International Journal of Computer Applications (0975 – 8887)
Volume 123 – No.17, August 2015

The Applicability of Genetic algorithm to Vertex Cover

Harsh Bhasin
Assistant Professor
Dept. of Computer Science
Jamia Hamdard

Mohammad Amini
Student
Dept of Computer Science
Jamia Hamdard

ABSTRACT

The Vertex Cover Problem calls for the selection of a set of vertices(V) in a way that all the edges of the graph, connected to those vertices constitute the set E of the given graph G= (V, E). The problem finds applications in various fields and is therefore, one of the most widely researched topics in NP Complete Problems. The problem is an NP Complete problem this work proposes a Genetic Algorithm based solution to handle the problem. The proposed algorithm has been implemented and tested for various graphs. These instances vary in the number of vertices and connectivity. The results are encouraging. This paper also explores the available techniques in order to put the things in the perspective. The future scope of this work intends to apply Diploid Genetic Algorithms to the problem to incorporate robustness into the proposed algorithm.

Keywords

Keywords are Vertex Cover Problem, NP Completeness, Genetic Algorithm, Artificial Intelligence.

1. INTRODUCTION

The Vertex Cover Problem (VCP) is one of the most important NP Complete problems. The problem has been extensively dealt-with by various researchers. This paper explores the techniques that have been successfully applied to VCP. In order to do so, an extensive literature review has been carried out. The paper also proposes a Genetic Algorithm based method to handle the problem. The work is an extension of one of our previous works [1]. This technique has been verified by taking 15 instances. The results have been presented in the paper.

The Vertex Cover Problem (VCP) can be defined as follows. Given a Graph G= (V, E); it is needed to find $C \subseteq V$ such that every edge $e \in E$ has an endpoint in C [2]. The problem has been extensively dealt-with by various researchers. VCP has various applications in fields such as mathematics, civil engineering, Management, biology and chemistry.

The paper has been organized as follows. Section 2 of the paper presents the literature review. Section 3 gives an overview of various NP complete problems. Section 4 presents the proposed work. Section 5 discusses the results and finally section 6 concludes. The work assumes importance because of the applicability of VCP in various fields, as stated earlier. Also, the future scope of the work would apply Diploid Genetic Algorithm to VCP, as an extension to one of our works.

2. LITERATURE REVIEW

An extensive literature review has been carried out to explore various techniques. The following discussion presents various techniques and discusses the advantages and disadvantages of the techniques. The following discussion finds the gaps in the existing techniques and puts them in perspective.

Various evolutionary algorithms for the (VCP) have been proposed by numerous researchers like Evans [3]. The experiments carried out showed that these algorithms perform better in the case of local search. The applicability of evolutionary algorithms for VCP has been verified and validated [4]. Oliveto et. Al. compared evolutionary algorithms to approximation algorithms [5]. In the work the theoretical proofs of time complexity and the Vertex Cover size have been explored and enhanced. These proofs form the theoretical foundations of the applicability of evolutionary algorithms in VCP.

In order to get an idea of the applicability of other soft computing techniques in Vertex Cover an Ant Colony Optimization (ACO) algorithm was studied [6]. The ACO algorithm incorporates features in order to find the vertex set where the total weight can be minimized. It may be stated that the solution of this problem does not necessarily constitute a tree which makes this problem different as compared to other problems like Minimum Spanning Tree. The author has verified the algorithm for graphs with vertices ranging from (50-300).

The Vertex Cover Problem as an immensely important problem and the Minimum Vertex Cover Problem is an NP-Complete version of it. The problem has also been applied to DNA computation via self-assembly. The algorithm proposed by the authors required Θ (n*m) types of tiles, n is the number of vertices and m being the number of edges [7].

A multi -start iterated Tabu Search (TS) algorithm has also been used to handle the minimum weight VCP. In the work the public benchmark instances have been used to prove the validity of the work. The work has been verified by taking graphs in which the number of vertices vary from 10-1000.

A similar work [8] used hybridized TS, and controlled simulated annealing [9]; in order to tackle the problem. The experiments carried out proved veracity of the work.

The problem has also been solved earlier by phased local search, which work by either randomly selecting a vertex or on the basis of degree of a vertex or dynamic adjustments. This technique has been verified using benchmark programs [10].

