# Approximate Maximum Clique Algorithm (AMCA): A Clever Technique for Solving the Maximum Clique Problem through Near Optimal Algorithm for Minimum Vertex Cover Problem

Muhammad Fayaz<sup>1</sup>, Shakeel Arshad<sup>2</sup>, Abdul Salam Shah<sup>3\*</sup> and Asadullah Shah<sup>4</sup>

<sup>1,2</sup> Department of CS and IT, University of Malakand KPK, Pakistan
 <sup>3</sup> University of Kuala Lumpur (UniKl-MIIT), Kuala Lumpur, Malaysia
 <sup>4</sup> International Islamic University Malaysia (IIUM), Kuala Lumpur, Malaysia
 <sup>1</sup>hamaz\_khan@yahoo.com, <sup>2</sup>shakeelswati@gmail.com,
 <sup>3\*</sup>shahsalamss@gmail.com, <sup>4</sup>asadullah@iium.edu.my

#### Abstract

**Background and Objective:** The process of solving the Maximum Clique (MC) problem through approximation algorithms is harder, however, the Minimum Vertex Cover (MVC) problem can easily be solved using approximation algorithms. In this paper, a technique has been proposed to use the approximation algorithms of Minimum Vertex Cover (MVC) for the solution of the Maximum Clique (MC) problem.

Materials and Methods: To test the proposed technique, selected approximation algorithms have been deployed to small graph instances. The algorithms that were used for experiments are Maximum Degree Greedy (MDG), Vertex Support Algorithm (VSA), Mean of Neighbors of Minimum Degree Algorithm (MNMA), Modified Vertex Support Algorithm (MVSA), Maximum Adjacent Minimum Degree Algorithm (MAMA), and Clever Steady Strategy Algorithm (CSSA).

**Results:** The development of an efficient approximation algorithm for the Maximum Clique (MC) problem is very difficult due to its complex nature. The only way left is to use the approximation algorithm of Minimum Vertex Cover (MVC) for the solution of the Maximum Clique (MC) problem. The experimental analysis of the proposed algorithm has revealed that the Maximum Clique (MC) problem can be efficiently solved with approximation algorithms of Minimum Vertex Cover (MVC). The proposed algorithm has efficiently solved the Maximum Clique (MC) problem within the reduced time limit.

Conclusions: It is a difficult task to directly solve the Maximum Clique (MC) problem through approximation algorithms. The proposed method provides a platform to efficiently solve the Maximum Clique (MC) problem by using the approximation algorithm of Minimum Vertex Cover (MVC).

**Keywords:** Graph Instances, Minimum Vertex Cover, Maximum Independent Set (MIS), Maximum Clique Problem, NP-Complete Problem.

**Significant Statement:** Maximum Clique (MC) is one of the earlier NP problems which were proved as NP-complete. The polynomial algorithm has not yet been discovered to solve the NP-complete problem in polynomial time.

The Maximum Clique (MC) problem is correlated to some well-known NP-hard problems, such as Graph Coloring, Maximum Independent Set (MIS), SAT problems, and the Minimum Vertex Cover (MVC) problem [1]. The MVC problem can be tackled by

ISSN: 2005-4297 IJCA Copyright © 2018 SERSC Australia

-

Received (October 15, 2017), Review Result (December 13, 2017), Accepted (January 15, 2018) \* Corresponding Author

approximation algorithms and exact algorithms, while Branch and Bound is an exact technique which is the best choice to solve the MC problem. It has been noticed that the simple approximation algorithms always provide the poor result to solve the MC problem, but provide even better results for the MVC problem. Several approximation algorithms which are based on simple heuristic are designed to solve the MVC problem and several exact algorithms based on Branch and Bound technique are proposed to solve MC problems. The main objective of this study is to use the approximation algorithms of Minimum Vertex Cover (MVC) for solving the MC problem.

#### 1. Introduction

The undirected graph is the group of nodes and edges, where if the two nodes are connected to each other are said to be adjacent vertices. A set of entire adjacent vertices in a graph is a clique, where MC is the clique having more nodes as compared to the other cliques. A maximum clique is the optimization problem in which one desires to discover Maximum Clique (MC) of a graph [2].

