QUALITY EVALUATION MEASURES OF GENETIC ALGORITHM AND INTEGER LINEAR PROGRAMMING FOR MINIMUM DOMINATING SET PROBLEM

ADEL A. ALOFAIRI[1*], RASHAD ISMAIL[2], EMAD MABROUK[3], FISAL SAEED[4],

IBRAHIM E. ELSEMMAN[5]

[1,3,5]Department of Mathematics, Faculty of Science Assiut University, Egypt.
[12]Department of Computer Science, Faculty of Science, Ibb University, Yemen.
[2]Department of Mathematics, King Khalid University, Mohail Assir, Saudi Arabia.
[3]College of Engineering and Technology, American University of the Middle East, Kuwait.
[4]College of Computer Science and Engineering, Taibah University, Medina, Saudi Arabia.
E-mail: [1]adelofairi@gmail.com, [2]rismail@kku.edu.sa, [3]emad.mabrouk@aum.edu.kw,
[4]fsaeed@taibahu.edu.sa.[5]elsemman@aun.edu.eg.
(*corresponding author: Adel A. Alofairi)

ABSTRACT

In network science, controlling the elements of complex networks with a few numbers of nodes has recently become a significant subject of research and a major challenge. Nowadays, the minimum dominating set (MDS) represents an important modern network topic in this context. During the last decade, many methods have been developed to solve the MDS problem and generate different solutions for the same graph. In this paper, we evaluated the performance of two approaches to solve the minimum dominating set problem: the first one is based on the genetic algorithm, called a hybrid of a genetic algorithm (HGA-MDS), and the second approach is based on the integer linear programming (ILP-MDS). We proposed three measures to evaluate the quality of the obtained solution, which are the domination number, the nodes degree, and the betweenness centrality. We utilized a number of publically available benchmark test and real-world graph data sets. The experimental results have shown that the ILP-MDS outperforms HGA-MDS in calculating the domination number, the optimal solution, and in handling the big data graphs. The results also showed the close performance of two methods in calculating the average of both nodes degree and betweenness centrality for the obtained best dominating set.

Keywords: *Minimum dominating set; Domination number; Genetic algorithm; Integer linear programming; Betweenness centrality; Nodes degree.*

1. INTRODUCTION

The topic of controlling network elements with a few numbers of nodes has recently become a significant research field [1, 2]. Moreover, it represents an attractive topic in many applications that depend on the network representation of their components, which vary between the field of wireless networks, social networks, biological networks, and the growth of these areas [3-5]. Minimum dominating set (MDS) represents a small-optimized subset of graph nodes, such that each other node must be adjacent to at least one of them [6]. MDS had recently gained much attention and application in many domains [7]. However, computing the MDS is NP–hard problem that cannot

be solved in polynomial time [6]. To obtain the optimal or even nearer to the optimal solutions for the MDS problem, a wide variety of techniques have been developed. In this context, Meta-heuristics methods [8, 9] and integer linear programming methods (ILP) [10, 11] represent the most popular methods and they have given valuable solutions.

Meta-heuristics are optimization methods that orchestrate an interaction between local improvement procedures and higher-level strategies to create a process capable of escaping from local optima and performing a robust search of a solution space [12]. Conceptually, various meta-heuristic algorithms, which are derived from the behavior of biological and physical systems in nature, have been

proposed as powerful methods for the MDS problem and they have given valuable solutions [13-16]. The efficiency of existing meta-heuristic algorithms are still not satisfactory, especially for hard and large-scaled instances. Among those methods, the genetic algorithm (GA) is one of the best methods developed to solve the MDS problem [13].

The GA is a stochastic search method that is inspired by natural biological evolution to generate useful solutions to optimization and search problems [17]. Nowadays, GA is considered to be the most widely known and applicable type of meta-heuristics algorithm, and it is a population-based search methodology [9]. The GA is widely used to solve real-world problems by representing the problem using GA, design the fitness function that controls the quality of the solution, and tuning the required parameters including population structure, population size, a sequence of genetic operators, the operator's parameters, and termination conditions [18]. Although the GA has shown successful performance in solving many combinatorial searches, the resulting solution still needs to be evaluated for improvement. Actually, developing a measure for evaluating search algorithms is difficult without distinctly defined evaluation aims and related certain criteria [12].

Hedar and Ismail in [13] proposed a hybrid of a genetic algorithm and local search for MDS with a specific fitness function. This algorithm is called a hybrid genetic algorithm for the MDS problem (HGA-MDS). Based on the obtained results from wide range of experiments, the proposed algorithm was successful and promising. However, this algorithm has yet to be implemented with large and real data and therefore, needs the real evaluation of its performance and quality of the solution. Furthermore, as a mathematical optimization approach, integer linear programming (ILP) is a highly feasible way of finding the solution of many problems in reality [19, 20]. Specifically, Nacher and Akutsu in [10]proposed an integer linear programming representation (ILP-based) model to determine an optimal solution for the MDS problem. Practically, this ILP-Based model was used to find driver sets and analysis big real word data networks [21-23].