Heuristics based study states genetic algorithm has been compared with three standard algorithms and verified using DIMACS benchmark [11]. The problem has also been solved by fixed parameter evolutionary algorithm wherein the running time $\Theta(f(OPT)n^c)$, where c is a small constant [12].

The problem has also been solved using genetic algorithm and compared with 2-approximation algorithm [13]. A brief overview of the techniques for solving the VCP has been presented in Table 1.

Table 1. Review of existing techniques

SR NO	NAME OF AUTHORS	TECHNIQUE	VERIFICATION	INCREMENTS
1	F.Wu et. Al	The paper proposes a new DNA algorithm based on the DNA tile self-assembly model.	The verification of the proposed technique was done by mathematical proof and simulation experiments using Xgrow simulator [7].	The new algorithm can significantly improve the accuracy of the results up to $\Theta(n \times m)$. n=vertices, m=edges
2	Isaac K. Evans	The approach relies on a binary decision diagram embedded within an effective traditional heuristic.	The encoding was empirically examined on a variety of vertex cover problem classes, including graphs derived from ISCAS-89 circuit benchmarks and graphs included in the DIMACS benchmark suite for maximum clique[3].	The approach also allows initial EA populations to be seeded in known promising regions of the search space.
3	Marija Milanovic´	The binary representation, the mutation with frozen genes, limited number of different individuals with the same objective value and the caching technique were used.	The experiments were carried out on randomly generated instances with up to 500 vertices and 100 000 edges [13].	The genetic algorithm outperformed both CPLEX solver and 2-approximation heuristic
4	M.Safar et. Al	The paper encoded the vertex-cover problem into the evolutionary domain, where the objective function is to select a minimal set of sensors out of the coverage sensors to act as the vertex-cover set	The experiments were carried out using Sensor CAD Visualizer.[15]	As with the coverage model, our communication modeling has reduced the solution space into a discrete optimization problem so that it can achieve the maximum communication possible with the least number of the coverage sensors (i.e., vertex covers)
5	A.Singh et. Al	In this paper a heuristic based steady-state genetic algorithm is presented for the maximum clique problem.	The paper is verified by extensive computational results based on the algorithm[16].	The used algorithm is compared with three best evolutionary approaches and the overall best approach, which is non-evolutionary, for the maximum clique problem and finds that our algorithm outperforms all the three evolutionary approaches in terms of best and average clique sizes found on majority of DIMACS benchmark instances.
6	S.Kratsch et. Al	The notion of fixed-parameter evolutionary algorithms was introduced to examine how the runtime of search heuristics depend on structural properties of a given problem.	Using this approach it was examined the runtime and approximation behavior of evolutionary algorithms with respect to the value of an optimal solution [12].	It is shown that evolutionary algorithms solve the vertex cover problem efficiently if the size of a minimum vertex cover is not too large, i.e., the expected runtime is bounded by O(f (OPT)· n^c).
7	T.Zhou et. Al	An MS-ITS is being proposed for solving the MWVCP. We apply the TS to find the local optimum where first a problem specific neighborhood with the FES is proposed to reduce the computational time.	The MS-ITS algorithm was programmed in C and executed it on a personal computer with AMD A6-3400M APU with 1.40 GHz, while RGES was tested in PIV with 3.2 GHz processor running under Windows XP, and PBIG was executed on a cluster of	Experimental results demonstrate the high effectiveness of our proposed MT-ITS algorithm compared with other state-of-the-art algorithms

