Bulletin of Electrical Engineering and Informatics
Vol. 11, No. 3, June 2022, pp. 1526~1534
ISSN: 2302-9285, DOI: 10.11591/eei.v11i3.3694

Intelligent university timetable scheduling system using sudoku grid with magic square

Isra Natheer Alkallak[1], Ruqaya Zedan Shaban[2], Yazen Hudhaifa Shakir Alnema[3]
[1]Department Basic Sciences, College of Nursing, University of Mosul, Mosul, Iraq
[2]Computer unit, College of Medicine, University of Mosul, Mosul, Iraq
[3]Department of System and Control Engeneering, College of Electronics Engineering, Ninevah University, Mosul, Iraq

Article Info

Article history:

Received Feb 13, 2022
Revised Apr 2, 2022
Accepted Apr 15, 2022

Keywords:

Exam timetabling
Heuristic
Magic square
Sudoku grid

ABSTRACT

Examination timetabling is problem faced by many academic educational institutions. The situation formulated as artificial intelligence problems as Sudoku grid with Magic square. Problem assigning a set of examinations to a fixed number of examination periods so that no student is required to take more than one examination at a time. The research describes two proposed heuristic algorithms to create university examination timetabling and avoiding the overlapping of exams. We proposed two algorithms, the first algorithm is based on Sudoku grid, a number of hard constraints as well as a number of soft constraints are proposed. We assigned exams as 9 to period as 27 slot times and to 3 rooms, with 3 classes. Each student in each class takes 9 exams While, the second algorithm simulates the classic manual methods while observing the hard and soft constraints of the problem. The heuristic approach used to further optimize the result to include hard constraints with soft. The proposed methodology for creating the Exam timetabling problem has been implemented in Matlab and closing the gap between classical methods and intelligent methods. The results obtained with very less computational effort through two algorithms.

Corresponding Author:

Isra Natheer Alkallak
Department Basic Sciences, College of Nursing, University of Mosul
Mosul, Iraq
Email: alkalak.isra@uomosul.edu.iq

1. INTRODUCTION

Examination timetabling problem is combinatorial problems [1]-[3]. The task of creating an examination timetable is an NP-Hard combinatorial optimization problem. Assignment of courses to be examined as well as applicants to time periods, can be defined. It is a significant issue in every educational institution [4]-[7]. It arranges a series of examinations in a set of time slots and no student is assigned to more than one exam at a time. Exams are assigned to days and time windows within those days based on availability, fair distribution of student workload, and room capacities [8]-[11]. It entails arranging a set of examinations to set of exam periods in such a way that the problem's hard constraints are met while soft constraints are minimized. From one institution to the next, the hard and soft limitations are different [11]-[13]. Exam timetabling is a difficult task that involves scheduling tests, minimizing exam overlap, and spreading exams out as widely as feasible for students [14], [15]. The meeting is made up of a variety of resources, including students and rooms. The soft constraint is to provide each student day break between two consecutive examinations. Alkallak has completed many scientific articles in the matter of the Magic Square and the Sudoku grid, including Alkallak and Shaban [16] solved the rostering problem, while Alkallak [17] solved Sudoku grid using Magic square of order3. Also, Alkallak *et al.* [18] proposed a hybrid algorithm to construct knight tour by Sudoku grid. This research, begins with a Sudoku grid which is solved by Alkallak [17] through the magic square of order3. The algorithm by Alkallak is the cornerstone in proposing creating university exam timetabling and prepared a map for students to sit in their seats and exams associated to timeslots.

The aims of the research to connecting the bridge between manual and intelligent algorithms for examination timetabling by as set of examinations allocated into exam period. Overlap or harmony between the basics of the Magic square and the Sudoku network to solve the problem of Timetabling. The proposed algorithm guarantees to access solutions. To find a good solution of timetabling. To minimize the number of conflicts in the Examination timetabling problem. This research structured several sections, in first section was introduction while the second section were preliminary of Examination Timetable Problem with Sudoku grid. In third section reviewed the specifications of the first proposed exam schedule, proposed hard constraints and proposed soft constraints. Proposed algorithms appeared in section four, while section five contained the result and discussion. Section six were conclusion.

