Optimization
Iranian Journal of Optimization
Volume 13, Issue 2, 2021, 115-121
Research Paper

Online version is available on: www.ijo.rasht.iau.ir

Islamic Azad University
Rasht Branch
ISSN: 2588-5723
E-ISSN:2008-5427

A new Method for Solving of the Graph Coloring Problem a Based on Fuzzy Logic and Whale Optimization Algorithm

Taha Mostafaie [1], Farzin Modarres Khiyabani[*1], Nima Jafari Navimipour[2] and Behrooz Daneshian[3]

[1]Department of Mathematic, Tabriz Branch, Islamic Azad University, Tabriz, Iran
[2]Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran
[3]Department of Mathematics, Central Tehran Branch, Islamic Azad University, Tehran, Iran

Revise Date: 10 August 2021
Accept Date: 19 December 2021

Abstract

In recent years, Graph Coloring Problem (GCP) is one of the main optimization problems from literature. Many real world problems interacting with changing environments can be modeled by dynamic graphs. Graph vertex coloring with a given number of colors is a well-known and much-studied NP-hard problem. Meta-heuristic algorithms are a good choice to solve GCP because they are suitable for problems with NP-hard complexity. But, in many previously proposed algorithms, there are common problems such as runtime algorithm and low convergence of algorithm. Therefore, in this paper, we propose the Fuzzy Whale Optimization Algorithm (FWOA), a variety of basic Whale Optimization Algorithm (WOA), to improve runtime and convergence of algorithm in the GCP. Since WOA at first was introduced to solve continue problem, we need a discrete WOA. Hence, to use FWOA to discrete search space, the standard arithmetic operators such as addition, subtraction and multiplication extant in FWOA encircling prey, exploitation phase and exploration phase operators based on distance's theory needs to be redefined in the discrete space. Parameters p, r are defined randomly in the WOA algorithm in FWOA algorithm defined as fuzzy and are selected in fuzzy tragedy. A set of graph coloring benchmark problems are solved and its performance is compared with some well-known heuristic search methods. Results illustrate that FWOA algorithm is the original focus of this work in most cases success rate is nearly 100% and the runtime and convergence algorithm has been improved on some graphs. But as we have illustrated that compared with other manners, we cannot deduce that our algorithm is the best in all instance of graphs. It can be said that a proposed algorithm is able to compete with other algorithms in this context. Obtained results approved high performance of proposed method.

Keywords:
Graph benchmark
Graph coloring
Discrete optimization
Runtime

***Correspondence E-mail**: f.modarres@iaut.ac.ir

INTRODUCTION

A graph in simple phrase is a diagram which illustrates a system of connectors or interconnections among two or more existence with succor of dots, lines, shafts etc. A graph is a structure achieving to a set of things in which some doubles of matters are in some items "related". Subjects correspond to mathematical defalcations called vertices (also called points or nodes) and each of related to double of vertices is called an edge (also called an line or arc)(Trudeau 1993). Usually, a graph is contacted in diagrammatic form as a set of points for vertices, joined by lines or arcs for edges.

In other words, a graph is defined as a set of vertices and edges. Edges connect one of the vertex to the other vertices. These vertices are considered as points or nodes in the graph. Edges are considered as lines or arcs. There are many problems associated with graph theory, as it has been seen in the past, that graph is better technique for showing a problem. The theory of graphs is a branch of mathematics that discusses graphs(Guichard 2017, Mondal and De 2017). This topic is in fact a branch of a topology that is closely bound up with algebra and matrix theory. Graph theory, unlike other branches of mathematics, has a definite start point, and is the publication of an essay by Swiss mathematician Leonard Euler to solve the Königsberg bridge problem in۱۷۳۶(Grimaldi 2000). Recent advances in mathematics, especially in its applications, have greatly expanded to theory of graph, so that the graph theory is now a very good tool for research in various fields such as coding theory, operation research, statistics, electrical networks, computer science, chemistry, biology, Nanoscale Network Materials, Social sciences and other fields (Agnarsson and Greenlaw 2007, Deo 2017, Kaveh and Ghazaan 2017, Kamath and Pai 2018, Trinajstic 2018, Vecchio, Mahler et al. 2021). In graph theory, GCP is a special feature of the graph labeling problem (Kokosinski and Ochał , Menaka and Arthy , Jensen and Toft 2011, Goudet, Grelier et al. 2021). An overall approach is to paint edges or vertices and surface of graph, with number of these colors at least. In the simplest case, coloring is desired in which two vertices adjacent to the same colors do not (Vertex coloring)(Morris and Song 2021). Additionally, the color of edges is defined in a same way as two edges in same joint are not same color. Also, surface color of the graph is surface of graph, which does not have two adjacent levels of same colors. As it has been said, the GCP can occur in three different ways, but main purpose of this paper is to vertices coloring. GCP has many applications in many theoretical and practical fields. In addition to classical problems defined in this field, due to various constraints on type of graphs, method of coloring and even a number and color of graph elements of problem are defined and resolved with extensive programs in the industry and science. To be despite the fact that this is still a growing science. Due to its extensive use in areas such as communication and scheduling, much attention has been paid. Due to the variety of graphs and the discrete GCP, an NP-Hard problem is optimized (Mahmoudi and Lotfi 2015, Zhou, Fang et al. 2017, Alves, Nascimento et al. 2021). Many practical problems can be represented by graphs that can be used to model different types of relations and process dynamics in physical, social and information systems. In mathematics, graphs are useful in geometry and certain parts of topology. Algebraic graph theory has close links with group theory(Biggs, Biggs et al. 1993). In computer science (Deo 2017), graphs are used to represent networks of communication (Evans 2017), data organization (Panda, Dutt et al. 1997), computational devices (Rao 1993), flow of computation, etc. Graph theory is also used to study molecules in chemistry and physic condensed matter physics(Schmidt, Guenther et al. 2016), three dimensional structure of complicated simulated atomic structures can be studied quantitatively by gathering statistics on graph - theoretic properties related to topology of atoms. Comprehensive researches are doing to solve the GCP. Many methods and algorithms have used to solve GCP(Mostafaie, Khiyabani et al. 2020, Postigo, Soto-Begazo et al. 2021), including, accurate, approximation algorithms, heuristic and meta-heuristic algorithms(Xu and Chen 2021). The Whale Optimization Algorithm (WOA) is a powerful meta-heuristic algorithm inspired by behavior and method of nourishing whales (Mirjalili and Lewis 2016). It has mostly been used in many real-world applications and NP-Hard issues, but has not been used to solve the GCP. In this paper, we use the Fuzzy Whale Optimization Algorithm (FWOA) based on fuzzy logic to solve GCP. The FWOA algorithm performs this by changing sensitivity of random-to-fuzzy parameters. Then, we compare obtained results of proposed method in terms of runtime and convergence in comparison with other meta-heuristic algorithms. Briefly, main objectives of this paper are as follows:

- Providing a new meta-heuristic algorithm in discrete space for GCP.
- Improving runtime of algorithm in comparison with other meta-heuristic algorithms.
- Improving the convergence of algorithm in comparison with other meta-heuristic algorithms.

The structure of this paper is organized as follows. The related work is analyzed and reviewed in section 2. Proposed method with all definitions, assumptions and details with three titles is presented in section 3. Experiments and results analysis are given in section 4. Finally, section 5 summarizes our conclusion and future works.

RELATED WORKS

Chiarandini and Stützle (have presented computational experience on algorithms for coloring large, general graphs. They considered 1260 graphs which were built by controlling a few structural parameters so it is similar to pick up a superior insight into relationship between algorithm implementation and graph features. They considered graph size, edge density, sort of graph, and attributes of color classes. They suggested that if described features are fully examined, the algorithm would be used on a graph with 1000 vertices. Although they have studied a good algorithm, they have not studied meta-heuristic algorithms and do not compare these algorithms.

Douiri and Elbernoussi (2015) have interested in an elaboration of an approximate solution for the Minimum Sum Coloring Problem (MSCP), and they have tried to give a lower bound for MSCP by searching for a decomposition of the graph predicated on the meta-heuristic of Ant Colony Optimization (ACO). They have tested different instances to validate the approach. They also have proposed a contribution concerning the study of the lower bound of the MSCP. For this, they emphasized coloring of the complement graph. An ACO is also adapted to this problem for obtaining a clique's decomposition of the original graph. This clique decomposition offers a better lower bound for MSCP over on other graph decompositions. As a result, benefits of this paper include high convergence, improvement of lower bounds. However, number of high iteration of algorithms and some unsuccessful performances are main disadvantage of this paper.

Tomar, Singh et al. (have performed using an Artificial Bee Colony (ABC) optimization algorithm for GCP. GCP with challenge of coloring nodes of each graph with the smallest possible number of while confirming at two nearby nodes do not receive an identical color and optimizing the new ABC algorithm for GCP. In this paper, they compared the proposed algorithm with three other coloring algorithms, namely, prioritization, instructions based on a large-scale and degree order based on saturation. Furthermore, these results show that the ABC converges in several repetitions, and it can optimize colors to the vertices of a graph. However, it has a high runtime, and it runs on small-scale graphs.