The maximum clique problem plays an important role in theoretic and applied areas. Many of the algorithms incorporate maximum clique as a subtask for the solution of the particular problem. MC has a lot of importance in the field of data analysis for finding similar data, like the classification of diseases based on symptom correlation [3]. Other application areas of Maximum Clique are Computer Vision and Coding Theory, Pattern Recognition, Image Processing, Protein Structure Alignment [4] Fault Diagnosis, Geometry, Bioinformatics, Drug Design, Clustering, Design of Quantum Circuits, Data Mining, Design of Wireless Networks [5], Cryptography, Cryptanalysis, Telecommunications, Radio Frequency Assignment [6], and Biochemistry [7].

Maximum Clique (MC) is among the pioneer NP problems which were proved as NP-complete [8]. The polynomial algorithm has not yet been discovered for solving the NP-complete problem in the reasonable amount of time [9]. The MC problem is in association with some other well-known NP-hard problems, such as Graph Coloring, MIS, SAT problems, and the Minimum Vertex Cover (MVC) problem. For the testing of algorithms a graph plays an important role. The technique in this paper is based on complemented graph which is an undirected graph  $\hat{G}=(V,\hat{E})$ , where  $\hat{E}=\{(vi,vj)\mid vi,vj\in V,i\neq j,(vi,vj)\not\in E\}$  [10]. A simpler graph G and its complemented graph  $\hat{G}$  is shown in Figure 1 (a) and (b).

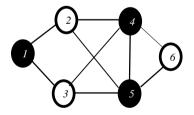
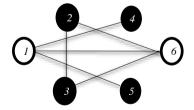
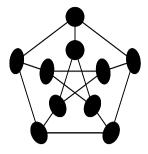


Figure. 1. (a). Original Graph,



(b). Complemented Graph

A little bit complicated graph G and its complemented graph  $\hat{G}$  is shown in Figure 2 (a) and (b).



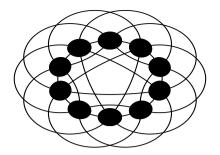


Figure. 2. (a). Original Graph,

(b). Complemented Graph

To tackle the MC problem normally approximation algorithms and exact algorithms are used. Approximate algorithms are fast, but optimality is not guaranteed, they are suitable for large problems when near optimal solution is required in a reasonable time [9]. On the other side, the exact algorithms are suitable for small problems when the optimal solution is required within no time limit. The computation grows as the problem grows. The branch and bound is the most important exact algorithm to solve the MC problem [11]. Interestingly most of the algorithms for the MC problem are complete and use some form of the branch and bound algorithm. Recently many authors have tried to solve the MC problem using the branch and bound algorithms. The approximate colouring is used to obtain an upper bound, but there can be a large difference between  $\chi$  (G) and  $\omega$  (G),  $\chi$  (G) is not a tighter bound of w (G) [12] [13]. Most of the proposed algorithms for the maximum clique problem focused on improving the upper bound for branch and bound method because the tighter outer bound reduces the unnecessary searching.

Several approximate methods have been deployed to the MC problem. Due to the nature of the MC problem, it is difficult to design a better approximate algorithm for it. In contrast, MC problem approximation algorithms are more suitable to solve the MVC problem. The main aim of this paper is to utilize the approximation algorithms of Minimum Vertex Cover (MVC) for the solution of the Maximum Clique (MC) problem with reduced effort and time.

The remaining structure of the paper is organized as: the literature review has been carried out in Section 2, the proposed method has been explained in Sections 3, Section 4 contains evaluation and results, finally, the conclusion is provided in Section 5.

### 2. Literature Review

The background study is important for the better understanding of the topic. The literature review contains a detailed discussion of some well-known approximation algorithms for the MVC problem and exact algorithm for the Maximum Clique (MC).