Several previous studies have emphasized the importance of nodes' degree and centrality in playing important roles in network architecture [24, 25]. Moreover, several studies confirmed that the node degree and betweenness are important. For example, the network properties of biologically central genes exhibit some topological centrality compared to the rest of proteins in the consideration network and high-betweenness preferential in detecting biological central genes [22, 26, 27]. Based on the literature, we introduced three quality measures: domination number, nodes degree, and nodes centrality test for evaluating the quality of the generating solutions using the HGA-MDS and ILP-MDS methods.

This paper is organized as follows. In the next section, we briefly presented the MDS problem as preliminaries needed throughout the paper. In addition, we highlighted the HGA-MDS and ILP-MDS methods. Section 3 describes the three measures to evaluate the performance of HGA-MDS and ILP-MDS methods. In Section, 4 we present the conducted experimental setup, while the results and discussion are presented in Section 5. Finally, the conclusion makes up Section 6.

2. PRELIMINARIES

2.1 Domination in Graphs
2.1.1 Graph representation

A graph is a mathematical structure that is used to model pairwise relations between objects [1].
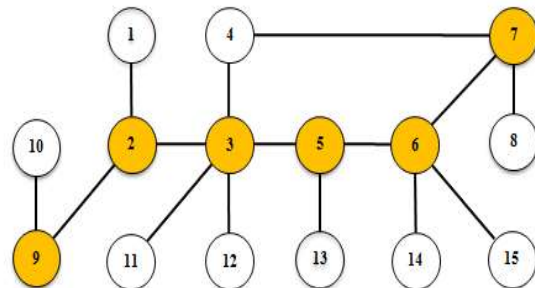

Figure 1: The example of graph. The circles represent the nodes (15) and the lines represent the edges (15).

It is a vital data structure to represent and analyze the real applications of networks in certain areas such as data mining, natural language processing, and information retrieval [2, 3]. Ordinarily, a graph can be mostly denoted by a pair $G = (V, E)$ that is made up of vertices (also called nodes or points) which are denoted by a set of V and connected by set of edges (also called links or lines) that are denoted by set of E. Figure 1 shows a simple example of graph in which circles represent the nodes (15) and lines represent the edges (15).

As a complete description, usually, the graph can be represented by the list of links between nodes. For example, the network in toy example of Figure 1 can be described by a list of ordered pairs like this $\{(1,2), (2,3), (2,9), ...\}$. However, for mathematical representation, it is confidence to use the adjacency matrix to represent the network. The adjacency matrix $A(n \times n)$ of undirected graph $G = (V, E)$ is a matrix with rows and columns indexed by V.

Figure 2 shows the adjacency matrix that represents the network example in Figure 1 in simple form in which, $A_{ij} = 1$ if node i is connected to the node j in the given graph, and $A_{ij} = 0$ if the node i is not connected to the node j in the given graph.

Figure 2: The adjacency matrix representation for example graph in Figure 1.

2.1.2 Graph domination

The concept of domination in graphs is a rapidly developing area of research in graph theory. The concept of domination has existed for a long time and early discussions on the topic can be found in the works of Ore [28] and Berge [5]. Currently, domination is considered to be one of the fundamental concepts in graph theory and its various applications to ad hoc networks, biological networks, distributed computing, social networks,

and web graphs [6, 7, 11]. This partly explains the increased interest. Such applications usually aim to select a subset of nodes that will provide some definite service such that every node in the network is close to some node in the subset. Hence, the MDS covers all the targeted graph nodes.

2.2 The Minimum Dominating Set Problem

An undirected graph $G = (V, E)$ consists of a set V of vertices and a set E of edges. A dominating set is a subset of vertices $D \subseteq V$ such that for all $u \in V - D$, there exist a node $v \in D$, for which $uv \in E$ (we say that D dominates V). This means that each vertex is either a member of the dominating set or it is adjacent to some member of the dominating set (see Figure 1 black nodes). The MDS problem is that of finding a dominating set of minimum cardinality in a graph. The size of a MDS in a graph G is called the domination number of G and is denoted by $\gamma(G)$. The MDS problem is a fundamental problem in algorithmic graph theory. This is one of the central problems of combinatorial optimization, classified as NP–hard problem [11, 29]. A wide range of methods have been used to acquire the optimal, or even near the optimal solutions for the MDS problem like meta-heuristics methods [12, 13].

2.3 Genetic Algorithm

Genetic algorithms were invented by Holland in [30], early 1975, as a search heuristic that mimics the process of natural evolution. This heuristic is routinely used to generate useful solutions to optimization and search problems. It belongs to the larger class of evolutionary algorithms, which generates a solution to optimization problems using techniques inspired by natural evolution such as inheritance, selection, crossover, and mutation. In other words, the GA is a procedure that tries to mimic the genetic evolution of a species. Specifically, GA simulates the biological processes that allow the consecutive generations in a population to adapt to their environment. The adaptation process is mainly applied through genetic inheritance from parents to children and through survival of the fittest.

