Automated University Timetable Generation using Prediction Algorithm

K R Rashmi[1] and Dr. Abhishek M B[2]

[1]M. Tech Scholar, Dayanand Sagar College of Engineering, Bangalore, India
[2]Professor, Dayanand Sagar College of Engineering, Bangalore, India

---

Abstract - *The University Timetable scheduling has been confirmed to be an NP-complete problem and also a strenuous job for educationalist with reference to manual efforts and time consumption and it is a laborious and complex problem. The timetable scheduling problem schedule exams in a specified duration, comprises of scheduling a sequence of classes between students and lecturers in a prescribed time slots and also numerous types of constraints are fulfilled. Several combinatorial optimization and Artificial Intelligence approaches have been proposed to tackle timetabling problem, which contrast each other based on the different deck of limitations. In this paper, automated timetable generation is achieved using a prediction algorithm combining Genetic algorithm and Particle Swarm Optimization algorithm. In the first phase required departmental data is collected, constraints to be considered are defined. In the second phase, timetable is generated using the proposed prediction algorithm.*

Keywords- Artificial Intelligence, Particle swarm optimization, Genetic Algorithm, NP-complete problem, Combinatorial optimization.

1.INTRODUCTION

Based on the requirement the lecture needs, combination of number of lecturers, classes, time slots, and days are arranged by preparing the timetables in the University and the combination is designated as departmental resources, that varies for allocation of timetable. A flawless timetable creates a natural rhythm and routine for productive learning and teaching process. Even now the lecture timetable preparation is repeatedly done by hand because of its built-in complications, while most of the college organization's work has been Automated.

Timetable Scheduling problem normally demand for many persons and days of work if the solution is manual. Moreover, the result attained may be inadequate in certain way; for example, if timetable is scheduled at the same time a student may not be able to take the courses, he/she desires. For the same reason, automated timetabling is given dedicated substantial attention. Vast number of exceptional instances of the timetable scheduling problem have been proposed in the composition, that implies contradictions to one another cantered on the different institutions involving variety of limitations. University timetable Scheduling is specified to perform a collection of tasks considering all the constraints by allocating resources over time, so that input requests are met in a timely and effective way. It is split into two main types:

1.1 University Course timetable Scheduling

Given a prescribed time periods and number of rooms, the university course timetable scheduling problem schedules a collection of classes for each course. Common students can be there in university courses in contrast to the school problem, where disjoint group of students are there in classes of school. Conflict arises, when common students are there in these two courses and scheduling at the alike period is unable. Each course with specified credits is allotted to a professor with prescribed amount of time for the lectures.

1.2 University Examination timetable Scheduling

Within a given amount of time, the examination timetabling problem schedules a given number of exams described as one exam for every course. The course timetable scheduling and examination timetable scheduling are similar, and is difficult to make a clear distinction between the two problems. In reality, both in examination timetabling problem and course timetabling one, some certain problems can be formulated. Allocation of the exams towards restricted set of time intervals by satisfying the highest number of limitations that differentiates considerably from organizations to organizations is essentially known as examination timetable scheduling difficulty. Therefore, the examination timetabling problems differs from their constraints, size and complexity.

2. RELATED WORKS

Department Course timetabling is a N-P complete problem for which the solution can only be achieved at near best level. Perfect solution cannot to be achieved only the optimal solution can be obtained by various traditional and new Artificial Intelligence techniques. This section describes some of the approaches used to tackle the problem.

2.1 Tabu Search Algorithm

Authors D. Nguyen, K. Nguyen, K. Trieu, and N. Tran(2010), have used Tabu search algorithm to automate university timetabling problem [11]. Feasible results are observed in search space which is utilized by Tabu Search

Algorithm. Set of achievable solutions to the problem is known as search space. Author make use of fundamental elements called "Tabu". Tabus exists to move aside along non-improving moves and from local optima to stop cycling. Tabu search predominantly keep away from getting trapped at local maxima[11]. For the same reason, when it is trapped in local optima this search permits non-improving moves. Benefit of Tabu Search procedure restrains cycling back to the earlier results by the use of memories therefore making added possible development, but evaluating resources is expensive and formulating the problem is hard which is the drawback of this approach.

2.2 Graph Colouring Heuristic Method

Authors N. M. Hussin and A. Azlan (2013), have implemented graph colouring heuristic method for constructing stages of process scheduling difficulty [10]. Timetable scheduling difficulty is dealt as graphical representation problem. Events are ordered using specific domain heuristics and then events are assigned subsequently into reasonable time intervals, thus no rules are vandalized for every time slot. Scheduling difficulty is deteriorated to basic portions employing graph colouring method [10]. Node depicts subject and edge depicts conflicts in this graph method. The prime part is that the construction phase which produces an occupant of prime solutions. The next phase is improvement phase where it produces the ultimate best solution. In this method, very prolonged time is taken to resolve a problem and it does not schedule soft constraints.