The NP-Complete set is comprised of numerous problems, such as Maximum Independent Set (MIS), Graph Coloring, MVC, Set Cover, and Maximum Clique. These approaches are used in numerous fields, and every problem has own domain of applications. The nature of every problem is also different from each other, therefore different methods are required to tackle them. MIS problem is a very important NP-complete problem and has been successfully used in different areas. The minimum vertex problem is suitable to be tackled by approximate methods and exact methods, such as branch and bound is the best choice to solve the maximum clique problem. It is interesting that the simple approximate method always provides very poor result for solving the maximum clique problem while providing even

better results to solve the maximum clique problem. Similarly, the branch band which is an exact technique is appropriate for solving the maximum clique problem. Different approximation algorithms based on simple heuristic are designed to solve the MVC problem and several exact algorithms based on the branch and bound technique have been proposed for solving the maximum clique problems.

The initially recognized algorithm for the MVC problem is Maximum Degree Greedy algorithm (MDG) [14]. This method is based on a greedy strategy to select the vertex having the largest degree in G. The usefulness of this strategy is that it is very fast and simple, but the lack of intelligent tweak in this algorithm leads to failure on many instances. Keeping in mind the maximum degree algorithm, another strategy has been proposed by Balaji *et al.*, known as Vertex Support Algorithm (VSA) [15]. In VSA first the support of a node is the sum of all degrees of adjacent degrees to a vertex is calculated and the vertex having maximum support is calculated. Like MDG it is also based on greedy approach, hence it also fails in many cases.

A modification has been carried in VSA and another algorithm is suggested by Imran et al, named Modified Vertex Support Algorithm (MVSA) [16], like VSA in this approach the support of each vertex is computed, and the neighbour vertex of the minimum support vertex is chosen for vertex cover. The MVSA is not based on greedy approach like MDG and VSA, and provide an even better approximate solution as compared to MDG and VSA, but more complex as compared to them. In contrast to the approaches based on MDG vertex support Fayaz *et al.*, proposed another approach, named Clever Steady Strategy Algorithm (CSSA) [17]. In CSSA, the minimum degree neighbour of the minimum degree vertex is selected for vertex cover. In case of a tie (a situation where more than 1 vertices have the same minimum degree), a neighbour of that minimum vertex is selected having the minimum degree in all neighbours of minimum degree vertices. In this algorithm, the tie solution has been broken which contribute to better performance. The proposed is as simple and fast as MDG and VSA, but provide a better near-optimal solution as compared to them, even provide the better result than MVSA.

The result of CSSA algorithm shows that the neighbour vertices to the minimum degree vertex have great importance, therefore Fayaz et. al., proposed another approach named Maximum Adjacent to Minimum Degree Algorithm (MAMA) [18]. In MAMA, again the neighbour of the minimum vertex is considered, but the maximum degree vertex of the minimum degree vertex is selected for the vertex cover. The advantages of MAMA and CSSA are same and both can be used as the alternative of each other. For some graph instances the CSSA provides the better result as compared to MAMA and in some cases, MAMA is better than CSSA. Other approximation algorithms for minimum vertex cover, such as max degree around algorithm (MDA) [19], Mean Neighbor to Minimum Degree Algorithm (MNMA) [20], Degree contribution algorithm (DCA) [21], and Advanced Vertex Support Algorithm (AVSA) [22], in the recent past for solving the minimum vertex cover problem.

Here in this study, we are interested to solve the maximum clique problem through approximation by using some tweaks to the graph. Most of the work for solving the maximum clique problem has been carried by using the branch and bound algorithms. In branch bound algorithm most efforts for providing lower bound have been done. Some attempts to solve the maximum clique problem through approximation algorithm have also been made, but fails to provide the optimal solution.