GAs were limitedly applied until their multipurpose presentation of Goldberg [17] in search, optimization, design, and machine learning areas. Nowadays, GAs are considered to be the most

widely known and applicable type of meta-heuristics, and it is a population-based search methodology [31]. The GA have shown successful performance in solving many combinatorial search problems as MDS problem on graph [13], but the resulting solution still needs to be evaluated for improvement.

2.4 Hybrid Genetic Algorithm for MDS

In [13] a hybrid algorithm based on a genetic algorithm in addition to local search strategies for the MDS problem (HGA-MDS) was developed. This algorithm uses a $0 - 1$ variable representation of solutions in searching for the MDS, and it invokes a new fitness function to measure the solution qualities. New fitness function and intensification elements have been set within HGA-MDS to achieve better performance and fit the problem. Based on the obtained results from wide range of experiments, the proposed algorithm was successful and promising.

2.5 ILP-Based MDS Model

As a mathematical optimization program, integer linear programming (ILP), where some or all the variables are bounded to be integers, is a highly feasible way of finding the solution of many problems in reality. In addition, constraints and objective function are linear, in case of $0 - 1$ variable representation it is called the binary integer linear programming problem [19]. ILP is used to represent and solve the NP-Complete problems [20]. To solve the MDS problem, Nacher et.al in [10] introduced the following integer linear programming formulation:

$$\text{objective}: \quad \min \sum_{j=1}^{n} x_j$$

$$\text{subject to.} \qquad\qquad\qquad (1)$$

$$\sum_{j=1}^{n} A_{ij} x_j \geq 1 \quad i = 1,2,\dots,n$$

$$x_j \in \{0,1\}$$

The network graph can be described by the adjacency matrix $A(n \times n)$, as Figure 2, where n is the number of nodes in the given network, $A_{ij} = 1$ if the node i interacts with the node j or $i = j$, and $A_{ij} = 0$ otherwise. Where n represents the number of nodes in the targeted graph. The solution of the

problem in Equation (1) is the binary vector x, where $x_i = 1$ if node i belongs to the generated MDS and $x_i = 0$ otherwise. This ILP formula can be applied to both directed and undirected networks. The domination number $\gamma(G)$ of a network G is the number of nodes in an MDS. After obtaining an MDS by solving problem in Equation (1) we can calculate the domination number as follows:

$$\gamma(G) = \sum_{i \in V} x_i \qquad\qquad (2)$$

The ILP-based method to solve the MDS problem is called shortly ILP-MDS.

3. QUALITY MEASURES

In this section, we review three famous measures that are used to reflect the quality of resulted minimum dominating sets.

3.1 Domination Number Measure

The domination number $\gamma(G)$, in Equation (2), measures the size $|D|$ of the resulted minimum dominating set D for the graph under consideration:

3.2 Nodes Degree Measure

Degree centrality is a simple measure that counts how many neighbors a node has. In other words, Degree centrality of a node v_i is the number of interacting partners of v_i and computed as:

$$deg(v_i) = \sum_{j=1, i \neq j}^{n} a_{ij}.$$

3.3 Betweenness Centrality Measure

Centrality defines how important a node is within a network. According to [22, 24, 25], the betweenness centrality is a global measure of centrality of the vertex v in a graph $G = (V, E)$ which indicating an interactions appearance in shortest paths through the whole network. The betweenness centrality of a node u can be calculated as following:

$$B_C(u) = \sum_{i \neq j \neq u \in V} \sigma_{ij}(u) / \sigma_{ij},$$

where σ_{ij} means the number of shortest paths between node i and j, and $\sigma_{ij}(u)$ is the number of shortest paths between node i and j running through the node u.

4. NUMERICAL EXPERIMENTS

The formulation of the MDS problem includes the initialization for the matrix that represents the constraints and two vectors for the objective function and the set of variables. We implemented the HGA-MDS method using Matlab programming environment (Mathwork Inc.). For implementing the ILP-based model in Equation (1), we utilized both MOSEK and Guropi optimization solvers (academic licenses) with Matlab. Moreover, all experiments in this section were implemented on a system with processor Intel(R) Core (MT) i5 2.53 GHz and 4.0 GB RAM. To compare the performance of the HGA-MDS method against the ILP-MDS method by the three quality evaluation measures, we used several test graphs from the literature [13].

The HGA-MDS was run ten times on each instance in the first test graphs in Subsection 4.1.1, and twenty times on each instance in the second test graphs in Subsection 4.1.2. Several numerical results have been reported in this section. All parameters used in HGA-MDS with their assigned values are shown in Table 1.

Table 1. Parameter configuration of HGA-MDS

Parameter	Definition	Value
P_{Size}	Size of the initial population.	40
P_c	Crossover probability.	0.8
P_m	Mutation probability.	0.01
η_{min}	Min expected values of the selection operator.	0.9
η_{max}	Max expected values of the selection operator.	1.1
$nStep$	Number of nodes used in the local search.	2
$DomiSet$	Max number of the best dominating sets used to update DS.	10
$CoreNo$	Prespecified number of best dominating sets used to compute x^{Core}	3
N_g	Max number of generations	100

4.1 Benchmark Test Graphs

We considered two different groups of test graphs of the MDS problem provided in the literature, such as [13, 32]. The first group of test graphs contains graphs with 400 to 800 nodes, while the second group contains graphs with 80 to 400 nodes.