2.3 Fuzzy Logic Method

Authors W. F. Mahmudy and R. E. Febrita (2017), use Fuzzy logic for achieving and implementing timetable scheduling including various genetic operators incorporated. [8] A Multivalent logic known as Fuzzy logic is used to solve the constraints. This is obtained from fuzzy set theory to alter approximate reasoning instead of precise. The membership values of formal fuzzy logic variables may not be solely 0 or 1, statement's percentage of truth may vary between the range 0 and 1 [8]. Unlike the classic logic, Fuzzy logic is not constrained to two value logic. Results obtained depict that this approach usage can optimize complex scheduling goals and provide a result similar to the real world. More stable situation is reached in less time by using linguistic variables. Membership function evaluation is difficult, hard to create a fuzzy logic model, more calibrate tuning and simulation is required before using for any application. These are some of the major drawbacks of this method.

2.4 Constraint Satisfaction Modelling

Author T. Elsaka (2017), make use of constraint satisfaction modelling for Automated generation timetable [7]. Constraint Programming is focused on the constraints and variables domain rather than the objective function and depends on viability rather than improvisation. Author represents the time table generation procedure that minimizes domains of variables, combined with backtracking search by controlling the constraints along a system of constraints propagation. Constraint programming is significantly a precise statement of the constraints that serves as part of the program, is a vital advantage of this programming. This is disapproving in timetable issues, that makes the program straightforward to alter. Constraints and data are main components in this methodology [7]. Authors Abhishek M B and N. S. V Shet, have described a similar smart data management and monitoring water distribution using cyber physical perspective and have proposed data collection and computation methods which plays a critical role. Struggling to indicate soft constraints, potential complicated problems with intensifying the initial suitable solution and Time consuming are some of the hindrances of this process.

2.5 Genetic algorithm method

Authors S. Ab Saad, F. A. Adnan, W. Z. A. Wan Muhamad and Z. R. Yahya (2018) [4], analyses the utilization of Genetic algorithm in timetable scheduling difficulty by carrying out an inspection. Genetic algorithms are influenced by the procedure of natural selection and is a type of gradual developmental algorithms. The procedure of natural selection begins with the selection of appropriate individuals from a population [2]. Production with the appropriate individuals is established at the end in this procedure while it keeps on iterating. The prime timetabling problem terminates certain dimensions of the problem and consolidates those dimensions into constraints with multitude binary variables diminishing to the sustainable size. Genetic algorithm approach remarkably decreases the individual size with by assembling several binary variables into one gene value. Involving lot of parameters and Time consuming as it gives results with iterative approach are some of the flaws in this approach [4][5][6].

2.6 Hybrid Particle Swarm optimization with local search

Authors D. Apostolou and E. Psarra (2019), proposed an approach using hybrid optimization procedure which sorts out the difficulty of producing timetable of an educational institution. PSO is an artificial intelligence methodology utilized to obtain a significant result in a figure maximized and minimized difficulties [1]. PSO defeats the problem by integrating with local search. This problem is addressed with productive results by collaborating the classic PSO upgradation technique with prototype methodology. Time taken for the optimization procedure is up to the mark which varies from few couple of seconds to few minutes [3]. Merits of this approach is that particles upgrade themselves with the internal velocity and they also have memory, that is supreme to the algorithm. PSO does not have genetic operators like crossover and mutation which is a drawback.

3. PROPOSED FRAMEWORK

Literature review performed, showed that, compared to all other approaches Particle swarm optimization algorithm is better suited for global optimization in timetable scheduling problem but does not have operators for better accuracy in local optimization. Genetic algorithm is better suited for local optimization in timetable scheduling problem with good accuracy but number of iterations is more and lot of time is consumed for computation to achieve global optimization and solution.

