

On the Applicability of Genetic Algorithms in Subset Sum Problem

Apeksha Oberoi
Student
Echelon Institute of Technology

Jyoti Gupta
Student
Echelon Institute of Technology

ABSTRACT

The subset sum problem is one of the most important NP complete problems. Since the problem is not a deterministic one, an artificial intelligence search techniques can help to find the answers. The problem has been solved using a novel method that uses Genetic Algorithms. Genetic algorithms are generally used for optimization problems. The work analyses the effect of changing the parameters of the algorithm on the results. The results are encouraging. This work is a continuation of one of our previous works. In order to find the gaps in the literature a review was conducted. The technique has been verified and validated using standard benchmark.

Keywords

Subset sum, NP problems, complexity, Genetic Algorithms

1. INTRODUCTION

Solving hard problems has always been elusive for the computing fraternity. Problems as such can be divided into two types P and NP. P type problems are solvable in polynomial time [1]. NP problems, on the other hand can be solved by non-deterministic machines in polynomial time [1].

Formally P type problems can be defined as tractable, i.e., can be solved in polynomial time [2]. The NP problems are intractable, i.e. there is no polynomial time algorithm [2]. NP problems can be further divided into NP hard and NP complete. NP hard problems can be defined as the problems which have no polynomial time algorithm to solve the problem and the result cannot be verified in polynomial time [3]. The NP complete problems are the problems which do not have polynomial time solution but the result can be verified in polynomial time.

Subset sum problem is a NP complete problem. The problem can be defined as follows: Given a set S, and a number k, the problem is to find out the subset of the given set having sum of its elements equal to k [4]. The problem finds its applications in diverse areas including Bioinformatics, Networking etc.

The problem has been solved by many researchers. A comprehensive literature review has been carried out to find the state of the art. The paper proposes a novel approach to solve the problem. The approach uses a heuristic search technique called genetic algorithms. Genetic algorithms have been successfully applied to the problem and the results are encouraging. The results have been reported and analyzed. The paper has been organized as follows. The second section presents the literature review. The third section presents a brief overview of the background. The fourth section presents the proposed work. The next section presents the results and discussions. The last section concludes.

2. LITRATURE REVIEW

Subset sum problem has been harnessed by many researches by using various soft computing and hard computing technique. Before proposing and implementing the algorithm a literature review was conducted to find the gaps in existing work and to propose new methodology. The review was done in accordance with the guidelines proposed by Kitchenham [5]. The papers were selected from reputed journals and conferences. Along with this some of similar works regarding the topic were also included. The keywords used in search were as follows:

1. Subset sum
2. Genetic algorithm
3. NP problems

Some of the findings are summarized in table (Table 1). The table has three fields S.No. , Proposed Work & Reference.

Table 1: Literature Review

S.No.	Proposed work	References
1	Randomness was ascertained by computing autocorrelations for data values at varying time lags	[9]
2	generation of test cases via orthogonal testing	[10]
3	explores finding out the roots of an algebraic equation with the help of GA	[11]
4	It the process of natural selection in robust and efficient manner.	[12]
5	Works on binary decision diagram embedded within an effective traditional heuristic	[13]

3. BACKGROUND

3.1 NP Problems

Computational problems can be divided into two parts. The first category contains that problem which can be solved in polynomial time such problems are referred as P-Problem [1]. Shortest path problem, linear search and sorting are example of such problem. For these problems there exists a deterministic Turing machine that terminates in deterministic time.

Second class of problems is those which can be solved in polynomial time by Non deterministic machine such problem referred to as NP Problem.

NP Problems are further segregated into:

- a. NP Hard
- b. NP Complete

NP Complete are those problem that cannot be solved in finite time by deterministic machine but their solution can be verified in polynomial time. These problems come under the category of verifiable problems. K- Clique is example of such problem.

There are certain problems which can neither be solved in polynomial time nor their solution can be verified in polynomial time are known as NP Hard. Travelling Salesman is example of such problem.

3.2 Genetic Algorithm

Genetic Algorithms are heuristic search process based on the concept of survival of the fittest [1]. They are generally used for optimization. This theory is based on concept of survival of the fittest. Though genetic algorithms do not always produce best results but the average results are good enough. Genetic Algorithms have successfully been used in diverse fields like testing, NP problems, study of migration and so on.[2]

The process starts with the creation of a population of chromosomes having cells. These cells can be binary, hexadecimal or of some other form depending upon the application [1]. For example 10101 is a chromosome having cells 1, 0, 1, 0 and 1 respectively.