2. PRELIMINARY OF EXAMINATION TIMETABLE PROBLEM WITH SUDOKU GRID

Exam schedules are important and challenging for educational institutions because they require very human and computer resources and must be performed several times each year[19], [20]. Set of examinations, examination period to start date and end it [21]. Set of rooms each room has examination seating capacity. Timetabling can be defined to be the problem of allocating set of examinations to generate no conflicts between any two examinations. The task of scheduling examination timetables was previously done manually [22]-[25]. Sudoku is combinatorial optimization problem. Sudoku puzzle consists of 81 cells, contained in 9×9 cells [17], [18].

3. SPECIFICATIONS OF THE FIRST PROPOSED EXAM SCHEDULE ALGORITHM

This research used artificial intelligent ideas for creating scheduling the exams. The research described an approach of the problem is characterized by assigning a set of exams into a limited number of timeslots subject to set constraints. The constraints are classified as hard and soft constraints. The specifications of the proposed exam schedule as follows:

a. The number of examinations per day is 1.
b. Sudoku has nine original numbers, thus total number of scheduled events (exams) for each class are 9.
c. The Sudoku grid has three vertical sub-grids left, center and right, so we have three set where all three vertical sub-grids are assigned to the class.
d. Also, the number of examination rooms are 3. This rooms represented by three Sudoku grids.
e. The summation for each row, column and diagonal in Magic square are 15, thus number of teachers are 15.
f. Number of days are 27. The total number of exam days is 27 therefore, using three Sudoku grids.
g. Number of Sudoku cells are 81 cells, thus number of students are 81 for each class.
h. Off day between exams slot is met.
i. Take advantage of the maximum time limit for exams with an early start.
j. The number of seats is sufficient, one exam for one class per day, in order to conduct distancing through Covid 19.
k. The student takes the exam in different rooms, and accordingly, the rooms are diversified.
l. The number of students in one examination room is 27 students.
m. Each student has one seat in the examination room.
n. Since we have three classes and, in each class, 81 students take the exam, then the total number of students is 243 students.
o. Give preference to some exams.

Proposed hard constraints as follows:

a. Overlap needs to avoid overlapping exam days need to take in the slot day. A student donot assigned to two exams as same time. No student assigned to different events at same time. Separate all examinations with different duration. A student donot assigned to two exams as same time. No student assigned to different events at same time. Separate all examinations with different duration.
b. Capacity which includes examination room availability. No examination room is assigned to different events at same time. There cannot be any students sitting for more than one exam at the same time. The total number of students assigned to each examination room cannot exceed the examination room capacity through enough rooms for all the students. Health teams sterilize examination room to prevent the spread of the COVID-19 pandemic.
c. Examination period a start date and end it. All examinations of a student over the examination period as much as possible. Should not violate the timeslot length of exams.

d. Spread all student can attend the exam in the same room or not.

Proposed soft constraints as follows:

a. Spread exams in several examination room.
b. Schedule all exams, while the exams had priority, as possible. Distribution preferences is set between two or more exams.
c. Optimum utilization of room space is achieved during examinations. There is no more than one test in the room, through virus Covid 19.
d. Each lecturer has at least one time gap between invigilation sessions.

4. PROPOSED ALGORITHMS

In this research, we proposed two algorithms. First algorithm contained two phases. The number of solutions is eight with the number of Magic Square of order 3 models. While second algorithm simulated the manual method to overlapping between the manual and intelligent approaches.

4.1. First phase in first proposed algorithm

The steps of proposed algorithm with illustrated graphically to create the timetabling as follows:

Step 1: initialize of Sudoku grid solution as Figure 1. So, the matrix is 9×9. Magic square is foundation in Sudoku grid solving. Figure 1 shows the result of the researcher'sAlkallak [17], and this figure is the cornerstone of the proposed algorithm.

Step 2**:** from Figure 2, we assigned three vertical sub grids to first class, three vertical sub grids to second class and three vertical sub grids to third class.