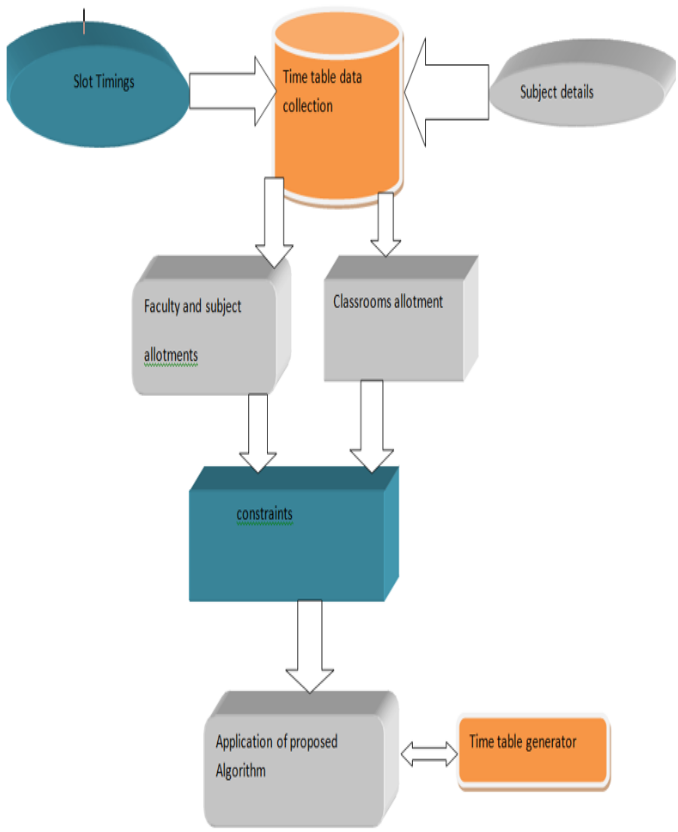


Fig1 – Proposed Framework for timetable generation

The Proposed framework as shown in Fig1 involves application of prediction algorithm with combination of genetic algorithm and particle swarm optimization algorithm for timetable generation. It involves collection of data of instructors, classrooms, subjects, timings as first step, defining constraints for a particular department as second step and generation of timetable using proposed prediction algorithm as final step.

4. Proposed Algorithm for Timetable generation

The data set acts as input to get structured output result. Process model for input output involves considering input parameters, application of proposed algorithm and generating the output timetable. The inputs are updated using a standard Database Browser for SQLite for creating and editing the dataset.

Instructors, rooms, sections, subjects and sharing acts as input parameters. QT and PyQt5 is used for Graphical User Interface for the scheduler. Inputs can be selected, edited and deleted using the GUI as shown below figure. Instructor and rooms names, available hours are selected. Available hours are indicated in green color and unavailable hours are indicated in red color.

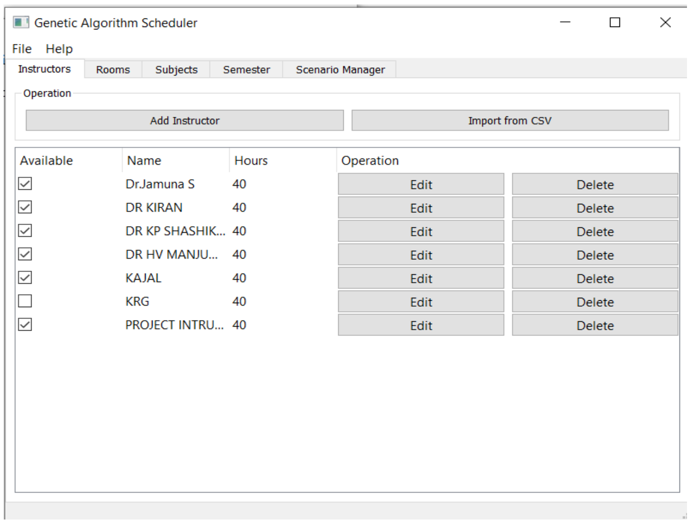


Fig2- Instructor names and available hours per week

Subjects are selected with the name, subject code, related instructor name and type that can be lecture (LEC) or laboratory (LAB). Number of hours that the subjects should be allotted is updated as shown in Fig6. Different number of sections can be selected with the related subjects and rooms.

After the selection of inputs through GUI or using standard database, algorithm settings have to be set with the values which includes starting and ending time, minimum population, maximum population and mutation rate.

Scheduling of lecture classes and examinations in the University involves consideration of a large amount of data and parameters and definition of constraints to meet the essential requirements. Two principal constraints are to be achieved to find the suitable solution for complex timetable scheduling problem. Hard and Soft constraints taken into account for university course scheduling is prescribed below.

Rules that must be fulfilled without any violation are termed as Hard constraints.

Examples for course timetabling include,

- Only one class can be handled by a professor at a given time slot.
- One classroom and one subject are assigned to a professor in the given time.
- Continuous assignment of classes to be avoided and should be scheduled consecutively.

- Time slots to be distributed as per the course credits.
- Appropriate subject and lab sessions assigned to Professors, Assistant Professors and Associate Professors in the department.

Examples for examination timetabling include,

- In a given single timeslot, only one exam to be scheduled when common students are attending the exam in a single classroom.
- Number of students taking the exam in a given classroom should not exceed the limited number of seats in the respective classroom.