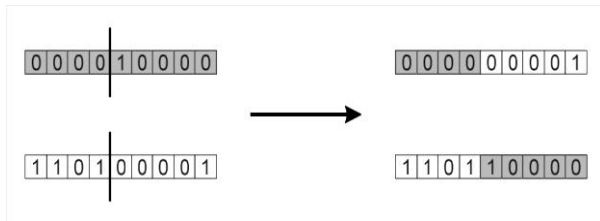


Figure 1: One point cross over

4. PROPOSED WORK

This paper proposes a novel method of determining the subset of a given set which has a sum equal to the given sum. The procedure works as follows (Algorithm 1).

Algorithm 1:

Given: Set S, sum m

Generate an initial binary population of n chromosomes.

Label: no_generation = 1;

For each chromosome

sum_chromo = 0;

For each cell

if (cell ==1)

sum_chromo+=S[i];

if (sum_chromo == sum)

return

else

fitness[i] = (1/((sum_chromo – sum))*100;

Apply Rowlett Wheel Selection;

Apply crossover;

These chromosomes are then subjected to various operators. The operators are selection, crossover, and mutation.

Crossover is a process which creates new chromosomes having properties of both parents. There are many types of crossover some of them are as follows.

1. One point crossover
2. Two point crossover
3. Uniform crossover
4. Multi point crossover

In a one point crossover, a new chromosome is formed by taking left part of first chromosome and right part of second chromosome (Figure 1).

Likewise, there is a multipoint crossover, in which the formation of new chromosomes is governed by selecting multiple random numbers and crossing over string along them.

Mutation operation breaks the local maxima. Mutation can be carried out by flipping a bit from a selected chromosome.

The chromosomes are selected using any type of selection like Rowlett Wheel Selection. The selection uses fitness of a chromosome crafted during the design process.

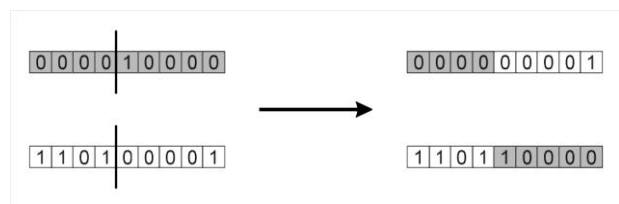


Figure 2: Two point cross over

Apply mutation;

Goto Label;

The above algorithm works as follows. For each chromosome, the sum of the subset formed is calculated. If the sum is equal to the desired sum, the procedure returns, else it continues. The procedure stops if a solution is found or the number of generations exceeds a threshold number.

The procedure uses a simple formula for the fitness of a chromosome. The selection method used in the procedure is Rowlett Wheel selection.

5. RESULTS

The work has been implemented and analyzed. The implementation has been done in C++. The procedure is tested against the benchmarks for n= 5, 10 , 15, 20 and 25.

The results have been depicted in Table 1 and graphs (Figure 1). Table 1 depicts the results of the 5 runs (in each case) of the sets for which the solution was obtained. Table 2, on the other hand, depicts the results of the fitness at the end of the 5th generation for the sets for which the solution cannot be obtained. Figure 1 graphically depicts the results of the second table.

Table 2: results of the 5 runs of the sets for which the solution was obtained

n	Result obtained in generation
5	4
5	3
5	6
5	5
5	4
10	6
10	7
10	6
10	8
10	6
15	9
15	13
15	12
15	13
15	8

Table 3: Results of the fitness at the end of the 5th generation

n	Result obtained in generation
5	87
5	89
5	82
5	91
5	82
10	79
10	81
10	88
10	78
10	82
15	72
15	71
15	70
15	69
15	73