Mahmoudi and Lotfi (2015) have proposed a new method for solving GCP based on Cuckoo Optimization Algorithm (COA). The COA algorithm introduced brilliant abilities such as high convergence rates and a global optimal response. Since COA was originally presented to solve continuous optimization problems, this paper uses the COA for coloring problem, so it is necessary to use the COA for discrete issues. Therefore, to apply COA to a discrete search space, standard math operators such as adding, subtracting and multiplication in the COA migration operator are redefined based on lack for space in discrete space. As a result, success rate is approximately 100% and a good balance between diversity and concentration of paper benefits. Falling in local optimal and high execution time of the algorithm is one of disadvantages of the paper.

Astuti (2015) have provided an execution examination with the data hiding scheme in light of GCP utilizing two evolutionary algorithms (EA), to be specific Genetic Algorithm (GA) and Article Swarm Optimization (PSO). Both PSO and GA as an evolutionary algorithm can be utilized to solve GCP on proposed conspire. PSO and GA in data hiding pattern plan give around same execution in concealing limit. In light of investigation of testing comes about, it can be reasoned that GCP utilizing an EA can bolster a data hiding scheme at medical image. Regarding computational time, GA plot always gives preferred execution over PSO conspire, which is utilized by a unique plan. In this way, they prescribe the utilization of GA in the data hiding plotted to be executed on cell phones. In subsequent investigation, they supply an execution examination with data hiding scheme in light of GCP utilizing two EAs, in particular, GA and PSO. Application of a proposed algorithm in real work, its suggestion for mobile use, and low runtimes of an algorithm are advantages of article. However, algorithm should be improved and should not be implemented for small-scale graphs**.**

Bensouyad and Saidouni (have introduced a discrete flower pollination algorithm for solving GCP combinations. As shown, proposed approach converted continuous values digitally using a round function. Then, using strategy of exchange, the algorithm improves. Such a strategy is useful to ensure diversification and to prevent any recession that may occur during to search process. Based on experimental results, they showed that the algorithm is an effective algorithm, while results are similar to less complexity compared to other meta-cognitive techniques. The algorithm's performance and computational results show promising results. Low complexity of the algorithm, high efficiency of the algorithm and achievement of similar results are advantages of using this algorithm. In the future, they will look at why a mutual point arises and try to use a method in a real world program.

Hsu, Horng et al. (2011) have presented a Modified Turbulent PSO (MTPSO) show to solve planar GCP in light of PSO. The presented model is comprising of the turbulent strategy, assessment scheme, and walking one strategy. The presented MTPSO model can solve planar GCP utilizing four- colors more effectively and precisely. Contrasted with outcomes appeared in Cui, Qin et al. (2008), not just experimental results of proposed model can get smaller average iterations yet can acquire higher adjustment coloring rate when number of nodes is beyond 30. Finally, we can say that the algorithm is implemented on small graphs with high runtime.

Han and Han (have proposed a Bi-objective GA (BGA) model for the GCP. Based on this new model, a bi-objective GA is proposed which employs novel crossover and mutation operator as the genetic operators. Convergence to global optimal set of BEA is proved. Simulation results demonstrate that the new algorithm is effective. Most important of all, BGA can provide several dominated solutions for decider by only one run. For an over-constrained GCP, the BGA is more prospected. Experimental results demonstrate. Given that proposed algorithm runs on small-scale graphs. The algorithm results are the best answer for each repetition with low runtime.

Chen and Kanoh (have proposed a discrete Firefly algorithm (FA) based on similarity and used to solve 3-coloring problems. Although the FA hybrid for GCP has been proposed (Fister Jr, Yang et al. 2012), the proposed approach with the discrete ABS, PSO has been compared to evaluate its performance. The original FA solves problems of continuous optimization. To apply it to discrete issues, they have defined original space again in a discrete space, and they call it the Discrete Firefly Algorithm (DFA). Experiment on 100 randomly generated graphs shows that proposed method gives good success, even if the graph size is very large.to solve flattening problems improves success rate of discrete PSO and ABC. In addition, because proposed method fails to use a original algorithm of FA directly from any other combination technique, it can be applied to large-scale Combinatorial Optimization Problems (COP). However, disadvantage of proposed method is also clear. Before finding an optimal answer, it will spend more time with its rivals. Therefore, they have not claimed that proposed method is the best way to solve graph coloring problems. Three things will be done in the future. At first step, problem of slow convergence will be studied in depth. Second, Hamming distance and similarities are used to measure difference between firefly creams in this work. Other methods will test distance and compare their performance. Finally, this method will be applied to other COPs and its performance will be tested.

PROPOSED METHOD

In order to improve solving the GCP problem, we use the FWOA algorithm. In this section, at first, problem definition is briefly introduced. On following, fuzzification section is described, and then the FWOA algorithm is proposed for The GCP. Finally, a flowchart of proposed method is shown in Fig. 1.

Problem definition

Let $G = (V, E)$ an indirect graph with a vertex set *V* and an edge set *E*. A subset of G is called an independent set if no two adjacent vertices belong to it. A k-coloring of G is a partition of V into k independent sets (called proper color classes). A graph is m-colorable if and only if it can be colored using m colors. GCP is a well-known combinatory problem, and important work in solving many real problems. An optimal coloring of *G* is a *k*-coloring with the smallest possible *k* (the chromatic number $\chi(G)\ of\ G$). The graph coloring problem is to find an optimal coloring for a given graph G. Formally, a m-coloring will be illustrated by a set $H = \{C(v_1), C(v_2),\ldots,C(v_n)\}$ such as $C(v_i)$ is color determined to the vertex v_i. If for all $\{u,v\}\in E$, $C(u) \neq C(v)$, then *H* is a feasible m coloring; otherwise, *H* is an unfeasible m-coloring. In optimization issue of the GCP, main purpose is to minimize whole number of colors used to color an assumed graph (That means $\chi(G)$ reaches its minimum value).

Formally, the GCP can be formulized as follows: According a m-coloring $H = \{C(v_1), C(v_2),\ldots,C(v_n)\}$ with the set $V = \{v_1, \ldots, v_n\}$ of vertices, the assessment function f counts the number of contradictory vertices created by *H* such that (Bensouyad and Saidouni , Djeloul, Layeb et al. 2015):

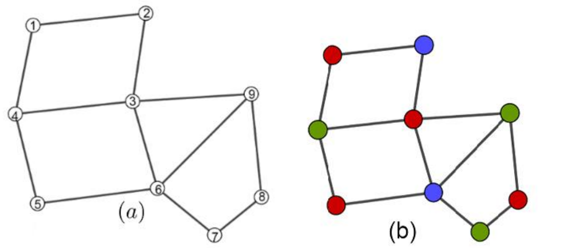
$$f(H) = \sum_{\{u,v\}\in E} \rho_{uv} \qquad (1)$$

Where:

$$\rho_{uv} = \begin{cases} 1, & if\ C(u) = C(v) \\ 0, & otherwise \end{cases} \qquad (2)$$

As a result, a coloring H with $f(H) = 0$ communicates to a feasible m-coloring.

As an example of such representation, we consider here connected graph of Fig. 3.a characterized by the set of vertices {1, 2, 3, 4, 5, 6, 7, 8, 9}. One acceptable coloring of such graph is shown in Fig. 3.b where the set of colors is {1: Red, 2: Blue, 3: Green} and its representation is shown in Fig. 1.c.



1	2	1	3	1	3	2	1	2

(c)

Fig. 1. (*a*) Uncolored graph, (b) Legal coloring using 3 colors, (c) Coloring deputation

Fuzzification

In algorithms, in order to get closer to reality and to consider uncertainty for some input parameters, parameters are considered as a trapezoidal fuzzy number, which shows following form of membership function of fuzzy parameters:

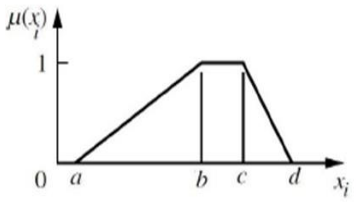


Fig. 2. The membership function of fuzzy parameters

For a trapezoidal fuzzy number, a membership function will be:

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \frac{c-x}{c-b}, & c < x < d \\ 0, & x \ge d \ or \ x < a \end{cases} \quad (3)$$

There are several ways to create fuzzy numbers. Another common method for fuzzy number formation is use of fuzzy mechanisms that define a relation for their membership functions, taking into account specific definitions and a data series. Different types of fuzzy instruments have different characteristics, depending on type of input or type of data, the fuzzy output numbers are required, each with a specific capability. In whale optimization algorithm, three parameters r_1, r_2 and p were generated with normal random distribution in interval (0.1). We consider each of these parameters as fuzzy and in form of a trapezoidal fuzzy number, and we obtain values of each of these parameters in an innovative manner. This approach is called Fuzzy of Whale Optimization Algorithm (FWOA). For example, for a fuzzy number ($\tilde{r}_1$):

$$\tilde{r}_1 = (b-a) \times rand + a \quad (4)$$

Where rand is a random number with normal distribution in range (0, 1) and values of a, b are calculated as follows:

We first select correct integer in the interval [1, 4].

- If number of the first step is equal to 1, then ($a = 0$, $b = 0.25$). Obviously, by placing a, b in relation (4), the value of $\tilde{r}_1$ is definitely in range (0.0.25).
- If number of the first step is 2, in this case, ($a = 0.25$, $b = 0.50$). According to preceding paragraph, it is evident that by placing a, b in relation (11), value of $\tilde{r}_1$ is definitely in range (0.25.0.50).
- If the number of the first step is 3, in this case, ($a = 0.50$, $b = 0.75$). Placing a, b in relation (4), value of $\tilde{r}_1$is definitely in range (0.50.0.75).
- Finally, if chosen number is 4, in this case, ($a = 0.75$, $b = 1$). And by placing a, b in relation (4), value of $\tilde{r}_1$ is definitely in range (0.75.1).