4.1.1 First test graphs

We adopted a referenced first test graph construction procedure explained in [13, 32]. The first set of test graphs contains 18 different instances of the MDS problem created from the two graphs $G_{p,d}^{400}$ and $G_{p,d}^{800}$, where p is the density of a graph is defined to be the number of edges in the graph divided by the number of edges of the complete graph of the same size and d is the domination number $\gamma(G)$, see Table 2.

4.1.2 Second test graphs

Following the generation instructions shown in [13] we generated the second set of test graphs that contains 32 instances of the MDS problem created from the six graphs $G_r^1 - G_r^6$, with different values of the range r. More details can be seen in Table 3.

4.2 Dense Random Graphs Dataset

We generated typical dense connected random graphs class of Erdoes-Renyi random graphs [33]. These graphs have been generated using the network workbench tool, the official version [34]. All details of the generated graphs are shown in Table 4, where n represents the number of nodes, p represents the linking probability, and $|E|$ represents the number of edges in the graph.

Table 2. First test problems

Test Graphs	No. of nodes	Density (p)	Domination No. (d)	No. of instances
$G_{0.1,d}^{400}$	400	0.1	$8, 11, 14, 23$	4
$G_{0.3,d}^{400}$	400	0.3	$3, 5, 8, 11$	4
$G_{0.5,d}^{400}$	400	0.5	$3, 8, 11$	3
$G_{0.1,d}^{800}$	800	0.1	$11, 14, 18$	3
$G_{0.3,d}^{800}$	800	0.3	$3, 5$	2
$G_{0.5,d}^{800}$	800	0.5	$3, 6$	2

Table 3. Second test problems

Test Graphs	No. of nodes	Area	Range (r)	No. of instances
G_r^1	400	3000×3000	210-240	4
G_r^2	350	2500×2500	200-230	4
G_r^3	300	2000×2000	180-220	5
G_r^4	200	1000×1000	100-160	7
G_r^5	100	600×600	80-120	5
G_r^6	80	400×400	60-120	7

Table 4: The dense random graphs dataset

Random Graphs	n	p	$|E|$	Min deg	Max deg	Avg. deg	Density
RG-1000	1000	0.5	498,176	452	548	497.2	0.9953
RG-2000	2000	0.5	1,997,242	927	1,080	998.6210	1.0001
RG-3000	3000	0.1	901,214	237	361	300.4	0.2003

5. RESULTS AND DISCUSSIONS

Here, we analyzed the performance of HGA-MDS and ILP-MD for the MDS problem. We used three measures to evaluate the performance of the two methods. These measures are the domination number, the nodes degree, and the betweenness centrality. The best solution (Best) measure gives the minimum number of nodes in the best solution found in all independent runs (around 20 times). This represents the best domination number of the given graph as shown in the result tables. Whereas, percentage (Hits) is the measure that gives the percentage of the optimal solution found throughout the independent runs. Initially, we measured the best domination number produced by HGA-MDS compared with ILP-MDS.

Tables 5 and 6 along with their graphical representation in Figure 3 show that ILP-MDS performs better than HGA-MDS in calculating the best domination numbers for the first test graphs $G_{p,d}^{400}$ and $G_{p,d}^{800}$. Moreover, we measured the best domination number produced by HGA-MDS and ILP-MDS. Results in Table 7 and Figure 4 reveal the superiority of the ILP-MDSet method compared with the HGA-MDS method for all datasets $G_r^1 - G_r^6$.

Table 5. Results of running HGA-MDS and ILP-MDS on $G_{p,d}^{400}$

No. of Nodes	No. of Edges	P	Optimal (d)	HGA-MDS Avg.	HGA-MDS Hits (%)	LIP-MDS Hits (%)
400	8944	0.1	8	8	100	100
400	8278	0.1	11	11.1	90	100
400	10606	0.1	14	14.4	60	100
400	16716	0.1	23	24.2	40	100
400	39394	0.3	3	3	100	100
400	33808	0.3	5	5.3	70	100
400	15600	0.3	8	8.1	90	100
400	8738	0.3	11	11	100	100
400	59062	0.5	3	3.1	90	100
400	6158	0.5	8	8	100	100
400	8338	0.5	11	11	100	100

Table 6. Results of running HGA-MDS and ILP-MDS on $G_{p,d}^{800}$

No. of Nodes	No. of Edges	P	Optimal (d)	HGA-MDS Avg.	HGA-MDS Hits (%)	LIP-MDS Hits (%)
800	31426	0.1	11	11.3	70	100
800	21058	0.1	14	14	100	100
800	27262	0.1	22	22.5	50	100
800	161478	0.3	3	7.9	60	100
800	150730	0.3	5	6.8	50	100
800	244526	0.5	3	4.4	50	100
800	203442	0.5	6	6.5	80	100

Table 7. Results of running HGA-MDS and ILP-MDS on graphs $G_r^1 - G_r^6$, where r is the range.