Step 3: the duration period of exams is 27 days. It distributed by three Sudoku grids 9×9. The total matrix 27×9. In The first Sudoku grid, for each row numbered from 1-9, while in the second Sudoku grid, for each row numbered from 10-18. Also, Figure 3 shows the third Sudoku grid, for each row numbered from 19-27.

Step 4: to create the exam timetabling by checking the cell with value one in the first sub grid. It assigned to the first exam in first class, while we checked the cell with value one in the second sub grid to assign to first exam in second class. In addition, we checked the cell with value one in the third sub grid to assign to first exam in third class by depending on Figure 1. Thus, all cells are examined for the rest of the numbers from 2 to 9 to determine the rest of the exams for the three classes.

Step 5: End.

2	**7**	**6**	**4**	**3**	**8**	**9**	**5**	**1**
9	**5**	**1**	**2**	**7**	**6**	**4**	**3**	**8**
4	**3**	**8**	**9**	**5**	**1**	**2**	**7**	**6**
6	**2**	**7**	**8**	**4**	**3**	**1**	**9**	**5**
1	**9**	**5**	**6**	**2**	**7**	**8**	**4**	**3**
8	**4**	**3**	**1**	**9**	**5**	**6**	**2**	**7**
7	**6**	**2**	**3**	**8**	**4**	**5**	**1**	**9**
5	**1**	**9**	**7**	**6**	**2**	**3**	**8**	**4**
3	**8**	**4**	**5**	**1**	**9**	**7**	**6**	**2**

Figure 1. Image of the result of the researcher'sAlkallak 2012

First class	**Second class**	**third class**
First class	**Second class**	**third class**
First class	**Second class**	**third class**

Figure 2. Assignment class

	First class			Second class			Third class		
1									Ex 1
2			Ex 1						
3						Ex 1			
4		Ex 2							
5					Ex 2				
6								Ex 2	
7				Ex 3					
8							Ex 3		
9	Ex 3								
10				Ex 4					
11							Ex 4		
12	Ex 4								
13									Ex 5
14			Ex 5						
15						Ex 5			
16		Ex 6							
17					Ex 6				
18								Ex 6	
19		Ex 7							
20					Ex 7				
21								Ex 7	
22				Ex 8					
23							Ex 8		
24	Ex 8								
25									Ex 9
26			Ex 9						
27						Ex 9			

Ex 1 represented the value one in Figure 1 in the three sub grids for first horizontal three sub grids (first sudoku).

Ex 2 represented the value two in Figure 1 in the three sub grids for second horizontal three sub grids (first sudoku).

Ex 3 represented the value three in Figure 1 in the three sub grids for third horizontal three sub grids (first sudoku).

Ex 4 represented the value four in Figure 1 in the three sub grids for first horizontal three sub grids (second sudoku).

Ex 5 represented the value five in Figure 1 in the three sub grids for second horizontal three sub grids (second sudoku).

Ex 6 represented the value six in Figure 1 in the three sub grids for third horizontal three sub grids (second sudoku).

Ex 7 represented the value seven in Figure 1 in the three sub grids for first horizontal three sub grids (third sudoku).

Ex 8 represented the value eight in Figure 1 in the three sub grids for second horizontal three sub grids (third sudoku).

Ex 9 represented the value nine in Figure 1 in the three sub grids for third horizontal three sub grids (third sudoku).

Figure 3. Three Sudoku grids represented duration period

4.2. A proposed map of the student sitting in the examination room

Since number of students in the exam room is 81, as 9×9 matrix, to create the map of the seating for students where Figure 4 shows the map of the student sitting will be as follows:

Step 1: initialize generate three Sudoku grids. In the first, the grids are empty. The students divided into three grids.

Step 2: from Figure 1, the values in the diagonal, in the three sub grids for first horizontal three sub grids are relied, by assigning the values in main diagonal in sub matrixes as first, second and third.

Step 3: to generate the three sub grids for second horizontal three sub grids by shifting the first sub grid to the left in step 2,

Step 4: to generate the three sub grids for third horizontal three sub grids by shifting the second sub grid to the left in step 3.