For fuzzy numbers ($\tilde{r}_2$) and p, we also apply the fuzzy number ($\tilde{r}_1$). By doing so, in practice, convergence occurs in the algorithm earlier, because with this, it is chosen from entire range (0, 1) with a numerical probability, which is advantage of using a trapezoidal fuzzy number.

The FWOA algorithm for GCP

The search space on the GCP of a graph is a discrete space, meaning that components of each population of a population can't have any arbitrary value, and permitted values are limited to natural numbers 1 to N. So, to solve the graph coloring problem, we deal with a discrete FWOA. In the standard FWOA algorithm, solutions are updated in search space directions successive-valued locations. However, in proposed discrete FWOA algorithm, search space is modelled as N-dimensional integer values where each value shows a color number. More precisely, proposed algorithm begins by initializing the number of colors k then, it constructs initial whales in population where each whale is a vector of integer values representing color assigned to each vertex in the graph. After applying a shrink updating or spiral updating, a vector of continuous values plots a series of integers using a simple round operation. After that, we apply a exchange strategy. At this stage, color of the most conflicting node will be exchanged with color of another random vertex which appertains to the smallest sized class i.e. This class contains vertices receiving color with minimum occurrence in whale.

Representation

In our implementation, we have applied an integer deputation design. In the other words, an solitary is a thorough assignment of k colors to graph vertices such that H = {C(v_1), C(v_2),…,C(v_n)} where C(v_i) indicates color of vertex i. More precisely, an singular is specified by a one-dimensional array with n elements as showed in Fig.1(c) such that C(v_i) is color determined to the vertex i. In order to apply easily FWOA principles on GCP, we need to map the graph coloring solutions into a FWOA representation that could be easily manipulated by FWOA operators. The graph coloring solution is represented as Coloring deputation (Fig.1(c)). In order to solve the GCP, position of each population in the

FWOA can be considered as a string of available colors, which are assigned in advance to the vertices of the graph, respectively. For example, if individual position in the nine-dimensional space to solve the GCP of graph, with nine vertices in form of (Fig.1(c)), states that the first, third, fifth and eighth vertices of graph have a color 1, and the second, seventh and ninth vertices have a color 2 And the fourth and sixth vertices are colored 3. So position vector is actually a N-dimensional vector, where N is number of vertices of graph.

Fitness Function

To define a fitness function, we need a function that states whether color assignment to the graph vertices is limited to the GCP. That is, do two adjacent vertices have different colors in the color assignment? This function returns value of conflict, which expresses number of vertices that have same colors. Since position of each whale is same as colors attributed to vertices and individuality of each person can be negative due to updating relations of (5) and (6), it is necessary to have a function that always corresponds to position and create assigned colors.

Let *A(G)* be (0,1) adjacency matrix (or neighborhood) matrix of a graph *G = (V, E)* where (a_{ij}) defined as follows (Djeloul, Layeb et al. 2015):

$$a_{ij} = \begin{cases} 1 & if\ (v_i, v_j) \in E \\ 0 & otherwise \end{cases} \quad (5)$$

Let conflicting matrix *conflict* of a coloring C is given by:

$$conflict\ t_{ij} = \begin{cases} 1 & if\ \mathrm{C}(v_i) = \mathrm{C}(v_j)\ and\ a_{ij} = 1 \\ 0 & otherwise \end{cases} \quad (6)$$

For a solution H, fitness function *f* (H) is given by Eq. (7).

$$f(H) = \sum_{i=1}^{n} \sum_{j=1}^{n} conflict\ t_{ij} \quad (7)$$

Purpose is then to minimize number of conflicts until acceding $f^*(H) = 0$, for a fixed k. Thus a correct coloring is found.

Exchange Strategy

Mostly strategy is used when cleavage with discrete problems. Basically, it selects randomly two vertices and simply exchanges their colors (see Fig. 3). Originally, exchange strategy helps to hold variety in population and then eschew slump problem that may turn up in research flow.

1	2	1	3	4	3	2	1	5
1	2	1	**5**	4	3	2	1	**3**

Fig. 3. Exchange strategy

This strategy could be well surveyed to minimize conflicts if chosen vertex to exchange is the most conflicting one in solution H.

Algorithm

First we create a primitive population of whales. Then, we consider K number of colors to the GCP. We assume each whale is a solution to problem. That is, each whale will have one N array as follows:

1	2	3	4	…	N
C(v_1)	C(v_2)	C(v_3)	C(v_4)	...	C(v_k)

Where C(v_i) indicates color of vertex i (i=1,2,3,…k) and N shows the number of graph nodes.

Each whale is considered as an answer to problem. For each whale we define a cost and a position. After we have defined each whale in its position, then we rate suitability or match of each whale that we use to evaluate conflict function as expressed in Eq. 6.

As the last step in the first part, we find the best public answer. The best answer is an answer, which is less than its worth. That is, answer is that number adjacent colored vertices less than rest of answers. An answers that has the least number of violations in constraint the GCP problem. Once search agent is better defined, other agents (whales) will try to update their position to the best answer. Agents move towards whales that have the lowest value for the HHHH function. Given that, humpbacked whale attack prey based on the bubble strategy. This method is done in two ways: a Shrinking and a spiral.

The Shrinking mechanism is formulated and updated as follows.

$$\vec{D} = \left| \vec{C}.\vec{X}^*(t) - \vec{X}(t) \right| \quad (8)$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A}.\vec{D} \quad (9)$$

Where in formulas, current iteration, $\vec{A}, \vec{C}$, coefficients of vectors, the X^* vector of the best answer so far, represent position vector, $\vec{X}$ is position vector. It should be noted that position vector of the best solution should be updated in each replication if a better solution than previous one is found. According to Eq. 9 search agents (whales) update their positions according to position of the best known solution (prey). The adjustment of values of *A* and *C* vectors control areas where a whale can be located on neighborhood of prey. Vectors $\vec{A}$ and $\vec{C}$ are obtained using Eq. 10 and Eq. 11.

$$\vec{A} = 2.\vec{a}.\vec{r_1} - \vec{a} \quad (10)$$

$$\vec{C} = 2.\vec{r_2} \quad (11)$$

The vector $\vec{a}$ decreases linearly from 2 to 0 over course of iterations (in both exploration and exploitation phases), and $\vec{r_1}, \vec{r_2}$ are the fuzzy vectors defined in Eq. (4).

collapse behavior is obtained by decreasing value of a in Eq. (3) according to Eq. 12.

$$a = 2 - t\frac{2}{\text{MaxIter}} \quad (12)$$

Where t is the number of repetitions and MaxIter has the maximum number of allowed duplicates.

The spiral mechanism is formulated and updated as follows.

In this method to simulate spiral-shaped path, first distance between vales located at position (y, x) and bait located at (y*, x*) is computed.

$$\vec{D}' = \left|\vec{X}^*(t) - \vec{X}(t)\right| \quad (13)$$

Then a spiral equation is used to create a neighboring position same as in the Eq. (14).

$$\vec{x}(t+1) = e^{bk}.\cos(2\pi k).\vec{D}' + \vec{X}^*(\text{t}) \quad (14)$$

Where b is a constant to define the shape of a logarithmic spiral, k is a random value in [-1, 1].

It should be noted that humpback whale swims along a curved circle along a spiral path around their hunt. To simultaneously simulate this behavior and contraction blockage mechanism, it is assumed that probability of occurrence of position with each of mechanisms during optimization process is 50%. A mathematical model of this action is expressed by Eq. (15).

$$\vec{X}(\text{t}+1) = \begin{cases} \vec{X}^*(t) - \vec{A}.\vec{D} & if\ \ P < 0.5 \\ e^{bk}.\cos(2\pi k).\vec{D}' + \vec{X}^*(t) & if\ \ p \geq 0.5 \end{cases} \quad (15)$$

Where p is a random number in [0,1].

A similar method based on variations of vector $\vec{A}$ can be used for this case. In fact, humpback whales perform random search according to position of each other. Therefore, we use vector A with a random value greater than 1 or less than -1 to force search agent to go away from reference whale. Therefore, we use vector A with a random value greater than 1 or less to force search agent to go away from reference whale. In contrast to exploitation phase, we are updating status of search agent based on the random selection of a search agent instead of selecting better agent. This mechanism and the value $|\vec{A}|> 1$ emphasize the discovery phase and allow whale algorithm to perform a general search. This mechanism can be mathematically modeled as in Eq. 16 and Eq. 17.

$$\vec{D}'' = \left|\vec{C}.\vec{X}_{rand} - \vec{X}(t)\right| \quad (16)$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A}.\vec{D}'' \quad (17)$$

The $\vec{X}_{rand}$ is a random position vector (a random whale) chosen of a current population. After that situation has been updated. In each situation, following solutions are found for problem.

1	2	3	4	…	N
C(v_1)	C(v_4)	C(v_3)	C(v_2)	…	C(v_k)

Determine worthiness of updated answers. One of the best solutions is to replace previous answer. Algorithm continues until a convergence condition is satisfied or total number of repetitions is completed.

A flowchart of a proposed method is shown in Fig. 4.