| Test Graphs	No. of Nodes	Domination number ILP-MDS	Domination number HGA-MDS		Test Graphs	No. of Nodes	Domination number ILP-MDS	Domination number HGA-MDS
G_{210}^1	400	67	79		G_{200}^2	350	54	67
G_{220}^1	400	62	77		G_{210}^2	350	47	63
G_{230}^1	400	57	73		G_{220}^2	350	43	55
G_{240}^1	400	53	70		G_{230}^2	350	41	51
G_{180}^3	300	42	54		G_{80}^5	100	18	19
G_{190}^3	300	39	48		G_{90}^5	100	15	16
G_{200}^3	300	35	41		G_{100}^5	100	13	14
G_{210}^3	300	33	40		G_{110}^5	100	11	11
G_{220}^3	300	31	36		G_{120}^5	100	9	10
G_{100}^4	200	35	39		G_{60}^6	80	15	15
G_{110}^4	200	30	35		G_{70}^6	80	12	13
G_{120}^4	200	23	27		G_{80}^6	80	9	10
G_{130}^4	200	22	26		G_{90}^6	80	8	8
G_{140}^4	200	20	23		G_{100}^6	80	7	7
G_{150}^4	200	17	21		G_{110}^6	80	6	6
G_{160}^4	200	16	20		G_{120}^6	80	5	5
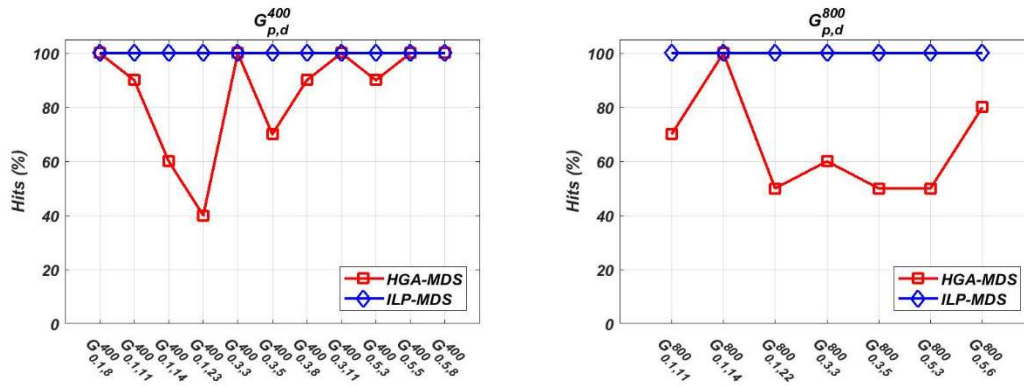
Figure 3. Percentages of the optimal solution hits of HGA-MDS and ILP-MDS on $G_{p,d}^{400}$ and $G_{p,d}^{800}$, with different values for the density p and domination number d.
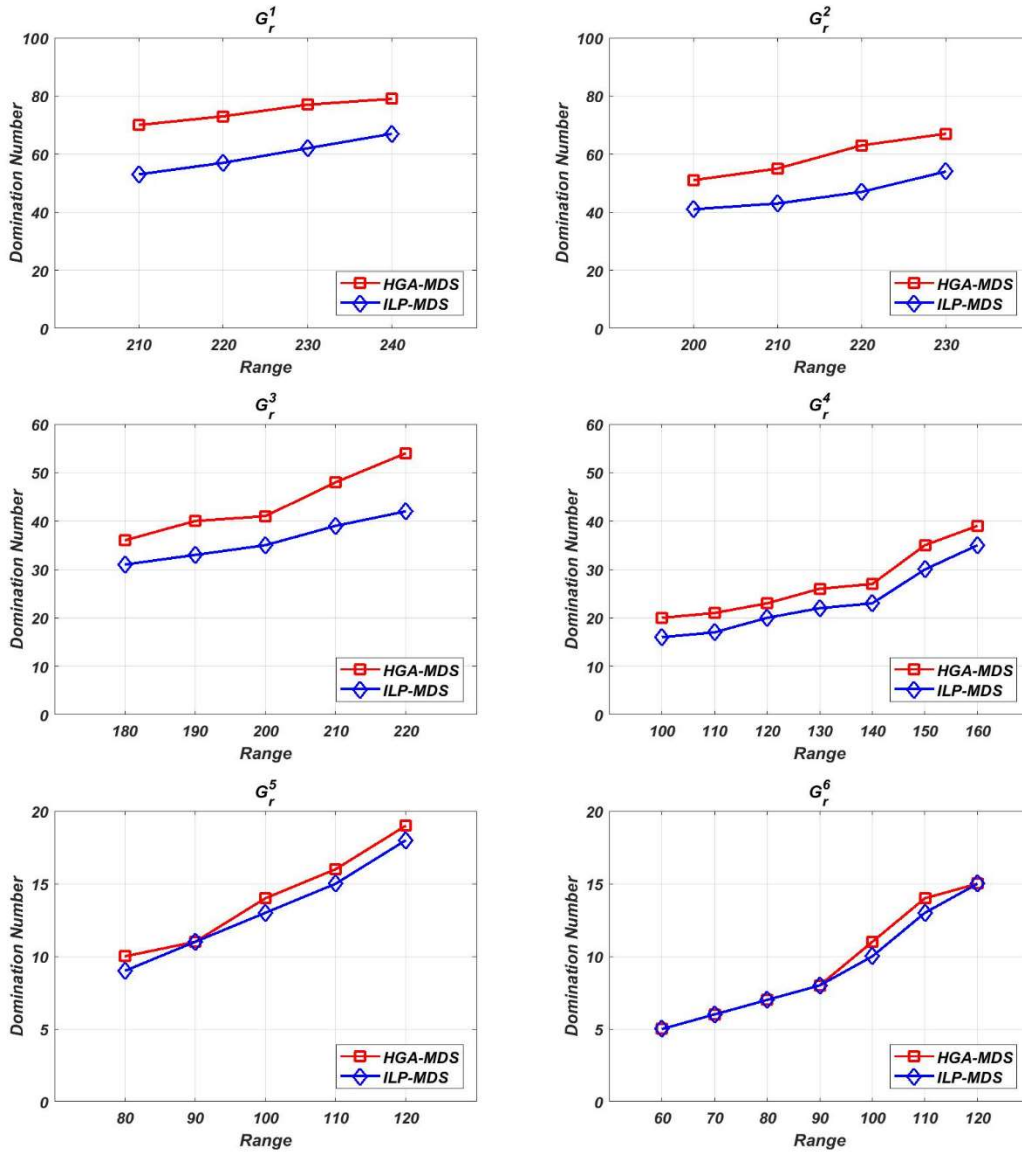

