genetic algorithms: theory and applications," Lecture notes, Fuzzy Logic Laboratorium, Linz-Hagenberg, 2003.
- [26] Soni, N. and Kumar, T., "Study of various mutation operators in genetic algorithms," International Journal of Computer Science and Information Technologies, Vol. 5, pp. 4519-4521, 2014.
- [27] Hasancebi, O. and Erbatir, F., "Evaluation of crossover techniques in genetic algorithm based optimum structural design", Computer Structure, Vol. 78, pp. 435-448, 2000.
- [28] Verma, G. and Verma, V., "Role and applications of genetic algorithm in data mining", International Journal of Computer Applications, Vol. 48, Issue 17, pp. 5-8, 2012.
- [29] Naik, R. and Deepika, N., "Data mining system and applications: A study" International Journal of Computer Science and Mobile Computing, Vol. 5, Issue 12, pp. 103-110, 2016.
- [30] Chadha, P. and Singh, G. N., "Classification rules and genetic algorithm in data mining", International Journal of Computer Science and Technology and Software Engineering, Vol. 12, Issue 15, pp. 50-54, 2012.
- [31] Cristianini, N. and Hahn M., "Introduction to computational genomics", Cambridge University Press, 2006.
- [32] Sindhu, S. and Sindhu, D., "Data mining and gene expression analysis in bioinformatics", International Journal of Computer Science and Mobile Computing, Vol. 6, Issue 5, pp. 72-83, 2017.
- [33] Li, J., Wong, L. and Yang, Q., "Data mining in bioinformatics", IEEE Intelligent System, IEEE Computer Society, 2005.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

- [34] Sindhu, S. and Sindhu, D., "Development of computational tools for metabolic engineering", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, Issue 5, pp. 9208-9217, 2016.
- [35] Salminen, M., Carr, J., Burke, D. and McCutchan, F., "Identification of breakpoints in intergenotypic recombinants of HIV-1 by bootscanning", AIDS Research on Human Retroviruses, Vol. 11, pp. 1423-1435, 1996.
- [36] Grassly, N. and Holmes, E., "A likelihood method for the detection of selection and recombination using nucleotide sequences", Molecular Biology and Evolution, Vol. 14, pp. 239-247, 1997.
- [37] Brauer, M. J., Holder, M. T., Dries, L. A., Zwickl, D. J., Lewis, P. O. and Hillis, D. M., "Genetic algorithms and parallel processing in maximum-likelihood phylogeny inference", Molecular Biology and Evolution, Vol. 19, Issue 10, pp. 1717-1726, 2002.
- [38] Pond, S. L. K., Posada, D., Gravenor, M. B., Woelk, C. H. and Frost, S. D. W., "Automated phylogenetic detection of recombination using a genetic algorithm", Molecular Biology and Evolution, Vol. 23, Issue 10, pp. 1891-1901, 2006. doi:10.1093/molbev/msl051
- [39] Michael, C. C., McGraw, G. E., Schatz, M. A. and Walton, C. C., "Genetic algorithms for dynamic test data generation," In: Automated Software Engineering, Proceedings of the 12th IEEE International Conference, pp. 307-308, 1997.
- [40] Andrews, J. H., Groce, A., Weston, M., and Xu, R.-G., "Random test run length and effectiveness," In: Proceedings of the 23rd IEEE/ACM International Conference on Automated Software Engineering, pp. 19-28, 2008.
- [41] Kudjo, P. K., Ocquaye, E. N. N. and Ametepe, W., "Review of genetic algorithm and application in software testing", International Journal of Computer Application, Vol. 60, Issue 2, pp. 1-6, 2017.
- [42] Alander, J. T. and Mantere, T., "Automatic software testing by genetic algorithm optimization, a case study," In: Proceedings of the 1st International Workshop on Soft Computing Applied to Software Engineering, pp. 1-9, 1999.
- [43] Tripathy, S. P. and Kanhar, D., "Optimization of software testing for discrete test suite using genetic algorithm and sampling technique," International Journal of Computer Applications, Vol. 63, 2013.