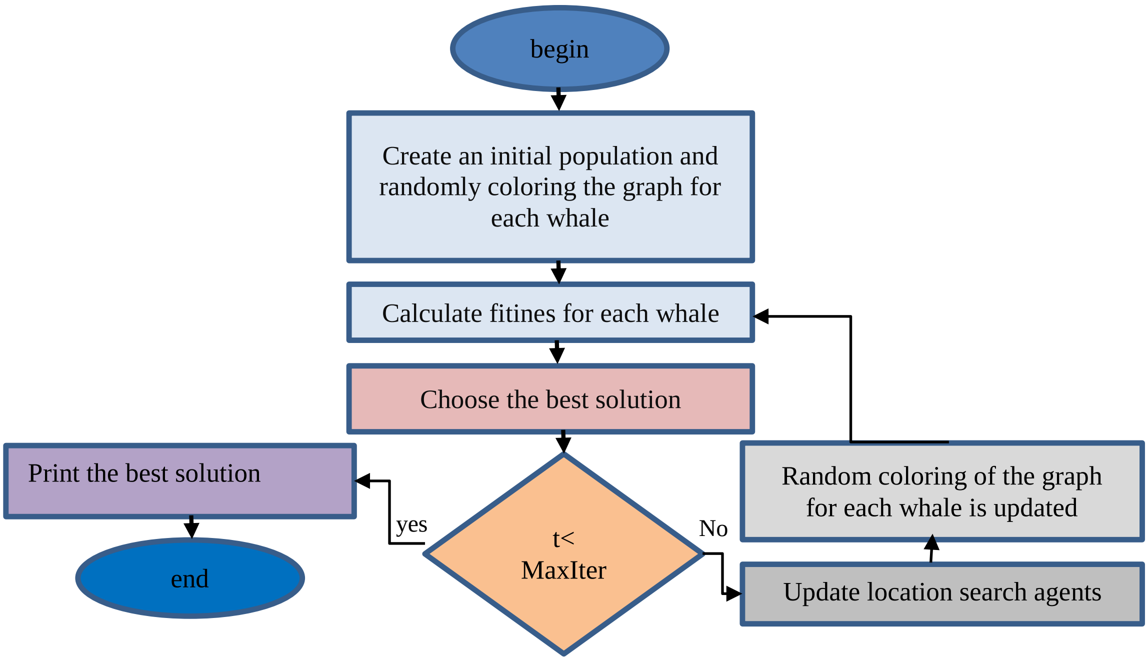


Fig.4. Flowchart of the proposed method

EXPERIMENT AND RESULTS ANALYSIS

FWOA for graph coloring is implemented in Matlab 10.9 with AMD FX-9830P RADEON R7, 12COMPUT CORES 4C+8G processor and 16 GB of memory and 64-bit Operating System. To estimate efficiency of our technique, a set of standard DIMACS benchmark (http://mat.gsia.cmu.edu/COLOR/instances.html) have been applied. We conduct a comparison study with other well-known meta-heuristic algorithm optimization technique in order to show supremacy and effectiveness of proposed FWOA algorithm contrasted to other methods.

Parameter settings

The basic parameters of the FWOA algorithm appropriated for the GCP are set as follows. For each specimen, we simulated 15 runs, maximal number of iterations is set to 100 and the number of is whales 100. The number of colors (k) is defined for each sample corresponding to that specimen. Parameters r and p are defined as fuzzy, as defined in Section 3.2.

Datasets

In order to specify efficiency of proposed approach, experiments are enforced on 18 benchmark graphs from the DIMACS graph reservoir (Galinier and Hertz 2006, Abbasian and Mouhoub 2013, Douiri and Elbernoussi 2015) . Table 1 shows details of used graphs such as number of nodes, number of edges and best known $\chi(G)$ in each graph.

Results and discussion

Execution of proposed FWOA algorithm is tested on 20 models. In 20 models of graph, optimal solution in 15 times will be obtained and results of execution includes digressions that in related Table 2 shown in Fig. 5, we show success rate of these specimens in terms of percent. In effect, the success rate; number of times that the algorithm is obtained a correct coloring of graph is counted. As it can be seen from Fig. 5, solving classes of graphs DSJ, queen 7-7and a few other graphs seems hard. Algorithms with success rates upper 70% have textually been stated in identity with color value and in most cases, success rate is caulk to 100%.

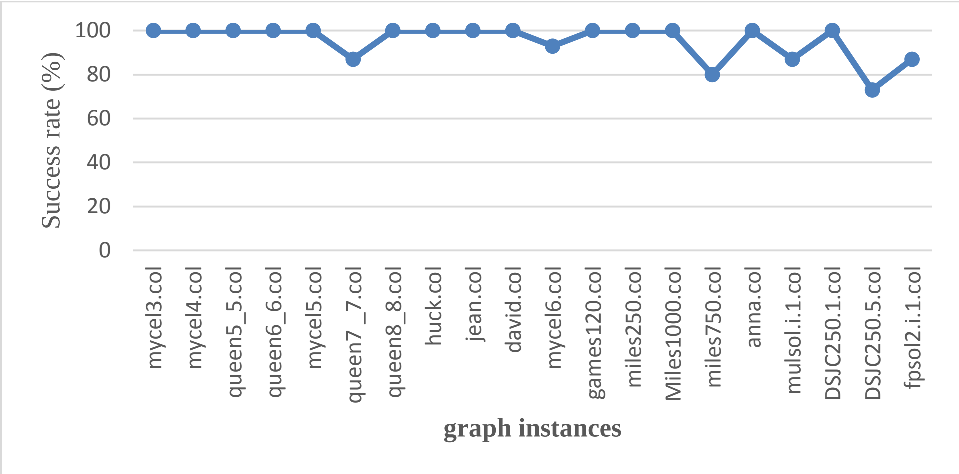


Fig. 5. The success rate in terms of percentage

In Table 2, results acquired from fulfilment the proposed algorithm FWOA with results of algorithms MCOACOL(Mahmoudi and Lotfi 2015), ABAC (Bui, Nguyen et al. 2008), BEECOL(Faraji and Javadi 2011) , GA(Hindi and Yampolskiy) and PGA (Abbasian and Mouhoub) are contrasted. At the first column shows the name of the graph, the second column shows number of vertices, the third column shows number of edges, best known $\chi(G)$ of a graph is shown in the fourth column and next columns indicates the $\chi(G)$ results of the six algorithms are contrasted. As seen at the table, based on number of vertices of graph are classified.

Table 1: Comparison of the FWOA algorithm with MCOACOL, ABAC, BEECOL, GA and the PGA algorithm.

Problem instances				Results of algorithms						
Graph name	|V|	|E|	Best known	FWOA	Success rate (FWOA)	MCOACOL (Mahmoudi and Lotfi 2015)	ABAC (Bui, Nguyen et al. 2008)	BEECOL (Faraji and Javadi 2011)	GA (Hindi and Yampolskiy)	PGA (Abbasian and Mouhoub)
mycel3.col	11	20	4	3	$^{15}/_{15}$	4	4	4	4	4
mycel4.col	23	71	5	5	$^{15}/_{15}$	5	5	5	5	5
queen5_5.col	25	160	5	6	$^{15}/_{15}$	5	5	5	5	5
queen6_6.col	36	290	7	8	$^{15}/_{15}$	8	7	8	8	8
mycel5.col	47	236	6	6	$^{15}/_{15}$	6	6	6	6	6
queen7 _7.col	49	476	7	7	$^{13}/_{15}$	7	7	8	8	8
queen8_8.col	64	728	9	9	$^{15}/_{15}$	10	9	10	-	-
Huck .col	74	301	11	11	$^{15}/_{15}$	11	11	11	11	11
Jean .col	80	254	10	10	$^{15}/_{15}$	10	10	10	10	10
David .col	87	406	11	11	$^{15}/_{15}$	11	11	11	11	11
mycel6.col	95	755	7	8	$^{14}/_{15}$	7	7	7	-	-
games120.col	120	638	9	9	$^{15}/_{15}$	9	9	9	9	9
miles250.col	128	387	8	8	$^{15}/_{15}$	8	8	8	8	8
Miles1000.col	128	3216	42	43	$^{15}/_{15}$	42	42	42	42	42
miles750.col	128	2113	31	32	$^{12}/_{15}$	31	31	31	-	-
Anna .col	138	493	11	11	$^{15}/_{15}$	11	11	11	11	11
mulsol.i.1.col	197	3925	49	49	$^{13}/_{15}$	49	49	49	-	-
DSJC250.1.col	250	6436	8	8	$^{11}/_{15}$	10	9	9	-	-
DSJC250.5.col	250	31366	28	29	$^{13}/_{15}$	33	30	30	-	-
fpsol2.i.1.col	496	11654	65	66	$^{11}/_{15}$	65	65	65	65	65

Performance of FWOA proposed algorithm with MCOACOL, ABAC, BEECOL, GA and PGA algorithms have been compared on 20 subjects of graph. As it can be seen from Table 1, FWOA algorithms as five algorithms that have been compared, easily and with punctuality and high speed are able to find known χ(G) of the graph.

In 9 cases: (mycel4.col; mycel5.col; Huck.col; Jean.col; Dvid.col; games120.col; miles250.col; Anna.col and mulsol.i.1.col) known χ(G) of our proposed algorithm is equal to known χ(G) of other compared algorithms. In 4 cases: (queen6_6.col; queen7 _7.col; queen8_8.col and DSJC250.1.col) known χ(G) of our proposed algorithm is equal to known χ(G) of some algorithms in comparison with some algorithms different are 1.