# 3. Proposed Method

The Approximate Maximum Clique Algorithm (AMCA) has been proposed to utilize the approximation algorithms of Minimum Vertex Cover (MVC) to find the maximum clique in a graph. The nature of the maximum clique problem makes it very difficult to design a better approximation algorithm for its solution. In the proposed approach we have used the approximate techniques of the Minimum Vertex Cover (MVC) problem to solve the Maximum Clique (MC) problem. First, the algorithm compliments the graph so that the Minimum Vertex Cover (MVC) algorithms can be deployed on the graph. The approximate algorithm returned the Minimum Vertex Cover (MVC) for the complemented graph. The remaining vertices of Minimum Vertex Cover (MVC) in the complemented graph will be considered as maximum clique in the original graph. The set of vertices in Minimum Vertex Cover will be removed from G, and the number of remaining vertices in the graph will be assigned to T. Degrees of all nodes are computed in the given graph and all vertices having the degree less than T-1 will be removed from G. The graph will be updated and the degree of each vertex of the updated graph will be calculated. If the degree of any vertex of the updated graph will become less than T-1, it will be removed from the updated graph G. If no vertex having the degree less than T-1 left, it will be checked that either remaining graph is equal to K<sub>n</sub> or not The  $K_n = n(n-1)/2$  indicates the optimal maximum clique for the given graph. The complete graph indicates the optimal result. In case of the not-complete resultant graph, the branch and bound can be applied. The method either returns the optimal result of the maximum clique problem or reduces the graph for maximum clique algorithms.

## 3.1. Terminologies

Let G=(V,E) indicates a graph where V represents vertex set and E represents edge set,  $K_n = n(n-1)/2$ , where n is the number of vertices in G.  $\hat{G}$  is the complement of the graph such as  $\hat{G}=(V,\hat{E})$ , where  $\hat{E}=\{\ (vi\ ,vj)\ |\ vi\ ,vj\in V,\ i\neq j,\ (vi\ ,vj)\not\in E\ \}$ .  $P_i$  is the set of vertices in the vertex cover returned by an approximation algorithm. The deg(vi) represents the number of degree for each vertex in G. |G| and |Pi| represent the all vertices in G and Pi respectively. Vt denotes the vertices less than T value. The pseudo code of the proposed algorithm is as under:

### Pseudo code of Approximate Maximum Clique Algorithm (AMCA)

```
Algorithm: C(G = (V, E))
Input: G = (V, E)
```

Output: MC

#### **Begin:**

- 1.  $MC \leftarrow \emptyset$
- 2.  $P_i \leftarrow \emptyset$
- 3. G = (V, E')//Take the complement of G = (V, E).
- 4. Ap\_MVC())
  //Apply the approximation algorithm for MVC (AP MVC) on G= (V, E').
- 5. P<sub>i</sub>= Ap\_MVC(G = (V,E'))

  // Assign the vertex cover returned by the approximation algorithm and assign to P<sub>i</sub>
- 6.  $T \leftarrow |G| \setminus |P_i|$ //Subtract the number of vertex cover vertices from the number of vertices in G.
- 7. T←T-1

```
8.
             For i \leftarrow 1 to n {
                                             Calculate deg (v<sub>i</sub>)}
                                      i.
             //Calculate the degree of each vertex in G = (V, E)
      9.
              IF deg(v_i) < T {
                                      ii.
                                             G \leftarrow G \backslash V_t
              //Calculate those vertices from G having degree less than T value.
       10.
              Else IF |G| = K_n {
                                      i.
                                             MC \leftarrow V }
                                      ii.
                                             Go to step 12
       11.
               Else {
                                      i.
                                             Apply any Clique algorithm to reduce the graph
                                             and get MC.
                                      ii.
                                             Go to step 12 }
      12.
               } Return MC
End
```

Figure 3 contains a flow diagram of the proposed method for the better understanding of the working mechanism of the algorithm.