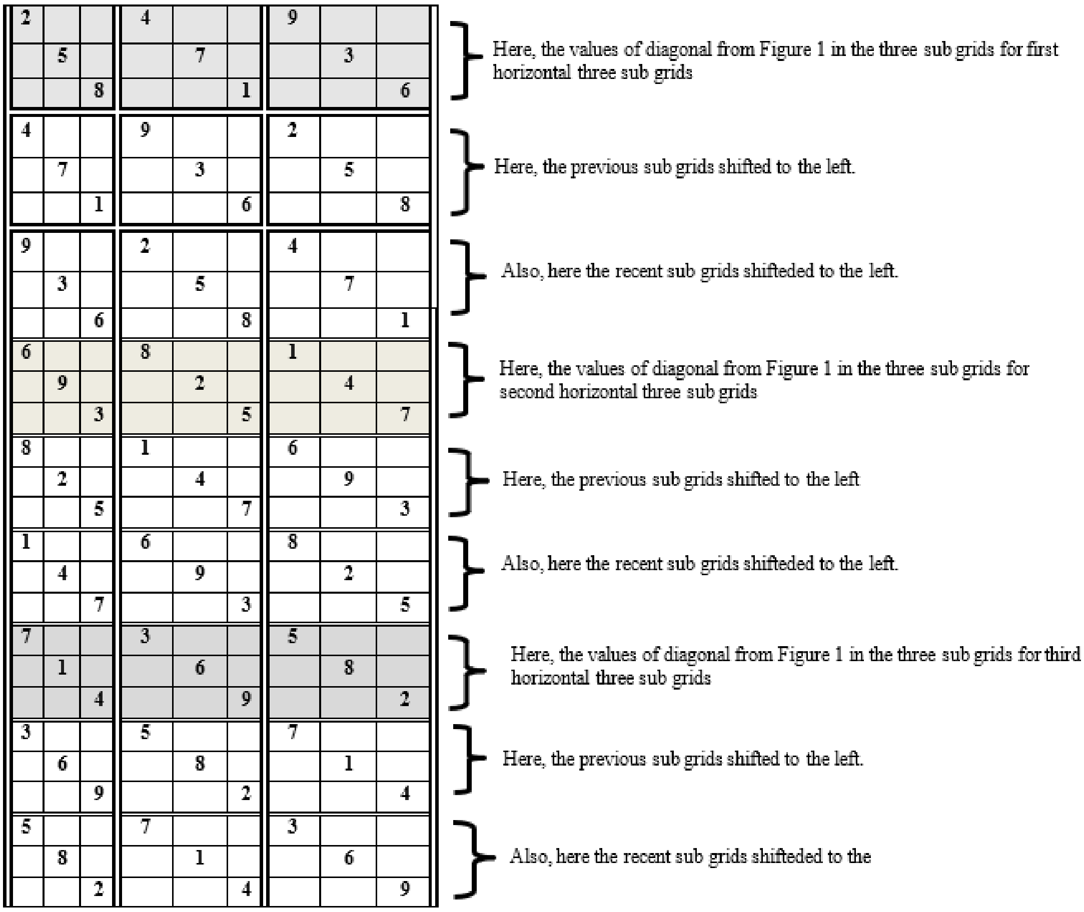

Figure 4. Map of the student sitting

4.3. Second proposed algorithm

The Figure 5 shows the flowchart of second proposed algorithm. The steps of the second proposed algorithm for solving timetable are:

Step 1: initialization
Begin
Let day ; no. of days exam.
Let class ; no. of classes.
Create matrix t(x,y) ; where x are number of the classes, y are no. of exams in each class.
let the matrix hs equal the number of hard and soft exam in each class.

Matrix t	
x	y
1	8
2	5
:	:
:	:
class	exams

```
For i=1 to class
   For j=1 to hs(i,2)
   Input exam_hard
   ah(j,i)= exam_hard
    End
```

```
End
For i=1 to class
   For j=1 to hs (i,3)
   Input examt_soft
   as(j,i)= exam_soft
    End
End
```

For example, Figure 6 Matrix (ah) represented hard exams graphically and Figure 7 Matrix (as) represented soft exams graphically.