In 5 cases: (queen5_5.col; mycel6.col; Miles1000.col; miles750.col and fpsol2.i.1.col) known X(G) of our proposed algorithm has1 more difference than known X(G) of other compared algorithms. In the mycel3.col sample, known X(G) of our proposed algorithm has 1 unit less than the known χ(G) of other compared algorithms. Even 1 unit is less than the best known X(G). That means, 1 unit has been improved. In the DSJC250.5.col sample, known X(G) of our proposed algorithm have 1 different with two ABAC and BEECOL algorithms and have 4 different with MCOACOL algorithm. We can say that the X(G) of the DSJC250.5.col sample is improved by our proposed algorithm compared to compared algorithms.

Generally, in five samples of graph our proposed algorithm in finding chromatic number, obtains better result than other compared algorithms. In eleven samples of graph, known XG) of our proposed algorithm is equal to known X(G) of the most compared algorithms. In four samples of graph our proposed algorithm in finding chromatic number, obtains worse results than other compared algorithms. From a perspective, this comparison and analysis can be seen in Fig. 6.

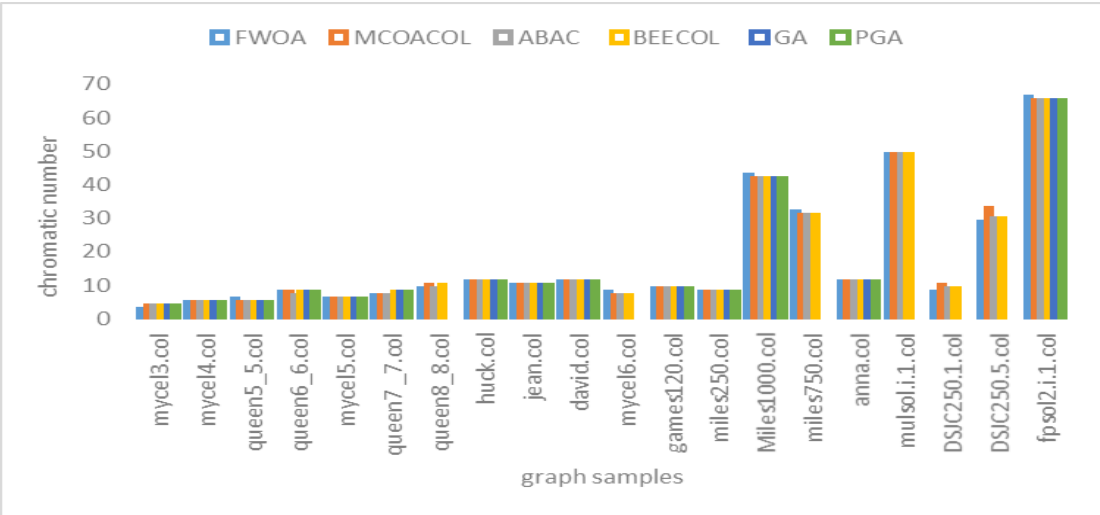


Fig. 6. Comparison of FWOA algorithm with compared algorithms

Algorithms FWOA reliability

To certify accuracy of proficiency of FWOA algorithm and illustrate its capabilities in solving GCPs, this section compares proposed algorithms in solving GCPs. We present experimental results obtained by FWOA and compares with algorithms SAGCP (Pal, Ray et al. 2012), MSAGCP (Pal, Ray et al. 2012). We compare the runtime and the best known chromatic number for each algorithm on the standard graph. As it can be seen from Table 2, FWOA algorithm in 12 samples of graph is compared to the SAGCP and MSAGCP algorithms.

Table 2: Comparison of FWOA algorithm with SAGCP and MSAGCP algorithms.

Problem instances				Results of algorithms					
Graph name	|V|	|E|	Best known	FWOA	Time of FWOA (S)	SAGCP (Pal, Ray et al. 2012)	Time of SAGCP (S)	MSAGCP (Pal, Ray et al. 2012)	Time of MSAGCP(S)
queen5_5.col	25	160	5	6	32	7	186	5	40
queen7 _7.col	49	476	7	7	59	9	276	7	188
Huck .col	74	301	11	11	40	11	1295	11	31
Jean .col	80	254	10	10	43	11	520	10	45
David .col	87	406	11	11	63	11	10672	11	2972
mycel6.col	95	755	7	8	124	8	12436	7	6683
games120.col	120	638	9	9	140	10	2874	9	122
DSJC125.1.col	125	736	?/5	6	148	10	1094	6	201
Miles1000.col	128	3216	42	43	873	50	25084	45	7702
miles750.col	128	2113	31	31	543	35	12767	31	2437
Anna. col	138	493	11	11	185	13	9855	11	2535
mulsol.i.1.col	197	3925	49	49	912	58	10439	52	2861

As it can be observed, FWOA algorithm in comparison with SAGCP algorithm in all samples gives better response and in comparison with MSAGCP algorithm for instance queen5_5.col gives worse response. So our technique has arrived a better answer. Our proposed algorithm in collation with MSAGCP algorithm and SAGCP for mulsol.i.1.col is results in chromatic number respectively in 3 and 9 different. Also, for Miles1000.col is results in chromatic number respectively in 2 and 8 different. Our algorithm has a very low runtime compared to SAGCP and MSAGCP algorithms compared to runtime of the algorithm. Figures 7 and 8 show these differences well. Figure 7 compares runtime of algorithms. Figure 8 compares the best known $\chi(G)$ algorithms.