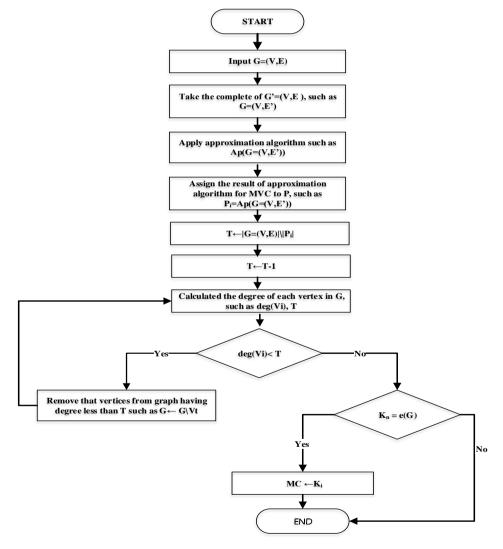


Figure. 3. Flow Diagram of the Proposed Method

### 4. Evaluation and Results

The proposed methodology has been deployed on the graph instances for better evaluation.

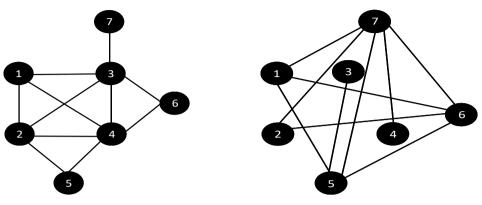


Figure 4. (a). Original Graph

(b). Complemented Graph

The graph is given in Figure 4(a) has been used for the evaluation of the algorithm. First, the complement of the given graph has been carried out as shown in Figure 4(b). Any approximation algorithm for Minimum Vertex Cover (MVC), such MDG, VSA, MVSA, MNMA, MAMA and CSSA can be deployed on the complemented graph. All these approximation algorithms have covered the complemented graph in 3 vertices, such as  $P_i \leftarrow \{1, 5, 7\}$ . Next, the minimum vertex cover vertices were removed from  $G(G\backslash P_i)$ , such as  $T_i \leftarrow \{1, 3, 4, 6\}$  and the number of vertices in  $T_i$  were calculated, such as  $T\leftarrow 4$ . Now the deletion of the vertices from G has started, the vertices from G having degree less than T-1 were deleted. The vertices  $\{5,6,7\}$  were removed from G, such as  $G\leftarrow \{1,2,3,4,5,6,7\}\setminus \{5,6,7\}$ , because these vertices had degrees less than T-1. The graph was updated as shown in Figure 5 (b) having vertices  $\{1,2,3,4\}$ . Again, the degree of each vertex of the updated graph was calculated. This time in the G has degree less than T-1, hence the  $K_n$  was calculated as 4(4-1)/2=6 which was equal to |G|. It indicates that for the given graph the  $\{1,2,3,4\}$  is optimal maximum clique as shown in Figure 5(a).

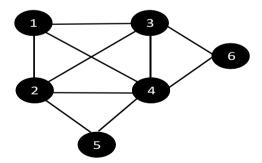
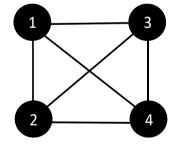


Figure 5. (a). Original Graph



(b). Updated Graph after Removing Vertices

### 5. Conclusion

This paper presents a new technique for the solution of the maximum clique problem. Most of the algorithms for maximum clique problems are exact algorithms, but in the proposed approach we have used the approximation algorithm of Minimum Vertex Cover (MVC) for the solution of the Maximum Clique (MC) problem [23]. The method provides a platform for the efficient solution of the

Maximum Clique (MC) problem by using the approximation algorithm of Minimum Vertex Cover (MVC). The proposed method was also tested on a sample benchmark instance to elaborate its working and efficiency. The algorithm has efficiently solved the Maximum Clique (MC) problem within the reduced time limit. The number of graphs used for the testing is limited, but if we consider the success of the method, it is expected that the method will perform better with other graphs as well. In this paper, small benchmark instances have been used for the testing of the proposed technique which is not enough for the exhaustive testing of the technique. The large benchmark instances like DIMACS and BHOSLIB will be the best choice for the further testing of the technique, in the future, we will improve the method and apply it on large benchmark instances for further testing and improvements [24].