Figure 4. Domination number measure comparisons between HGA-MDS and ILP-MDS on graphs $G_r^1 - G_r^6$, with different values for the range r.

Tables 8 and 9 together with Figures 5 and 6 show the average values of node degrees and betweenness centrality measures calculated from the MDS obtained by HGA-MDS and ILP-MDS for G_{210}^1, G_{220}^1, G_{200}^2, G_{210}^2, G_{180}^3, G_{190}^3. For the node degree measure, the minimum, average and maximum values of the obtained dominating sets are presented in Table 8 and Figure 5. Observably, the results showed the close performance of the two methods in calculating the average of node degrees for the obtained best dominating sets. For the betweenness centrality measure, the minimum, average, and maximum values of the obtained best dominating sets are presented in Table 9 and Figure 6 that shows the close performance of the two methods in calculating the average values of the betweenness centrality for the obtained dominating sets with a relative advantage of ILP-MDS. The result reveals that ILP-MDS is superior and provides the optimal solution for every instance and shows noticeably higher percentages (hits) for all instances

of the MDS problem. For HGA-MDS, results are not guaranteed.

Additionally, Table 10 reports the implementation results of the considered two methods on the random graphs that are described in Table 4. This class of graphs has a high density. Therefore, the ILP solver needs to check all the alternatives and don't give the final result as the experiments asserted. However, we found that if we interrupt the ILP solver execution, it gives result but there is no guarantee of the solution. The results with this type of graph also showed a preference for the solvers in some cases, where the Guropi was more accurate than the MOSEK. These results reveal the high performance of the HGA-MDS method for this class of random graphs. From this point, it appears significant to study different methods for MDS problem with different graphs kinds and sizes.

Table 8. Evaluate the performance of HGA-MDS and ILP-MDS by the degree measure.

Test Graphs	Degree			HGA-MDS			ILP-MDS		
	Min.	Avg.	Max.	Deg. Min.	Deg. Avg.	Deg. Max.	Deg. Min.	Deg. Avg.	Deg. Max.
G_{210}^1	1	7	16	1	6	14	1	6	16
G_{220}^1	1	8	16	1	7	13	1	7	16
G_{200}^2	2	8	19	2	7	18	3	7	19
G_{210}^2	2	8	20	2	8	20	2	8	19
G_{180}^3	2	8	15	2	9	15	3	8	15
G_{190}^3	2	9	20	2	9	17	2	9	16

Table 9. Evaluate the performance of HGA-MDS and ILP-MDS by the betweenness centrality measure.

Test Graphs	Betweenness Centrality			HGA-MDS			ILP-MDS		
	Min.	Avg.	Max.	Cent. Min.	Cent. Avg.	Cent. Max.	Cent. Min.	Cent. Avg.	Cent. Max.
G_{210}^1	0	0.0199	0.4516	0	0.0156	0.4516	0	0.0231	0.2917
G_{220}^1	0	0.0251	0.3529	0	0.0242	0.2806	0	0.0291	0.2255
G_{200}^2	0	0.0239	0.3272	0	0.0380	0.3201	0	0.0281	0.2796
G_{210}^2	0	0.0222	0.5228	0	0.0254	0.2385	0	0.0364	0.3028
G_{180}^3	0	0.0312	0.5514	0	0.0397	0.2954	0	0.0415	0.4156
G_{190}^3	0	0.0322	0.4285	0	0.0376	0.4172	0	0.0400	0.4172