			PCs equipped with Intel X3350, 2667 MHz processors, and ACO was performed on an Intel CoreTM(2)Duo, 4.00 GHZ CPU. Obviously, the machines used by RGES and PBIG are both about twice faster than ours, and the machine used by ACO is more than three times faster than ours[9].	
8	P.S.Oliveto et. Al	The (1+1)-EA finds the optimal cover of each instance of the considered graph class in polynomial time.	Theoretical proofs of how and why the (1+1)-EA performs better than VERCOV on the previously examined instance classes have been presented [5].	Given polynomial time the (1+1)-EA outperforms VERCOV in solution quality even when it is not able to find the optimal cover on the given graphs
9	S.Khuri et. Al	The study then focuses on the genetic-based heuristic.	Several problem instances are used with both algorithms and the results are compared [17].	The results found by the genetic algorithm are better than those obtained from the best known traditional heuristic, the vercov algorithm.
10	D. P. Chandu	This paper presents a parallel genetic algorithm for generalised vertex cover problem(GVCP) using Hadoop Map-Reduce framework	In this implementation fitness computation operations, crossover operations, and mutation operations are distributed among all the machines in Hadoop cluster running reduce phase[18].	The proposed Map-Reduce implementation helps to run the genetic algorithm for generalized vertex cover problem(GVCP) on multiple machines parallely and computes the solution in relatively short time

3. NP PROBLEMS

The problems that use deterministic algorithms generally take polynomial time. The complexity of these problems is O(n). According to the theory of Automata the corresponding Turing machine takes polynomial number of steps in order to accomplish, depicted by the above algorithms. On the other hand, if there is no deterministic algorithms to solve a problem but a problem can be solved by nondeterministic algorithm in polynomial time the problem is referred to as nondeterministic polynomial (NP) problem.

These problems are of two types: NP-Complete and NP-Hard. The NP-Complete problems cannot be solved in polynomial time but there is an algorithm to check whether the answer is correct or not in polynomial time. These problems are generally decision problems.

On the other hand there are some problems which can neither be solved in polynomial time nor there is an algorithm whether solutions satisfies in polynomial time; such problems are referred to as NP-Hard Problems. These problems are generally optimization problems. Various NP Problems have been presented in Table 2.

4. PROPOSED WORK

The previous sections discussed the conventional approaches to solve the VCP. This section proposes a novel Genetic based algorithm to solve the problem. The method is as follows.

First of all, a population is created. The population is binary. ‘1’in a cell represents the inclusion of the vertex and a ‘0’ represents the exclusion. The population is checked for the given graph. If it forms the vertex cover of the given graph, then it is transferred to another data structure called “fit_population”.

‘This population then undergoes crossover. The number of crossovers is given by the following formula. In the work the crossover rate is between 2-5%. One point crossover has been implemented.

The third step mutates the fit population. The number of mutations is given by the following formula.

The mutation rate has to be low. In the work, the mutation rate is in between 0.5 and 1%. In the implementation a random cell of a random chromosome is selected and flipped. The fitness of a chromosome is given by the following formula.

The Section used in the work is Roulette Wheel. The pseudo code of the procedure is as follows.

Pseudo-Code
The generalized Genetic Algorithm:

```
1-Generate the following for 20 times and each time store the chromosomes which are a vertex cover in the graph and name it fit_population[i]
        a. Randomly generate a population
        b. Perform a single point crossover on population
        c. Perform mutation on population
```

2-Find the fitness value of each chromosome in **fit_population**

3-Sort non-increasingly based on fitness value

4-Choose a new population using the cumulative frequency of the population (perform a roulette wheel selection on our fit_population)

5- Repeat the process till the solution is found or the number of generations exceeds a threshold value.

Table 2. NP Problems

SR NO	Problem	Definition
1	Traveling Salesman Problem	The traveling salesman problem consists of a salesman and a set of cities. The salesman has to visit each one of the cities starting from a certain one (e.g. the hometown) and returning to the same city. The challenge of the problem is that the traveling salesman wants to minimize the total length of the trip [2].
2	Clique problem	The clique problem refers to any of the problems related to finding particular complete sub graphs ("cliques") in a graph, i.e., sets of elements where each pair of elements is connected [2].
3	Subset Sum Problem	Input: set of n positive integers, {w0… wn-1}, maximum weight W • Output: a subset S of the input set such that the sum of the elements of S ≤ W and there is no subset of the input set whose sum is greater than the sum of S and ≤ W [19].
4.	SAT3 Problem	Determining the satisfiability of a formula in conjunctive normal form where each clause is limited to at most three literals[2]
5	Minimum Vertex Cover	In the mathematical discipline of graph theory, a vertex cover (sometimes node cover) of a graph is a set of vertices such that each edge of the graph is incident to at least one vertex of the set [2].
6	Hamiltonian Circuit	A cycle in a directed or undirected graph that visits each vertex exactly once [2].
7	Chromatic Number	Assignment of colors to vertices of a graph such that no two adjacent vertices share the same color. The minimum number of colors is called chromatic number [19].
8	Graph coloring	A proper coloring of a graph is an assignment of colors to the vertices of the graph so that no two adjacent vertices have the same color [19].
9	Partition	Deciding whether a given multi set S of positive integers can be partitioned into two subsets S_1 and S_2 such that the sum of the numbers in S_1 equals the sum of the numbers in S2 [19].
10	Knapsack	Given a set of items, each with a mass and a value, determine the number of each item to include in a collection so that the total weight is less than or equal to a given limit and the total value is as large as possible [19].