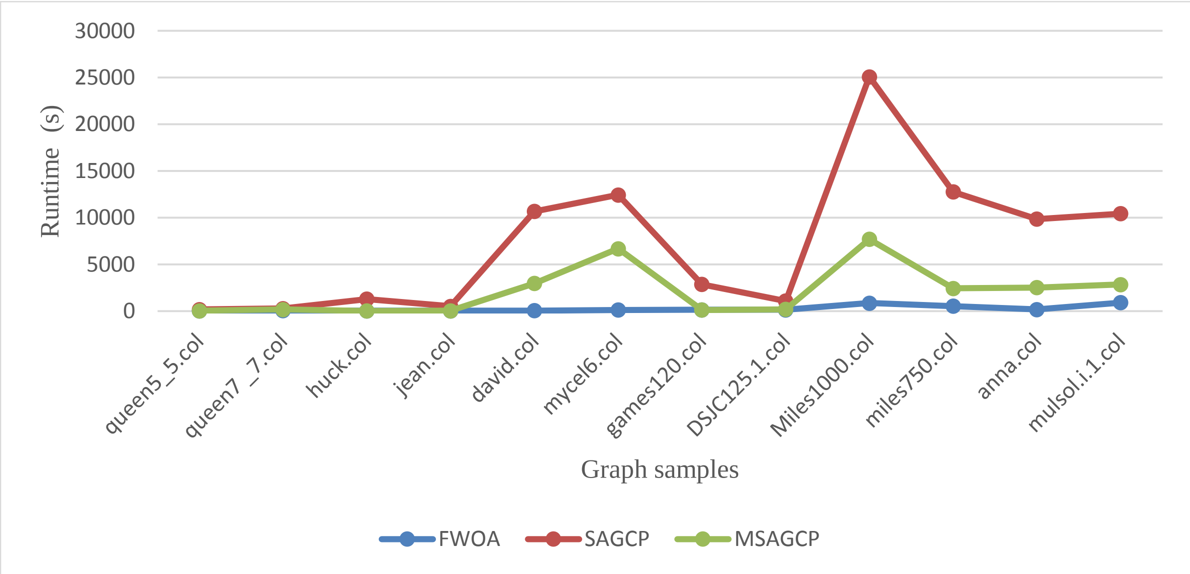


Fig. 7. Compare the runtime in of studied algorithms on sample graphs

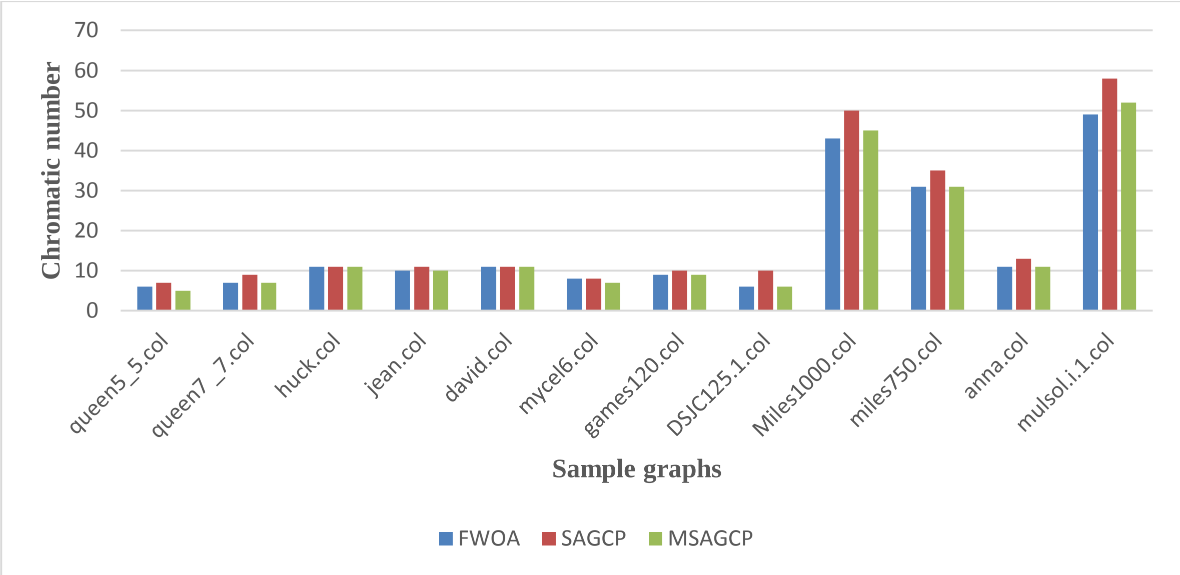
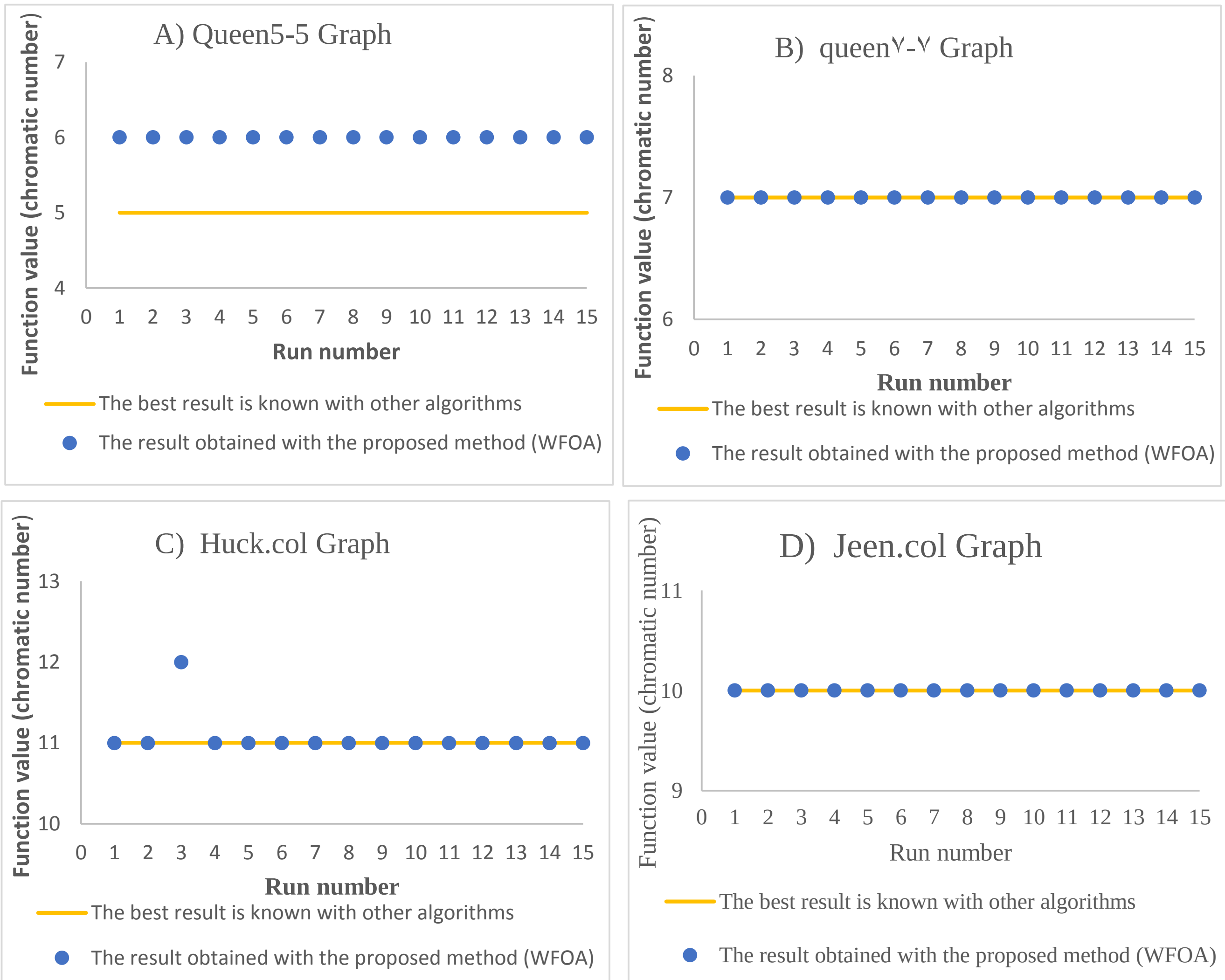


Fig. 8. Compare chromatic number in of studied algorithms on sample graphs

FWOA algorithm stability

To represent stability of algorithm, stability scheme for each chromatic numbe racquired in 15 times is drawn and Fig. 9 illustrates stability of algorithm on 10 graphs as nether.



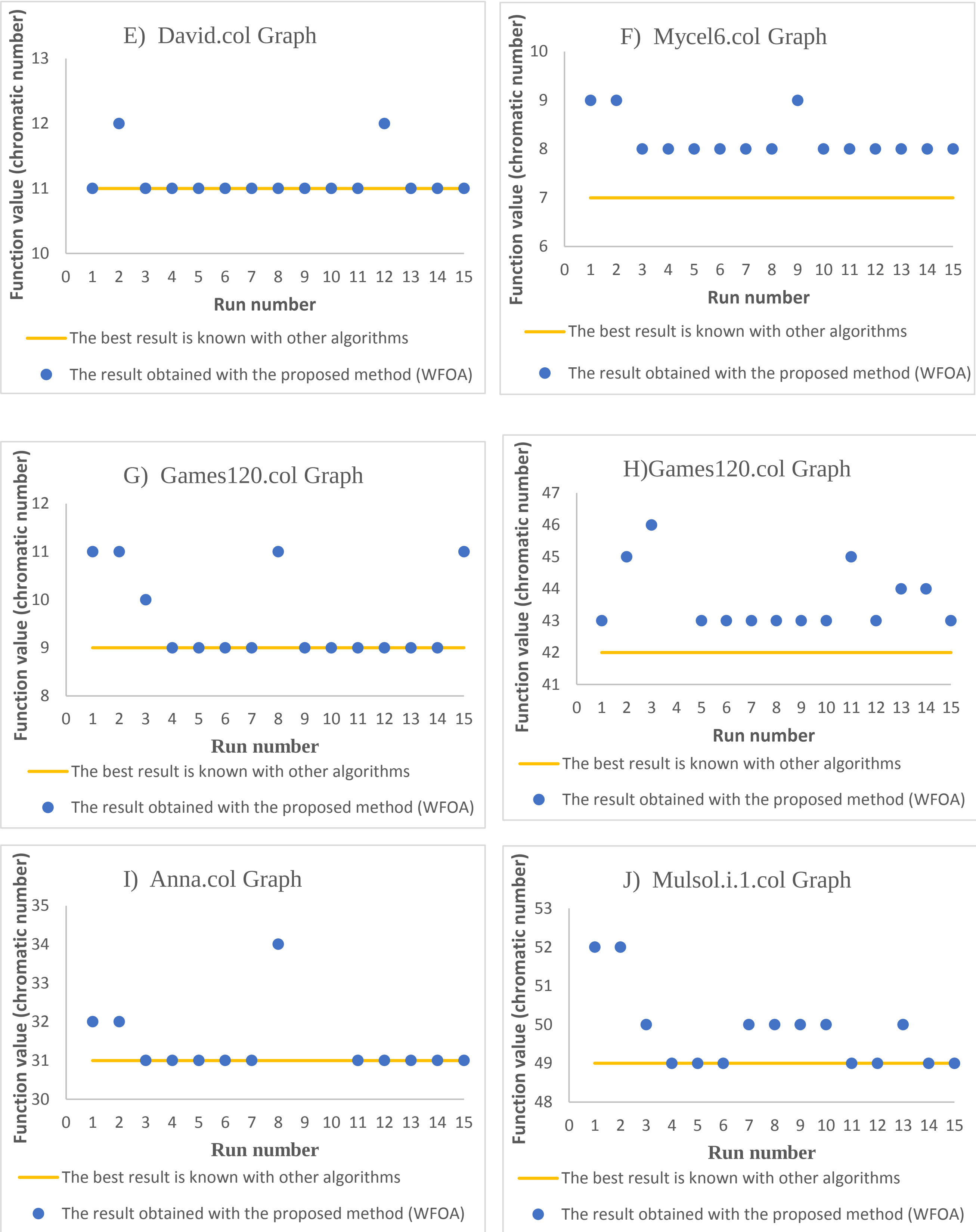
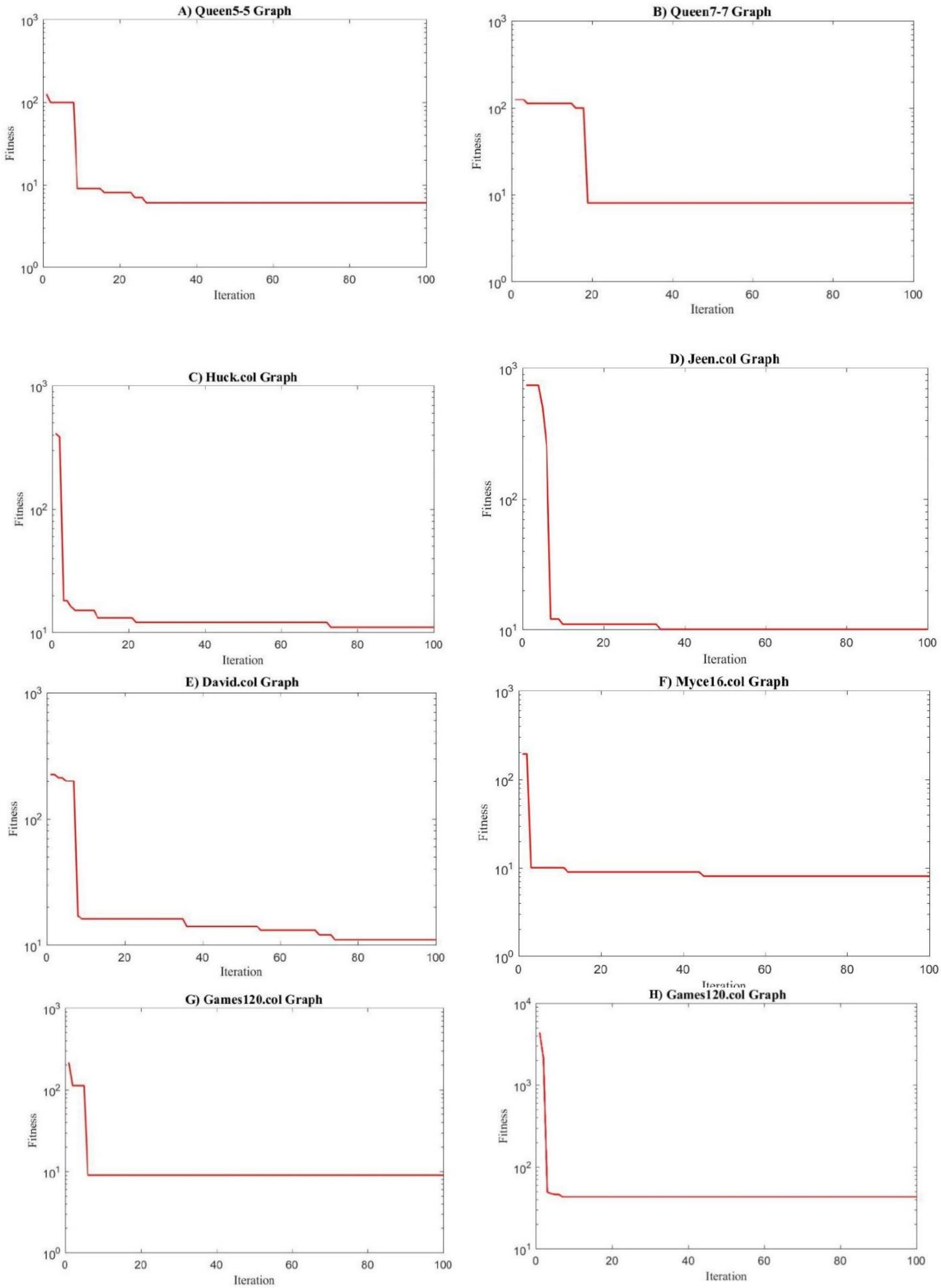