Table 10: Results of the dense random graphs dataset

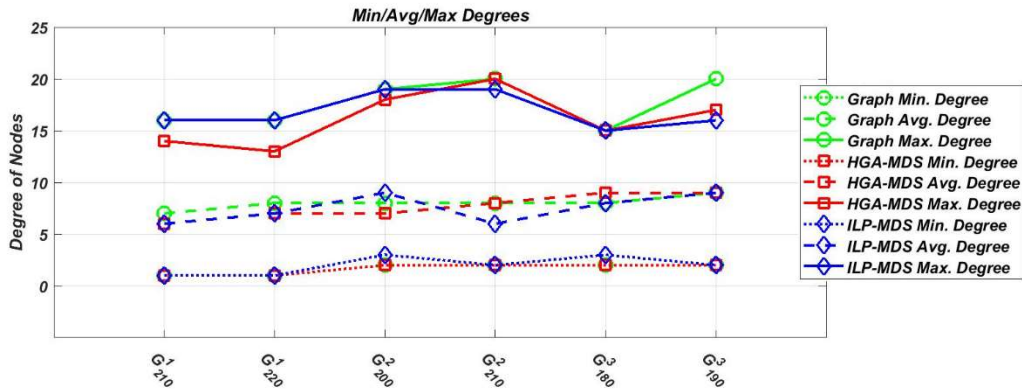| Test Graphs	*No. of Nodes*	p	$|E|$	Density	ILP-MDSet	HGA-MDS
RG-1000	1000	0.5	498,176	0.9953	6	6
RG-2000	2000	0.5	1,997,242	1.0001	8	7
RG-3000	3000	0.1	901,214	0.2003	47	31

Figure 5. Nodes degree measure comparisons between HGA-MDS and ILP-MDS for the graphs G_r^1, G_r^2 and G_r^3, with different values of the range r.
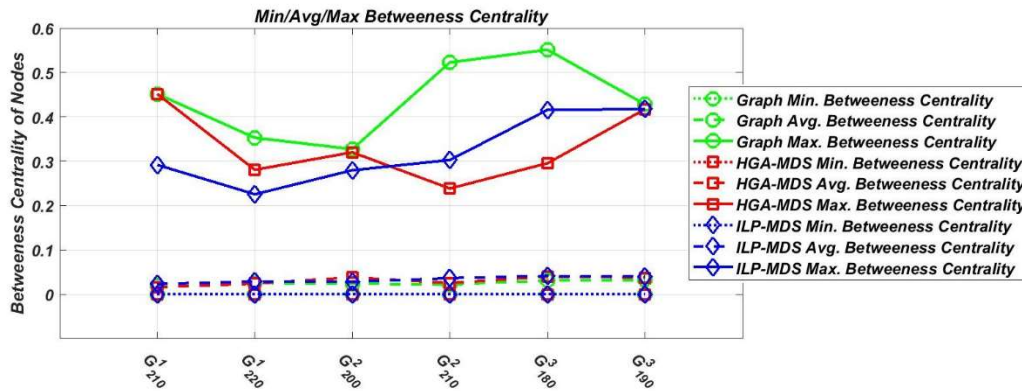

Figure 6. Betweenness centrality measure comparisons between HGA-MDS and ILP-MDS for the graphs G_r^1, G_r^2 and G_r^3, with different values of the range r.

6. CONCLUSION

In this paper, we evaluated the performance of two approaches that are used for solving the minimum dominating set problem: HGA-MDS and ILP-MDS. We used three quality evaluation measures to evaluate the quality of the obtained solutions. These measures are the domination number, the nodes degree, and the betweenness centrality. The experimental results on different standard benchmark test graphs show the efficiency of the two methods in terms of the quality of solution to compute and identify MDS. In addition, ILP-MDS outperforms HGA-MDS in calculating the domination number and the optimal solution. The results showed a close performance of the two methods in calculating the average of both node degrees and betweenness centrality for the obtained best dominating sets. However, despite the high performance of the GA-based method, there is a need to improve it from all the discussed aspects and applying on a real world with big data networks. In addition, using more domain-specific quality measures can help for further analysis. Moreover, we highly recommended the ILP-based MDSet model to be the measure of domination number performance for meta-heuristic-based-MDS methods. We observed that ILP-MDSet model faced a problem with solution search in the dense graph because of the large number of edges (high density) and the branch and bound strategy which model uses to solve the problem, hence, it needs to improve with this type of graph.

REFERENCES

[1]   Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *nature,* vol. 473, no. 7346, pp. 167, 2011.

[2]   D. Jungnickel, and D. Jungnickel, *Graphs, networks and algorithms*: Springer, 2005.

[3]   A.-L. Barabási, "Linked: how everything is connected to everything else and what it means for business," *Science, and Everyday Life. Plume Books,* vol. 2, no. 6.3, pp. 1, 2003.

[4]   A.-L. Barabási, *Network science*: Cambridge university press, 2016.

[5]   C. Berge, "The theory of graphs and its applications, Methuen & Co," *Ltd., London*, 1962.

[6]   T. W. Haynes, S. T. Hedetniemi, and P. J. Slater, "Fundamentals of domination in graphs Marcel Dekker," *Inc., New York*, 1998.