- [44] McCart, J., Berndt, D., and Watkins, A., "Using genetic algorithms for software testing: Performance improvement techniques," Proceedings of AMCIS, pp. 222, 2007.
- [45] Wappler, S. and Lammermann, F., "Using evolutionary algorithms for the unit testing of object-oriented software," In: Proceedings of the 7th annual conference on Genetic and Evolutionary Computation, pp. 1053-1060, 2005.
- [46] Rauf, A., Anwar, S., Jaffer, M. A. and Shahid, A. A., "Automated GUI test coverage analysis using GA," In: Information Technology: New Generations (ITNG), 2010, Seventh International Conference, pp. 1057-1062, 2010.
- [47] Vaishnav, P., Choudhary, N. and Jain, K., "Traveling Salesman Problem using genetic algorithm: A survey", International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Vol. 2, Issue 3, pp. 105-108, 2017.
- [48] Yang, Y. et al., "The improved hybrid genetic algorithm for solving TSP based on Handel-C," 3rd International Conference on Advanced Computer Theory and Engineering, 978-1-4244-6542-2, 2010.
- [49] Giráldez, R., Aguilar-Ruiz, J. and Riquelme, J., "Knowledge-based fast evaluation for evolutionary learning", IEEE Transactions on Systems, Man and Cybernetics: Part C, Vol. 35, Issue 2, pp. 254-261, 2005.
- [50] Guan, S. and Zhu, F., "An incremental approach to genetic-algorithms-based classification, IEEE Transactions on Systems, Man and Cybernetics: Part B, Vol. 35, Issue 2, pp. 227-239, 2005.
- [51] Marcelin, J.-L., "Integrated optimization of mechanisms with genetic algorithms", Engineering, Vol. 2, pp. 438-444, 2010. doi:10.4236/eng.2010.26057
- [52] Srivastava, S. and Yagyasen, D., "Implementation of genetic algorithm for agriculture system", International Journal of New Innovations in Engineering and Technology, Vol. 5, Issue 1, pp. 82-86, 2016.
- [53] Sindhu, S. and Sindhu, D., "Role of data mining techniques in agriculture improvement", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 6, Issue 6, pp. 654-663, 2017.
- [54] Judson, R., "Genetic algorithms and their use in chemistry", In: Reviews in Computational Chemistry, Boyd, D.B. and Lipkowitz, K.B. (eds), Vol. 10, pp. 1-73, Wiley, 1997.
- [55] Kasat, R. B., Kunzru, D., Saraf, D. N. and Gupta, S. K., "Multi-objective optimization of industrial FCC units using Elitist using non dominated sorting genetic algorithm", Industrial and Engineering Chemistry Research, Vol. 41, pp. 4765-4776, 2002,
- [56] Palit, S., "The future vision of the application of genetic algorithm in designing a fluidized catalytic cracking unit and chemical engineering systems - a far-reaching review", International Journal of ChemTech Research, Vol.7, Issue.4, pp 1665-1674, 2014-2015.
- [57] Lameijer, E. W., Bäck, T., Kok, J. N. and Ijzerman, A. D. P., "Evolutionary algorithms in drug design", Natural Computing, Vol. 4, Issue 3, pp. 177-243, 2005.
- [58] Douguet, D., Thoreau, E. and Grassly, G., "A genetic algorithm for the automated generation of small organic molecules: drug design using an evolutionary algorithm", Journal of Computer-Aided Molecular Design, Vol. 14, pp. 449-466, 2000.
- [59] Terfloth, L. and Gasteiger, J., "Neural networks and genetic algorithms in drug design", DDT Vol. 6, no. 15 (Suppl.), 2001.
- [60] Ghaheri, A., Shoar, S., Naderan, M. and Hoseini, S.S., "The applications of genetic algorithms in medicine". Oman Medical Journal, Vol. 30, Issue 6, pp. 406-416, 2015. doi: 10.5001/omj.2015.82.