## References

- [1] X-Y. Li and Y. Wang, "Simple Approximation Algorithms and PTASs for Various Problems in Wireless Ad Hoc Networks", Journal of Parallel and Distributed Computing, vol. 66, no. 4, (2006), pp. 515-530.
- [2] I. M. Bomze, M. Budinich, P. M. Pardalos and M. Pelillo, "The Maximum Clique Problem", In Handbook of combinatorial optimization, Springer US, (1999), pp. 1-74.
- [3] P. M. Pardalos and J. Xue, "The Maximum Clique Problem", Journal of Global Optimization, vol. 4, no. 3, (1994), pp. 301-328.
- [4] K. Baamann, "The Maximum Clique Problem-On Finding an Upper Bound with Application to Protein Structure Alignment", Georgia Institute of Technology, MS Thesis, (2003).
- [5] E. Tomita, Y. Sutani, T. Higashi and M. Wakatsuki, "A Simple and Faster Branch-and-Bound Algorithm for Finding a Maximum Clique with Computational Experiments", IEICE Transactions on Information and Systems, vol. 96-D, no. 6, (2013), pp. 1286-1298.
- [6] S. Pollatos, "Solving the Maximum Clique Problems on a Class of Network Graphs, with Applications to Social Networks", Monterey, California. Naval Postgraduate School, Ph.D Thesis, (2008).
- [7] D. Kumlander, "Comparing the Best Maximum Clique Finding Algorithms, which are using Heuristic Vertex Colouring", In Proceedings of the 10th WSEAS International Conference on Computers, (2006), pp. 932-937.
- [8] X. Geng, J. Xu, J. Xiao and L. Pan, "A Simple Simulated Annealing Algorithm for the Maximum Clique Problem", Information Sciences, vol. 177, no. 22, (2007), pp. 5064-5071.
- [9] R. M. Karp, "Reducibility Among Combinatorial Problems", R.E. Miller and J.W. Thatcher eds., Complexity of Computer Computations, Plenum Press, NY, (1972), pp. 85-103.
- [10] U. Feige, S. Goldwasser, L. Lovász, S. Safra and M. Szegedy, "Approximating Clique is Almost NP-Complete", In Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium, IEEE, (1991), pp. 2-12.
- [11] G. J. Woeginger, "Exact Algorithms for NP-Hard Problems: A Survey", Proceedings of 5th International Workshop on Combinatorial Optimization-Eureka, You Shrink! Papers dedicated to Jack Edmonds, M. Junger, G. Reinelt, and G. Rinaldi (Festschrift Eds.) Springer-Verlag, Lecture Notes in Computer Science 2570, (2003), pp. 184-207.
- [12] P. M. Pardalos and G. P. Rodgers, "A Branch and Bound Algorithm for the Maximum Clique Problem", Computers & Operations Research, vol. 19, no. 5, (1992), pp. 363-375.
- [13] C.-M. Li and Z. Quan, "Combining Graph Structure Exploitation and Propositional Reasoning for the Maximum Clique Problem", 2010 22nd IEEE International Conference on Tools With Artificial Intelligence (ICTAI), (2010) December 17.
- [14] K. L. Clarkson, "A Modification of the Greedy Algorithm for Vertex Cover", Information Processing Letters, vol. 16, (1983), pp. 23-25.
- [15] S. Balaji, V. Swaminathan and K. Kannan, "Optimization of Unweighted Minimum Vertex Cover", World Academy of Science, Engineering and Technology, vol. 67, (2010), pp. 508-513.
- [16] K. Imran and K. Hasham, "Modified Vertex Support Algorithm: A New Approach for Approximation of Minimum Vertex Cover", Research Journal of Computer and Information Technology Sciences, vol. 1.no. 6, (2013), pp. 7-11.
- [17] M. Fayaz and S. Arshad, "Clever Steady Strategy Algorithm: A Simple and Efficient Approximation Algorithm for Minimum Vertex Cover Problem", In Proceedings of the 2015 13th International Conference on Frontiers of Information Technology, IEEE, (2015) December 14-15.
- [18] M. Fayaz, S. Arshad, U. Zaman and A. Ahmad, "A Simple, Fast and Near Optimal Approximation Algorithm for Optimization of Un-weighted Minimum Vertex Cover", In Frontiers of Information Technology (FIT), 2016 International Conference on IEEE, (2016), pp. 176-180.