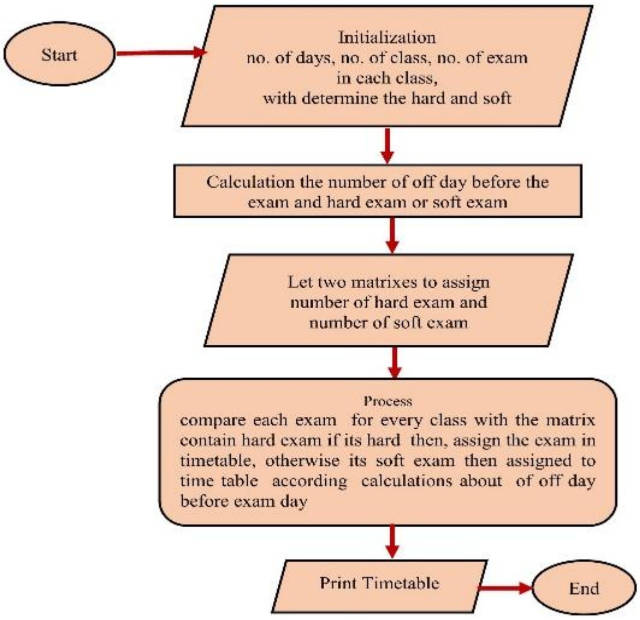

Figure 5. Flowchart of second proposed algorithm

Class1h	Class2h	Class3h
Ex 3	Ex 1	Ex 2
Ex 4	Ex 3	Ex 4
Ex 6	Ex 5	Ex 5
		Ex 6

Figure 6. Matrix (ah)

Class1s	Class2s	Class3s
Ex 1	Ex 2	Ex 1
Ex 2	Ex 4	Ex 3
Ex 5		
Ex 7		
Ex 8		

Figure 7. Matrix (as)

Step 2:

To calculate the number of off days that through exams, by divide total number of days on total number of exams. This operation produced the integer and remainder values as follows:n

```
For i=1 to class
    in = int(day/t(i,2))       ; represent the number of off days.
    r =day mod t(i,2)          ; add to off days for hard exam.
    z(i,1)=in
    z(i,2)=r
End
        For i=1 to class
               z(i ,1)=z( i,1) -1
      End
```

For example, Figure 8 matrix z represented off days graphically.

Matrix z			
No. of off days that through soft exam	1	3	2
No. of off days distributed before the hard exam	4	0	2

Figure 8. Matrix z represented Off days graphically

Step 3: process

compare each exam for every class with the matrix contain hard exam if its hard then, assign the exam in timetable otherwise, if its soft exam then assigned to time table according calculations about of off day before exam day.

```
For m=1 to c
      date =0
       k=  1
              While  k <> T(m,2)
      temp1 = z(m,1)         ; no. of days between for soft exam.
      temp2 = hs(m,2)       ; no. of hard exam.
      temp3=z(m,2)/hs(m,2)
               For v =1 to temp2
                            If a(k,1)==ah(v,1)
                               intable = date+temp1+temp3
                                table(intable)=a(k,1)  ;assign the exam a(k,1) to timetable
                               Break
                           Else
                        End
                            For w=1 to   hs(m,3)
                                If a(k,1)==as(w,1)
                               intable=date+temp1
                               table(intable)=a(k,1) ;assign the exam a(k,1) to timetable
                                 Break
                                End
                          Else
                     End
                          date=date+ temp1
                           k=k+1
                                End
End
```

5. RESULTS AND DISCUSSION

The problem has three classes in one day we assign exam subject to timeslot for one class. The first day is devoted to the examination for the first-grade students, then the second day is devoted to the examination for the second-grade students. As for the third day, it is devoted to taking the exam for the third-grade students. The goal is also to try to spread the tests throughout the test period. We try to customize each test for the best period of time that meets all the tough restrictions. We assign tests arbitrarily to time intervals when the test does not conflict with the tests that are already scheduled. The Figure 9 illustrated timetabling solved for first proposed algorithm. We conclude that the numbers from one to nine enable us to distribute the exams and also to generate the timetabling, as the scattering of each number in Sudoku in its rows and columns enabled us to generate the timetabling. Since the Sudoku sub grid consists of three rows and three columns, and the total number of sub grids is nine, and since the product of three times nine is 27, the maximum time period for exams is 27.