[7]   F. Molnár, S. Sreenivasan, B. K. Szymanski, and G. Korniss, "Minimum dominating sets in scale-free network ensembles," *Scientific reports,* vol. 3, pp. 1736, 2013.

[8]   T. Bäck, D. Fogel, and Z. Michalewicz, "Evolutionary Computation 1: Basic Algorithms and Operators. Inst," of Physics Publishing, 2000.

[9]   T. Bäck, D. B. Fogel, and Z. Michalewicz, *Evolutionary computation 2: Advanced Algorithms and Operators*: CRC press, 2000.

[10]   J. C. Nacher, and T. Akutsu, "Analysis of critical and redundant nodes in controlling directed and undirected complex networks using dominating sets," *Journal of Complex Networks,* vol. 2, no. 4, pp. 394-412, 2014.

[11]   J. C. Nacher, and T. Akutsu, "Dominating scale-free networks with variable scaling exponent: heterogeneous networks are not difficult to control," *New Journal of Physics,* vol. 14, no. 7, pp. 073005, 2012.

[12]   D. Whitley, S. Rana, J. Dzubera, and K. E. Mathias, "Evaluating evolutionary algorithms," *Artificial intelligence,* vol. 85, no. 1-2, pp. 245-276, 1996.

[13]   A.-R. Hedar, and R. Ismail, "Hybrid genetic algorithm for minimum dominating set problem." pp. 457-467.

[14]   C. K. Ho, Y. P. Singh, and H. T. Ewe, "An enhanced ant colony optimization metaheuristic for the minimum dominating set problem," *Applied Artificial Intelligence,* vol. 20, no. 10, pp. 881-903, 2006.

[15]   S. A. Abed, and H. M. Rais, "Hybrid bat algorithm for minimum dominating set problem," *Journal of Intelligent & Fuzzy Systems,* vol. 33, no. 4, pp. 2329-2339, 2017.

[16]   A.-R. Hedar, and R. Ismail, "Simulated annealing with stochastic local search for minimum dominating set problem," *International Journal of Machine Learning and Cybernetics,* vol. 3, no. 2, pp. 97-109, 2012.

[17]   D. E. Goldberg, "Genetic algorithms in search," *Optimization, and MachineLearning*, 1989.

[18]   K. Sugihara, "Measures for performance evaluation of genetic algorithms." pp. 172-175.

[19]   L. A. Wolsey, and G. L. Nemhauser, *Integer and combinatorial optimization*: John Wiley & Sons, 2014.

[20]   A. Schrijver, *Theory of linear and integer programming*: John Wiley & Sons, 1998.

[21]   X.-F. Zhang, L. Ou-Yang, Y. Zhu, M.-Y. Wu, and D.-Q. Dai, "Determining minimum set of driver nodes in protein-protein interaction networks," *BMC Bioinformatics,* vol. 16, no. 1, pp. 146, 2015.

[22]   S. Wuchty, "Controllability in protein interaction networks," *Proceedings of the National Academy of Sciences,* vol. 111, no. 19, pp. 7156-7160, 2014.

[23]   J. C. Nacher, and T. Akutsu, "Minimum dominating set-based methods for analyzing biological networks," *Methods,* vol. 102, pp. 57-63, 2016.

[24]   L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, pp. 35-41, 1977.

[25]   A. M. M. González, B. Dalsgaard, and J. M. Olesen, "Centrality measures and the importance of generalist species in pollination networks," *Ecological Complexity,* vol. 7, no. 1, pp. 36-43, 2010.

[26]   T. Milenković, V. Memišević, A. Bonato, and N. Pržulj, "Dominating biological networks," *PloS one,* vol. 6, no. 8, pp. e23016, 2011.

[27]   J. C. Nacher, M. Ishitsuka, S. Miyazaki, and T. Akutsu, "Finding and analysing the minimum set of driver nodes required to control multilayer networks," *Scientific reports,* vol. 9, no. 1, pp. 576, 2019.

[28] O. Ore, "Theory of graphs (Amer. Math. Soc. Coll. Publ. 38)," *Amer. Math. Soc., Providence*, pp. 49-53, 1962.

[29] M. R. Gary, and D. S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-completeness," WH Freeman and Company, New York, 1979.

[30] J. H. Holland, "Adaptation in natural and artificial systems Ann Arbor," *The University of Michigan Press,* vol. 1, pp. 975, 1975.

[31] A.-R. Hedar, R. Ismail, G. A. El-Sayed, and K. M. J. Khayyat, "Two Meta-Heuristics Designed to Solve the Minimum Connected Dominating Set Problem for Wireless Networks Design and Management," *Journal of Network and Systems Management*, pp. 1-41, 2018.

[32] L. A. Sanchis, "Experimental analysis of heuristic algorithms for the dominating set problem," *Algorithmica,* vol. 33, no. 1, pp. 3-18, 2002.

[33] R. Albert, and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of modern physics,* vol. 74, no. 1, pp. 47, 2002.

[34] N. Team, "Network Workbench Tool. Indiana University, Northeastern University, and University of Michigan," 2006.