5. RESULTS

The work has been implemented in C++. The work was verified by taking graphs having 5, 7, 10, 20 and 30 nodes. 3 instances of each were taken. The instances were crafted using DIMACS. The instances had all types of graphs: complete, almost complete and those having fewer edges. The results are encouraging. The results are shown in Table 3**.** It may be stated that the empty rows represent no results.

6. CONCLUSION

Vertex cover problem is one of the most interesting and important problems. Since it is an NP Complete problem, therefore its solution is not possible via conventional techniques. The present work uses Genetic Algorithms to handle the problem. The algorithm has been implemented and tested. The algorithm is an extension of the work done before and is able to handle the graphs which could not be handled earlier [13]. The results presented in the previous section, show that the technique works well for most of the instances. However, there are some cases where the method does not work. The cases wherein the graph is complete still require a better method. The future extension of this work would apply Diploid Genetic Algorithms to such instances. An extensive literature review has been carried out to understand the concept [20].Diploid Genetic Algorithms have already been applied to Travelling Salesman Problem [21, 22].

Table 3. Results

Number of nodes (no of the instance)	**Crossover Rate, Mutation Rate**							
	2 ,0.5	2, 1	3, 0.5	3, 1	4, 0.5	4, 1	5, 0.5	5, 1
5(1)	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111
10 chrom	10	10	10	10	10	10	10	10
5(2)	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111
10 chrom	10	10	10	10	10	10	10	10
5(3)	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111	01101, 01111
10 chrom	10	10	10	10	10	10	10	10
7(1)	0110111, 0111011	0110111, 0111011	0110111, 0111011	0111011, 0110111	0110111, 0111011	0110111, 0111011	0110111, 0111011	0111011, 0110111
21 chrom	21	21	21	21	21	21	21	21
7(2)	0110111, 0111011	0110111, 0111011	0110111, 0111011	0111011, 0110111	0110111, 0111011	0110111, 0111011	0110111, 0111011	0110111, 0111011
21 chrom	21	21	21	21	21	21	21	21
7(3)	0110111, 0111011	0110111, 0111011	0110111, 0111011	0110111, 0111011	0110111, 0111011	0110111, 0111011	0110111, 0111011	0110111, 0111011
21 chrom	21	21	21	21	21	21	21	21
10(1)	0111111110	0111111101	0111011111	0111101111	0111011111	0111011111	1111111011	0011111111
30 chrom	30	40	40	50	50	60	80	90
10(2)	0111111110	0111101111	0111011111	0111101111	0111011111	0111011111	1111111011	0011111111
30 chrom	30	30	40	50	50	60	80	90
10(3)	0111111110	0111101111	0111011111	0111101111	0111011111	0111011111	1111111011	0011111111
30 chrom	30	30	40	50	50	60	80	90
20(3)	01101111111001100011	01101101111001101111	00111111111010110111	01101010101101101111	11111110101000000111	11111111101000111101	11111111101000111101	11111111101000111101
100 chrom	100	100	110	120	130	130	130	130
20(2)	00111111111010110111	01111101111011001110						
110 chrom	110	135						
20(1)	0111110111101100111							
140 chrom	140							
30(1)								
300 chrom								
30(2)	111111101010000001110011111111							
250 chrome	250							
30(3)								
350 chrom								

7. REFERENCES

[1] Bhasin, H. et Al., 2012, Harnessing Genetic Algorithm for Vertex Cover Problem, International Journal on Computer Science and Engineering (IJCSE), ISSN : 0975-3397, Vol. 4 No. 02.