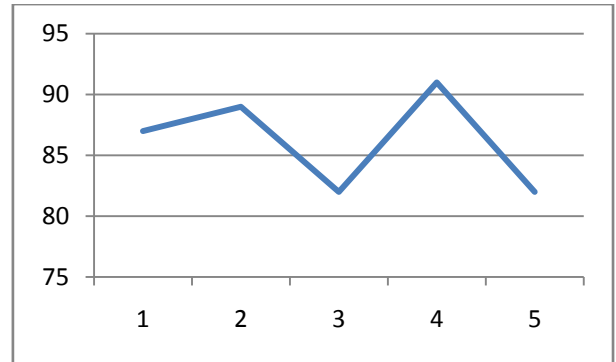


Figure 3: Results of the fitness at the end of the 5th generation

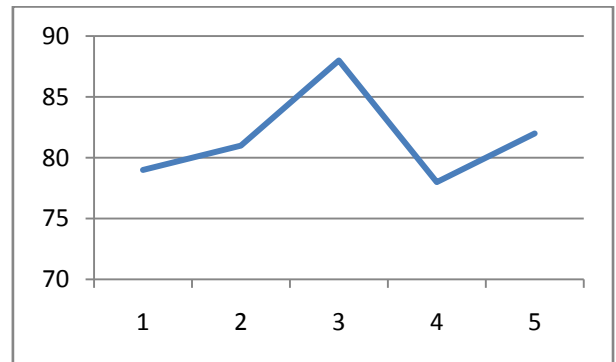


Figure 4: Result obtained in generation

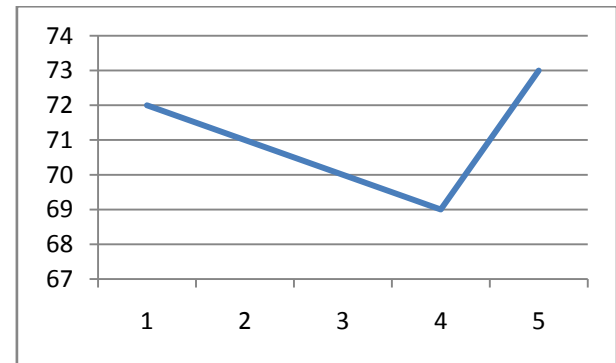


Figure 5: Result obtained in generation

6. CONCLUSION AND FUTURE SCOPE

The paper implements the proposed algorithm for subset sum problem. The algorithm has been implemented and verified. The results are encouraging.

The problem is now being tested for $n = 50$ and $n = 100$. For this reason, the sets are being crafted. The sets are being designed keeping in view constraints of Genetic Algorithms and that the sets form a good sample. The next step would be to implement the problem using Diploid Genetic Algorithms. The implementation for Diploid has been done and an extensive literature review has also been carried out [7]. The Diploid Genetic Algorithms are known for their robustness.

7. ACKNOWLEDGMENTS

Our thanks to the experts who have contributed towards development of the proposed work.

8. REFERENCES

- [1] T.H. Cormen, C.E. Leiserson, and R.L. Rivest, "Introduction to Algorithms", the MIT Press, Cambridge, Massachusetts, USA, 2009
- [2] Bhasin,H. "Algorithms" , Oxford University Press, 2015
- [3] Sahni, S "Fundamentals of Computer Algorithms" Computer Science Press 1803 Research Blvd. Rockville, Maryland 20850
- [4] Gerhard J. Woeginger" Exact Algorithms for NP-Hard Problems: A Survey" Combinatorial Optimization (Edmonds Festschrift), LNCS 2570, pp. 185–207,c Springer-Verlag Berlin Heidelberg 2003
- [5] David E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, first ed., Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA 198.
- [6] Bhasin, H.; Behal, G.; Aggarwal, N.; Saini, R.K.; Choudhary, S., "On the Applicability of Diploid Genetic Algorithms in Dynamic Environments," in Soft Computing and Machine Intelligence (ISCMI), 2014 International Conference on , vol., no., pp.94-97, 26-27 Sept. 2014
- [7] Bhasin,H, Behal,G , Aggarwal,N, Saini,R, Choudhary.S "On the applicability of diploid genetic algorithms in dynamic environments" Springer Berlin Heidelberg 2015
- [8] Bhasin, H.; Mehta. S."On the applicability of diploid genetic algorithms" Springer Berlin Heidelberg, 28 August 2015.
- [9] Goyat,S, "Cryptography Using Genetic Algorithms (GAs)" IOSR Journal of Computer Engineering (IOSRJCE) ISSN: 2278-0661 Volume 1, Issue 5 (May-June 2012), PP 06-08
- [10] Bhasin,H, "Use of Genetic Algorithms for Finding Roots of Algebraic Equations", IJCSIT, Vol. 2,Issue 4.
- [11] Richards,R "Classifier System and Genetic Algorithm" 0004-3702/89/\$3.50 © 1989, Elsevier Science Publishers B.V. (North-Holland)
- [12] Bhasin,H and Gitanjali , "Harnessing Genetic Algorithm for Vertex Cover Problem", International Journal on Computer Science and Engineering (IJCSE), Vol. 1, Issue 2, pp. 218 – 223
- [13] 1998 Isaac K. Evans. To appear in Evolutionary Programming VII, "Proceedings Seventh International Conference (EP98)"