Fig. 9. Stability diagram of FWOA algorithm for graphs sample.

The convergence of the FWOA algorithm

Fig.10 shows convergence of proposed algorithm on sample of graphs, than fitness function in 100 repetitions. Each of diagrams are one of convergence diagrams that was performed in 15 replications. According to diagrams, we see that algorithm converges to the optimal solution after several initial replication (less than 20 replicates). In sample (J), it has been optimized at iteration 40. This can be due to the complexity of the graph. The results of the convergence of the algorithm show a significant improvement. The main reason for this progress is that The main reason for this progress is that algorithm starts with a better initial solution and algorithm searches for a response in one of trapezoidal fuzzy areas.



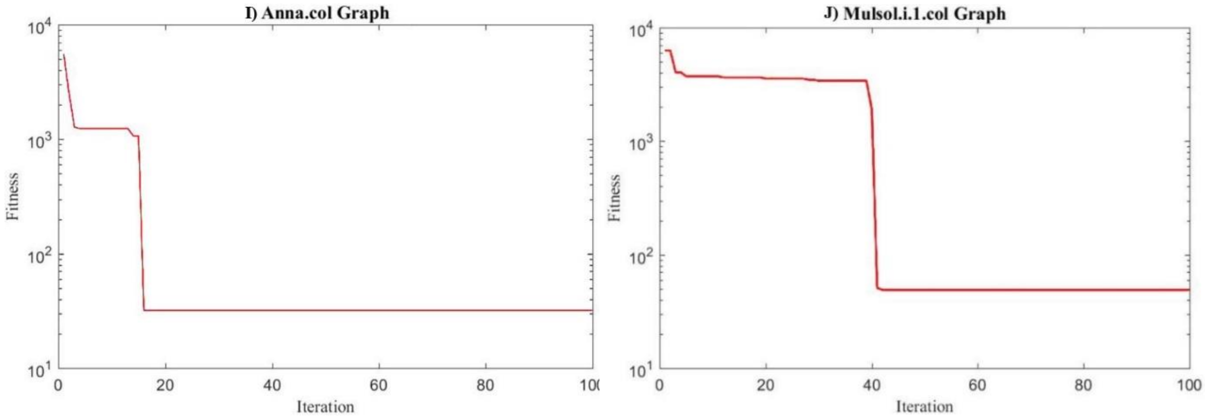


Fig. 10. Convergence diagram of FWOA algorithm for graphs sample.

CONCLUSION AND FUTURE WORKS

This paper proposes a new approach to solve GCP using a whale algorithm based on fuzzy logic. Capability of FWOA in discrete space will prepare a motivation. Idea to discretization of FWOA that produced by implementation, effective results of this discretization is procured. Discretization on the most popular combinational problems (GCP) is tested and evaluated, GCP have done results on a range of benchmark graph of this problem and excavated and we compared proposed algorithms in this contents. In this study it was tried to a wide range with all attributes of graph selected to test.

Since the GCP is a NP-hard problem as well, some graphs need more time to solve that we based on a maximum iterative algorithm and it can be improved by augmenting quality and better results is acquired. Results illustrate that FWOA algorithm is original focus of this work in most cases success rate is nearly 100% but as we have illustrated that compared with other manners, we cannot deduce that our algorithm is best in all instance of graphs. It can be said that proposed algorithm is able to compete with other algorithms in this context. This method creates a good balance between speed and accuracy. Results of this paper could be helpful to solve pragmatic and applied problems of graph problems.

Due to flexibility of the FWOA algorithm in combination with other ultra-exploratory algorithms, powerful algorithms can be developed to solve the GCP and obtaining better results in future work. Also, in the future, to solve large and complex graphs, we can use fuzzy clustering in evaluation section of proposed algorithm to improve runtime and convergence of algorithm.

REFERENCES

Abbasian, R. and M. Mouhoub An efficient hierarchical parallel genetic algorithm for graph coloring problem, ACM.

Abbasian, R. and M. Mouhoub (2013). "A hierarchical parallel genetic approach for the graph coloring problem." Applied intelligence 39(3): 510-528.

Agnarsson, G. and R. Greenlaw (2007). Graph theory: Modeling, applications, and algorithms, Pearson/Prentice Hall.

Alves, M. S., J. R. Nascimento and U. S. Souza (2021). "On the complexity of coloring-graphs." International Transactions in Operational Research 28(6): 3172-3189.

Astuti, W. (2015). "Graph Coloring Based on Evolutionary Algorithms to Support Data Hiding Scheme on Medical Images." Procedia Computer Science 74: 173-177.

Bensouyad, M. and D. Saidouni A discrete flower pollination algorithm for graph coloring problem, IEEE.

Biggs, N., N. L. Biggs and E. N. Biggs (1993). Algebraic graph theory, Cambridge university press.

Bui, T. N., T. H. Nguyen, C. M. Patel and K.-A. T. Phan (2008). "An ant-based algorithm for coloring graphs." Discrete Applied Mathematics 156(2): 190-200.

Chen, K. and H. Kanoh A discrete firefly algorithm based on similarity for graph coloring problems, IEEE.

Chiarandini, M. and T. Stützle An analysis of heuristics for vertex colouring, Springer.

Cui, G., L. Qin, S. Liu, Y. Wang, X. Zhang and X. Cao (2008). "Modified PSO algorithm for solving planar graph coloring problem." Progress in Natural Science 18(3): 353-357.

Deo, N. (2017). Graph theory with applications to engineering and computer science, Courier Dover Publications.

Djeloul, H., A. Layeb and S. Chikhi (2015). "Quantum Inspired Cuckoo Search Algorithm for Graph Coloring Problem." Quantum 22: 23.

Douiri, S. M. and S. Elbernoussi (2015). "Solving the graph coloring problem via hybrid genetic algorithms." Journal of King Saud University-Engineering Sciences 27(1): 114-118.

Evans, J. (2017). Optimization algorithms for networks and graphs, Routledge.

Faraji, M. and H. H. S. Javadi (2011). "Proposing a new algorithm based on bees behavior for solving graph coloring." International Journal Contemp. Math. Sciences 6(1): 41-49.

Fister Jr, I., X.-S. Yang, I. Fister and J. Brest (2012). "Memetic firefly algorithm for combinatorial optimization." arXiv preprint arXiv:1204.5165.