We note in the seating map that the number of students is equal to 27 students in each room, and thus the total number becomes 81 students for one class. We note that sitting according to the restrictions of the Magic Square, there is no repetition of the student number in the row, column, sub-diagonal, main diagonal. We note that there is a void next to the student in order to achieve distancing between students in the time of Covid 19. We say that there is a magic relationship between sudoku grid and magic square. The first proposed algorithm proved its superiority in scheduling exams in an intelligent approach, through the overlap between the concept of scheduling exams with one of the artificial intelligence problem, including the Sudoku grid.

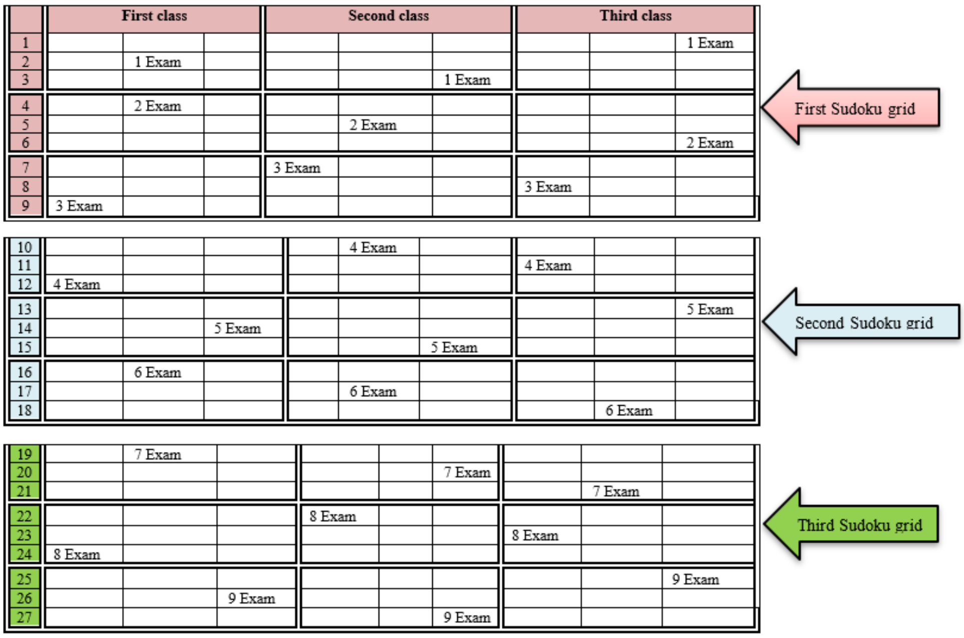

Figure 9. Timetabling solved through three sudoku grids

6. CONCLUSION

Despite the different conduct of exam scheduling in educational institutions, artificial intelligence methods have highlighted their role in finding the best solution to the problem. We conclude that there is a magical attraction of the Magic Square and the Sudoku grid to solve the problem of timetabling exams. This research dealt with the timetable of the exams of an educational institution by suggesting two algorithms and using intelligence. We get disqualified students who take two consecutive exams on the same day. We get the distribution of exams over time intervals. Manual methods are done within several days or weeks of iterative repair. In addition, in this research, we have to define a certain set of exams, so that no student takes two exams at the same time. The proposed algorithm is able to produce high quality solutions when compared with traditional methods, through a set of tests assigned in the testing period. The proposed algorithm guarantees to find solutions. To find an exact and good solution timetabling. The proposed algorithm eliminated the gap between classical methods and intelligent methods. It had been able to generate a better timetabling.

ACKNOWLEDGEMENTS

The authors research their gratitude to the College of Nursing, the Medicine College, Mosul University and College of Electronics Engineering of Ninevah University for the facilities they provided, which helped improve the quality of this work.

REFERENCES

[1] R. Qu, E. Burke, B. Mccollum, L. T. G. Merlot, S. Y. Lee, and C. R. Qu, "A Survey of Search Methodologies and Automated Approaches for Examination Timetabling A Survey of Search Methodologies and Automated Approaches for Examination," *J. Sched.*, vol. 12, no. 1, pp. 55–89, 2009, doi: 10.1007/s10951-008-0077-5.