- [19] M. Fayaz, S. Arshad, A. S. Shah and A. Shah, "Max Degree Around (MDA) Algorithm: A Smart and Efficient Approximate Algorithm for Vertex Cover and Independent Set Problems", Sindh University Research Journal-SURJ (Science Series), vol. 48, no. 4D, (2016), pp. 17-26.
- [20] M. Fayaz, S. Arshad, A. S. Shah and A. Shah, "An Optimal Approximation Algorithm for Optimization of Un-Weighted Minimum Vertex Cover Problem", Sindh University Research Journal-SURJ (Science Series), vol. 48, no. (4D), (2016), pp. 175-182.
- [21] I. Khan and H. Khan, "Degree Contribution Algorithm for Approximation of MVC", International Journal of Hybrid Information Technology, vol. 7, no. 5, (2014), pp. 183-190.
- [22] I. Ahmad and M. Khan, "AVSA, Modified Vertex Support Algorithm for Approximation of MVC", International Journal of Advanced Science and Technology, vol. 67, (2014), pp. 71-78.
- [23] M. Parnas and D. Ron, "Approximating the Minimum Vertex Cover in Sublinear Time and a Connection to Distributed Algorithms", Theoretical Computer Science, vol. 381, no. 1-3, (2007), pp. 183-196.
- [24] K. Xu, "Vertex Cover Benchmark Instances (DIMACS & BHOSLIB)", International Journal of Experimental Algorithms, vol. 3, no. 1, (2012), pp. 1-18.

## **Authors**



**Muhammad Fayaz**, is currently perusing Ph.D. in Computer Science. He received MS in Computer Science from SZABIST, Islamabad, Pakistan in 2014. He did MSC from the University of Malakand, KPK, Pakistan in 2011. His areas of interest are NP problems, Approximation Algorithms, Image Processing, and Pattern Recognition.



**Shakeel Arshad** is Assistant Professor at University of Malakand, Dir(L), KPK, Pakistan, His research interest area are Genetic Algorithm and Machine learning, theoretical computer science, and more specifically in approximation algorithms, computational geometry, distributed algorithms, and randomization.



Abdul Salam Shah, is currently perusing Doctor of Philosophy (Information Technology) at Malaysian Institute of Information Technology, University of Kuala-Lumpur (UniKL), Kuala-Lumpur, Malaysia. Furthermore, he has completed MS degree in Computer Science from SZABIST, Islamabad, Pakistan in 2016. He did his BS degree in Computer Science from Isra University Hyderabad, Sindh Pakistan in 2012. In addition to his degree, he has completed short courses and diploma certificates in Databases, Machine Learning, Artificial Intelligence, Cybercrime, Cybersecurity, Networking, and Software Engineering.

He has completed specialization in Management Information System (MIS) from Virtual University of Pakistan in 2017. He has published articles in various journals of high repute. His research area includes Machine Learning, Artificial Intelligence, Digital Image Processing, Cybersecurity, and Data Mining.



**Dr. Asadullah Shah** is working as Professor and Head of Department of Information Systems (HOD) at the Kulliyyah of ICT, International Islamic University Malaysia (IIUM) before joining IIUM, he worked as Head of Telecommunication Engineering & Management department, IoBM Karachi Sindh, Dean Faculty of Computer and Management Sciences, Isra University Hyderabad Sindh and Head of Telecommunication Engineering and IT, Sukkur IBA, Sindh-Pakistan.

He did his Ph.D. from the University of Surrey UK, in 1998, with the specialization in Multimedia Communication. He started his academic career from University of Sindh Jamshoro, Pakistan in 1986 as a lecturer.

He has published 200 research articles in highly reputable international and national journal in the field of computers, communication and IT. Also, he has published 12 books in his 30 years of the academic career. Currently, he is supervising a great number of postgraduate students, working in multiple disciplines, especially, animation, social media and image processing in the Department of Information Systems, Kulliyyah of Information and Communication Technology, International Islamic University Malaysia.