Galinier, P. and A. Hertz (2006). "A survey of local search methods for graph coloring." Computers & Operations Research 33(9): 2547-2562.

Goudet, O., C. Grelier and J.-K. Hao (2021). "A deep learning guided memetic framework for graph coloring problems." arXiv preprint arXiv:2109.05948.

Grimaldi, R. P. (2000). Discrete mathematics and combinational 1, Fatemi.

Guichard, D. (2017). "An Introduction to Combinatorics and Graph Theory." Whitman College-Creative Commons.(Cité en pages 33, 63 et 98.).

Han, L. and Z. Han A novel bi-objective genetic algorithm for the graph coloring problem, IEEE.

Hindi, M. and R. V. Yampolskiy Genetic Algorithm Applied to the Graph Coloring Problem.

Hsu, L.-Y., S.-J. Horng, P. Fan, M. K. Khan, Y.-R. Wang, R.-S. Run, J.-L. Lai and R.-J. Chen (2011). "MTPSO algorithm for solving planar graph coloring problem." Expert systems with Applications 38(5): 5525-5531.

Jensen, T. R. and B. Toft (2011). Graph coloring problems, John Wiley & Sons.

Kamath, V. and A. Pai (2018). "Applications of Graph Theory in Molecular Similarity Measurements." Research Journal of Pharmacy and Technology 11(4): 1375-1377.

Kaveh, A. and M. I. Ghazaan (2017). "Enhanced whale optimization algorithm for sizing optimization of skeletal structures." Mechanics Based Design of Structures and Machines 45(3): 345-362.

Kokosinski, Z. and Ł. Ochał Evaluation of metaheuristics for robust graph coloring problem, IEEE.

Mahmoudi, S. and S. Lotfi (2015). "Modified cuckoo optimization algorithm (MCOA) to solve graph coloring problem." Applied soft computing 33: 48-64.

Menaka, S. and J. Arthy "Study on complete coloring graph."

Mirjalili, S. and A. Lewis (2016). "The whale optimization algorithm." Advances in Engineering Software 95: 51-67.

Abbasian, R. and M. Mouhoub An efficient hierarchical parallel genetic algorithm for graph coloring problem, ACM.

Abbasian, R. and M. Mouhoub (2013). "A hierarchical parallel genetic approach for the graph coloring problem." Applied intelligence 39(3): 510-528.

Agnarsson, G. and R. Greenlaw (2007). Graph theory: Modeling, applications, and algorithms, Pearson/Prentice Hall.

Alves, M. S., J. R. Nascimento and U. S. Souza (2021). "On the complexity of coloring-graphs." International Transactions in Operational Research 28(6): 3172-3189.

Astuti, W. (2015). "Graph Coloring Based on Evolutionary Algorithms to Support Data Hiding Scheme on Medical Images." Procedia Computer Science 74: 173-177.

Bensouyad, M. and D. Saidouni A discrete flower pollination algorithm for graph coloring problem, IEEE.
Biggs, N., N. L. Biggs and E. N. Biggs (1993). Algebraic graph theory, Cambridge university press.
Bui, T. N., T. H. Nguyen, C. M. Patel and K.-A. T. Phan (2008). "An ant-based algorithm for coloring graphs." Discrete Applied Mathematics 156(2): 190-200.
Chen, K. and H. Kanoh A discrete firefly algorithm based on similarity for graph coloring problems, IEEE.
Chiarandini, M. and T. Stützle An analysis of heuristics for vertex colouring, Springer.
Cui, G., L. Qin, S. Liu, Y. Wang, X. Zhang and X. Cao (2008). "Modified PSO algorithm for solving planar graph coloring problem." Progress in Natural Science 18(3): 353-357.
Deo, N. (2017). Graph theory with applications to engineering and computer science, Courier Dover Publications.
Djeloul, H., A. Layeb and S. Chikhi (2015). "Quantum Inspired Cuckoo Search Algorithm for Graph Coloring Problem." Quantum 22: 23.
Douiri, S. M. and S. Elbernoussi (2015). "Solving the graph coloring problem via hybrid genetic algorithms." Journal of King Saud University-Engineering Sciences 27(1): 114-118.
Evans, J. (2017). Optimization algorithms for networks and graphs, Routledge.
Faraji, M. and H. H. S. Javadi (2011). "Proposing a new algorithm based on bees behavior for solving graph coloring." International Journal Contemp. Math. Sciences 6(1): 41-49.
Fister Jr, I., X.-S. Yang, I. Fister and J. Brest (2012). "Memetic firefly algorithm for combinatorial optimization." arXiv preprint arXiv:1204.5165.
Galinier, P. and A. Hertz (2006). "A survey of local search methods for graph coloring." Computers & Operations Research 33(9): 2547-2562.
Goudet, O., C. Grelier and J.-K. Hao (2021). "A deep learning guided memetic framework for graph coloring problems." arXiv preprint arXiv:2109.05948.
Grimaldi, R. P. (2000). Discrete mathematics and combinational 1, Fatemi.
Guichard, D. (2017). "An Introduction to Combinatorics and Graph Theory." Whitman College-Creative Commons.(Cité en pages 33, 63 et 98.).
Han, L. and Z. Han A novel bi-objective genetic algorithm for the graph coloring problem, IEEE.
Hindi, M. and R. V. Yampolskiy Genetic Algorithm Applied to the Graph Coloring Problem.
Hsu, L.-Y., S.-J. Horng, P. Fan, M. K. Khan, Y.-R. Wang, R.-S. Run, J.-L. Lai and R.-J. Chen (2011). "MTPSO algorithm for solving planar graph coloring problem." Expert systems with Applications 38(5): 5525-5531.
Jensen, T. R. and B. Toft (2011). Graph coloring problems, John Wiley & Sons.
Kamath, V. and A. Pai (2018). "Applications of Graph Theory in Molecular Similarity Measurements." Research Journal of Pharmacy and Technology 11(4): 1375-1377.
Kaveh, A. and M. I. Ghazaan (2017). "Enhanced whale optimization algorithm for sizing optimization of skeletal structures." Mechanics Based Design of Structures and Machines 45(3): 345-362.
Kokosinski, Z. and Ł. Ochał Evaluation of metaheuristics for robust graph coloring problem, IEEE.
Mahmoudi, S. and S. Lotfi (2015). "Modified cuckoo optimization algorithm (MCOA) to solve graph coloring problem." Applied soft computing 33: 48-64.
Menaka, S. and J. Arthy "Study on complete coloring graph."
Mirjalili, S. and A. Lewis (2016). "The whale optimization algorithm." Advances in Engineering Software 95: 51-67.
Mondal, B. and K. De (2017). "Overview Applications of Graph Theory in Real Field." International Journal of Scientific Research in Computer Science, Engineering and Information Technology 2(5): 751-759.
Morris, J. and F. Song (2021). "Simple vertex coloring algorithms." arXiv preprint arXiv:2102.07089.
Mostafaie, T., F. M. Khiyabani and N. J. Navimipour (2020). "A systematic study on

meta-heuristic approaches for solving the graph coloring problem." Computers & Operations Research 120: 104850.
Pal, A. J., B. Ray, N. Zakaria and S. S. Sarma (2012). "Comparative performance of modified simulated annealing with simple simulated annealing for graph coloring problem." Procedia Computer Science 9: 321-327.
Panda, P. R., N. D. Dutt and A. Nicolau (1997). "Memory data organization for improved cache performance in embedded processor applications." ACM Transactions on Design Automation of Electronic Systems (TODAES) 2(4): 384-409.
Postigo, J., J. Soto-Begazo, V. R. Fiorela, G. M. Picha, R. Flores-Quispe and Y. Velazco-Paredes (2021). Comparative Analysis of the main Graph Coloring Algorithms. 2021 IEEE Colombian Conference on Communications and Computing (COLCOM), IEEE.
Rao, N. S. V. (1993). "Computational complexity issues in operative diagnosis of graph-based systems." IEEE Transactions on Computers 42(4): 447-457.
Schmidt, C., N.-E. Guenther and L. Zdeborová (2016). "Circular coloring of random graphs: statistical physics investigation." Journal of Statistical Mechanics: Theory and Experiment 2016(8): 083303.
Tomar, R. S., S. Singh, S. Verma and G. S. Tomar A novel abc optimization algorithm for graph coloring problem, IEEE.
Trinajstic, N. (2018). Chemical graph theory, Routledge.
Trudeau, R. J. (1993). Introduction to Graph Theory (Corrected, enlarged republication. ed.), New York: Dover Pub.
Vecchio, D. A., S. H. Mahler, M. D. Hammig and N. A. Kotov (2021). "Structural Analysis of Nanoscale Network Materials Using Graph Theory." ACS nano 15(8): 12847-12859.
Xu, Y. and Y. Chen (2021). "A Cuckoo Quantum Evolutionary Algorithm for the Graph Coloring Problem." arXiv preprint arXiv:2108.08691.
Zhou, J., S.-C. Fang and W. Xing (2017). "Graph coloring is a classical NP-hard combinatorial optimization problem with many practical applications. A broad range of heuristic methods exist for tackling the graph coloring problem: from fast greedy algorithms to more time-consuming metaheuristics. Although the latter produce better results in terms of minimizing the number of colors, the former are widely employed due to their simplicity." Computational Optimization and Applications 66(1): 163-185.