[2] M. A. Jebur and H. S. Abdullah, "Timetabling problem solving based on best-nests cuckoo search," *Bull. Electr. Eng. Informatics*, vol. 10, no. 6, pp. 3333–3340, 2021, doi: 10.11591/eei.v10i6.3206.

[3] M. Alzaqebah and S. Abdullah, "The Bees Algorithm for Examination Timetabling Problems," *International Journal of Soft Computing and Engineering*, vol. 1, no. 5, pp. 105–110, 2011.

[4] M. A. Selemani, E. Mujuni, and A. Mushi, "En Examination Scheduling Algorithm Using Graph Coloring – the case of Sokoine University of Agriculture," *International Journal of Computer Engineering and Applications*, vol. 2, no. 1, pp. 116-127, 2013.

[5] E. S. Sin, "Hyper Heuristic Based on Great Deluge and its Variants for Exam Timetabling Problem," *Int. J. Artif. Intell. Appl.*, vol. 3, no. 1, pp. 149–162, 2012, doi: 10.5121/ijaia.2012.3112.

[6] E. Mujuni and A. Mushi, "Solving the Examination Timetabling Problem Using a Two-Phase Heuristic: The case of Sokoine University of Agriculture," *J. Inf. Comput. Sci.*, vol. 10, no. 3, pp. 220–227, 2015.

[7] N. R. Sabar, M. Ayob, R. Qu, and G. Kendall, "A graph coloring constructive hyper-heuristic for examination timetabling problems," *Appl. Intell.*, vol. 37, no. 1, pp. 1–11, 2012, doi: 10.1007/s10489-011-0309-9.

[8] B. Al-Khateeb and A. Turki, "Meerkat swarm optimization algorithm for real world university examination timetabling problem," *J. Adv. Res. Dyn. Control Syst.*, vol. 10, no. 13, pp. 2103–2113, 2018.

[9] E. Özcan, M. Misir, G. Ochoa, and E. K. Burke, "A Reinforcement Learning-Great-Deluge Hyper-Heuristic for Examination Timetabling," *Int. J. Appl. Metaheuristic Comput.*, vol. 1, no. 1, pp. 39–59, Jan. 2010, doi: 10.4018/jamc.2010102603.

[10] A. K. Mandal and M. N. M. Kahar, "Performance analysis of graph heuristics and selected trajectory metaheuristics on examination timetable problem," *Indones. J. Electr. Eng. Informatics*, vol. 8, no. 1, pp. 163–177, 2020, doi: 10.11591/ijeei.v8i1.1660.

[11] M. Mazlan, M. Makhtar, A. F. K. Ahmad Khairi, and M. A. Mohamed, "University course timetabling model using ant colony optimization algorithm approach," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 13, no. 1, pp. 72–76, 2019, doi: 10.11591/ijeecs.v13.i1.pp72-76.

[12] B. McCollum, P. McMullan, A. J. Parkes, E. K. Burke, and R. Qu, "A new model for automated examination timetabling," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 291–315, 2012, doi: 10.1007/s10479-011-0997-x.

[13] S. Abdullah, H. Turabieh, B. Mccollum, and P. McMullan, "A Tabu-based Memetic Approach to the Examination Timetabling Problem," in J. Yu, S. Greco, P. Lingras, G. Wang, and A. Skowron, (eds) *Rough Set and Knowledge Technology, Lecture Notes in Computer Science*, vol 6401, 2010, doi: 10.1007/978-3-642-16248-0_78.

[14] A. R. Mushi, "Two Phase Heuristic Algorithm for the University Course Timetabling Problem: The Case of University of Dar Es Salaam," *Tanzania J. Sci.*, vol. 37, no. 1, 2011.

[15] M. Alzaqebah and S. Abdullah, "Hybrid bee colony optimization for examination timetabling problems," *Comput. Oper. Res.*, vol. 54, pp. 142–154, 2015, doi: 10.1016/j.cor.2014.09.005.