[2] Corman H. Thomas, Leiserson E. Charles, Rivest L. Ronald, Stein Clifford "Introduction to Algorithms" Second Edition McGrawHill Book Compan(EP98).

[3] Evans, I.K. (1998) Evolutionary algorithms for vertex Cover, ©1998. To appear in Evolutionary Programming VII, Proceedings Seventh International Conference (EP98).

[4] Friedrich, T., et Al,(2008) Analyses of Simple Hybrid Algorithms for the Vertex Cover Problem, by the Massachusetts Institute of Technology, Evolutionary Computation

[5] Oliveto, P.S., He, J. and Yao, X. (2007) Evolutionary algorithms and the vertex cover problem 1-4244-1340-0/07/ IEEE

[6] S.J, et Al.(2004) An Ant Colony Optimization Algorithm for the Minimum Weight Vertex Cover Problem, Annals of Operations Research 131, 283–304.

[7] Wu, F., Li, K., Sallam, A. and Zhou, X. (2013) A molecular solution for minimum vertex cover problem in tile assembly model, Published online: 16 March 2013 © Springer Science+Business Media New York 2013, J Supercomput (2013) 66:148–169 DOI 10.1007/s11227-013-0892-0

[8] Voß, S. and Fink, A. , (2012) , A hybridized tabu search approach for the minimum weight vertex cover problem, J Heuristics 18:869–876 DOI 10.1007/s10732-012-9211-9

[9] Zhou, T., Lu, Z., Wang, Y., Ding, J. and Peng, B. (2015) Multi-start iterated tabu search for the minimum weight vertex cover problem, Springer Science+Business Media New York 2015, J Comb Optim DOI 10.1007/s10878-015-9909-3

[10] S. Balaji, , (2013) A New Effective Local Search Heuristic for the Maximum clique problem , World Academy of Science, Engineering and Technology International Journal of Mathematical, Computational, Statistical, Natural and Physical Engineering Vol:7, No:5.

[11] Pullan, W.,(2007) Approximating the maximum vertex/edge weighted clique using local search, Published online: 4 May 2007 Springer Science+Business Media.

[12] Kratsch, K. and Neumann, F. (2012) Fixed-parameter evolutionary algorithms and the vertex cover problem Springerlink.com, Algorithmica (2013) 65:754–771 DOI 10.1007/s00453-012-9660-4

[13] Milanovic, M. (2010) Solving the generalized vertex cover problem by genetic algorithm, Computing and Informatics, Vol. 29, 1251–1265

[14] Bhasin, H., On the applicability of Diploid Genetic Algorithms in Dynamic Environments,(2014), IEEE, 2014 Intl. Conference on Soft Computing and Machine Intelligence

[15] Safar, M. and Habib, S. (2007) Hard constrained vertex cover communication algorithm for WSN , T.-W. Kuo et al. (Eds.): EUC 2007, LNCS 4808, pp. 635–649, IFIP International Federation for Information Processing

[16] Singh, A. and Gupta, A.K. (2006) A hybrid heuristic for the maximum clique problem, Springer Science + Business Media,, J Heuristics (2006) 12: 5–22 DOI 10.1007/s10732-006-3750-x

[17] Khuri, S. and Back, T. (1994) An evolutionary heuristic for the minimum vertex cover problem http://www.researchgate.net/publication/2581134

[18] Chandu, D.P. (20014) A parallel genetic algorithm for generalized vertex cover problem (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (6) , 7686-7689

[19] Karp, R.M., (1972). "Reducibility Among Combinatorial Problems" (PDF). In R. E. Miller and J. W. Thatcher (editors). Complexity of Computer Computations. New York: Plenum. pp. 85–103.

[20] Bhasin, H. et. al, On the Applicability of Diploid Genetic Algorithms, AI and Society, Springer (In Press).

[21] Bhasin, H. et. al., On The Applicability of Diploid Genetic Algorithms in Dynamic Environments, Soft Computing, Springer. DOI: 10.1007/s00500-015-1803-5.

[22] Bhasin, H., et. al., On the Applicability of Diploid Genetic Algorithms, AI and Society, Springer, DOI: 10.1007/s00146-015-0591.