[16] I. N. Alkallak and R. Z. Shaban, "Establishing a cyclic schedule for nurse in the health unit," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 3, pp. 2876–2884, 2022, doi: 10.11591/ijece.v12i3.pp2876-2884.

[17] I. N. Alkallak, "Using Magic Square of Order 3 to Solve Sudoku Grid Problem," *Tikrit J. Pure Sci.*, vol. 17, no. 4, pp. 244–248, 2012.

[18] I. N. Alkallak, Y. H. Alnema, and R. Z. Sha'ban, "A proposed hybrid algorithm for constructing knight tour problem by sudoku grid," *J. Adv. Res. Dyn. Control Syst.*, vol. 10, no. 10, pp. 2333–2342, 2018.

[19] B. Hussin, A. S. H. Basari, A. S. Shibghatullah, S. A. Asmai and N. S. Othman, "Exam timetabling using graph colouring approach," *2011 IEEE Conference on Open Systems*, 2011, pp. 133-138, doi: 10.1109/ICOS.2011.6079274.

[20] B. A. Aldeeb, M. A. Al-betar, A. O. Abdelmajeed, and M. J. Younes, "A Comprehensive Review of Uncapacitated University Examination Timetabling Problem," *Int. J. Appl. Eng. Res.*, vol. 14, no. 24, pp. 4524–4547, 2019.

[21] S. Dewi, R. Tyasnurita, and F. S. Pratiwi, "Solving examination timetabling problem within a hyperheuristic framework," *Bull. Electr. Eng. Informatics*, vol. 10, no. 3, pp. 1611–1620, 2021, doi: 10.11591/eei.v10i3.2996.

[22] S. K. N. A. Rahim, A. Bargiela, and R. Qu, "Hill Climbing versus genetic algorithm optimization in solving the examination timetabling problem," *ICORES 2013 - Proc. 2nd Int. Conf. Oper. Res. Enterp. Syst.*, pp. 43–52, 2013, doi: 10.5220/0004286600430052.

[23] P. Demeester, B. Bilgin, P. De Causmaecker, and G. Van Den Berghe, "A hyperheuristic approach to examination timetabling problems: Benchmarks and a new problem from practice," *J. Sched.*, vol. 15, no. 1, pp. 83–103, 2012, doi: 10.1007/s10951-011-0258-5.

[24] A. Muklason, G. B. Syahrani, and A. Marom, "Great deluge based hyper-heuristics for solving real-world university examination timetabling problem: New data set and approach," *Procedia Comput. Sci.*, vol. 161, pp. 647–655, 2019, doi: 10.1016/j.procs.2019.11.168.

[25] S. K. Jha, "Exam Timetabling Problem Using Genetic Algorithm," *Int. J. Res. Eng. Technol.*, vol. 3, no. 5, pp. 649–654, 2014, doi: 10.15623/ijret.2014.0305120.

BIOGRAPHIES OF AUTHORS

Isra Natheer Alkallak completed her M.Sc. in Computer Science/Artificial Intelligence in 2003 and B.Sc. in computer science in 1991. She has worked in the Basic Science Department in the College of the Nursing University of Mosul. Her research interests include optimization algorithms, heuristic, and swarm intelligence for applications. She can be contacted at email: alkalak.isra@uomosul.edu.iq.

Rukaya Zedan Sha'ban completed her M.Sc. in Computer Science/Artificial Intelligence 2001 and B.Sc. in computer science in 1989. She has worked in the Computer unit in the College of the Medicine University of Mosul. Her research interests include optimization algorithms, heuristic, and swarm intelligence for applications. She can be contacted at email: rzs@uomosul.edu.iq.

Yazen Hudhaifa Shakir Alnema is currently working as lecturer in Nineveh University in the field of robotics and automation in systems and control engineering department. He has awarded a Master degree with distinction (honor) in Mechatronics and Robotics Engineering from one of the superior 200 universities over the world, which is Leeds University in the United Kingdom. This scholarship has been fully funded by the Iraqi government in appreciation of the excellence in the BSc as one of the top students. He can be contacted at email: yazen.shakir@uoninevah.edu.iq.