Rules that must be satisfied with negligible violation allowed are termed as soft constraints.

Examples for course timetabling,

- Timeslots are allotted in the Professor's favoured time zones.
- Depending on the departments, lunch and short break to be scheduled.
- The practical courses are prioritized in morning session, and the theory courses are scheduled in afternoon session, also considering different lab batches.

As examples of examination timetabling,

- Same Exam cannot be scheduled on two different days and at two different time slots.
- Exams of the courses with more credits to be scheduled first than the courses with less credits.

The automated generation of timetable is achieved by application of proposed prediction algorithm using the combination of genetic algorithm and particle swarm optimization algorithm. The algorithm involves application of genetic algorithm in which the solution of problem is called as chromosomes, which gets modified in each iteration for giving better solution approach for the defined problem. Genetic algorithm has main two operators to achieve the modification in chromosomes which are crossover and mutation. Once the fitness value of chromosome is validated, the next stage is to select the chromosomes with more fitness value for the procedure of reproduction. The selected chromosomes are reproduced using crossover operation which randomly fix a locus and interchange the information between the two parent chromosomes to provide new distinct chromosomes which are more near to the expected solution. Next stage is mutation in which there is a gene swapping in the chromosome randomly to predict the near solution.

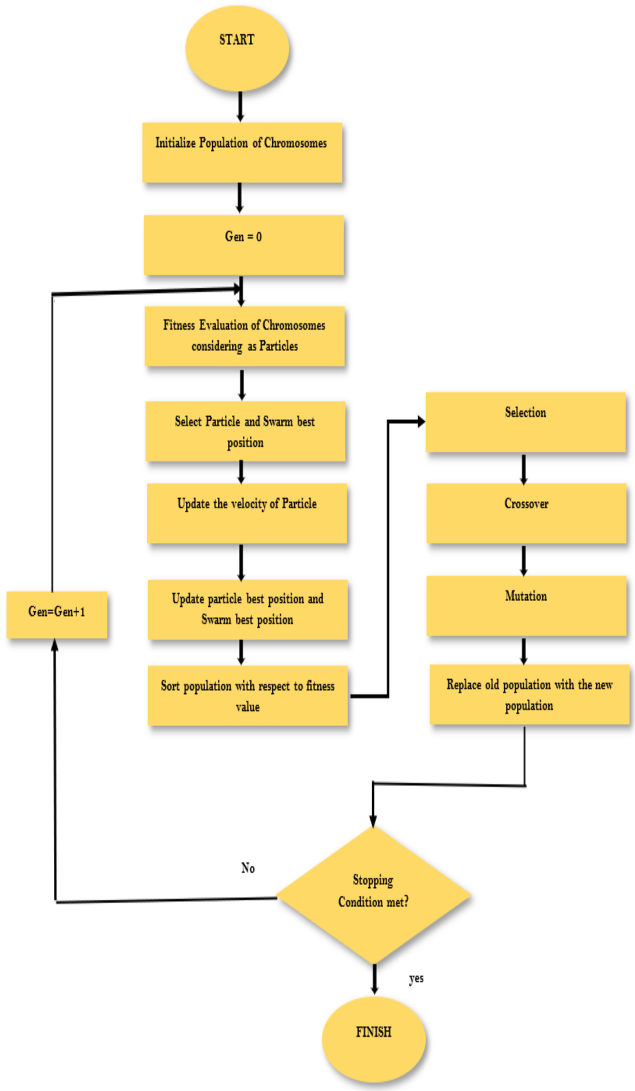


Fig3- Flow chart of Proposed Prediction Algorithm

Particle swarm optimization in contrast has particles similar to chromosomes in genetic algorithm. It is inspired the method of flock of birds searching for food. All particles in the space of search acts as solution to the problem and maintain the values of their personal best position and global best position to achieve the correct direction and velocity for optimal solution.

Proposed Algorithm initializes population with chromosomes randomly, these chromosomes are passed as particles for particle swarm optimization algorithm for fitness calculation. Fitness is calculated based on the constraint's satisfaction defined. Based on the fitness, the particle position and velocity is updated and later sent to genetic algorithm operators like selection, crossover and mutation for further optimization.

Once the fitness calculation is completed, the particles update their velocity based on particle and swarm best position using below formula:

v = wv + c1 * rand * (pBest – p) + c2 * rand * (gBest – p)

where v is the new velocity, w is the inertia to keep the particle in same direction, c1 and c2 are acceleration constants for self-best and swarm best positions respectively, pBest is particle best position and global best position. Based on the fitness value the population is sorted and further is selected using tournament selection for reproduction. Selected chromosomes are sent to crossover and mutation to get the most suitable solution to the problem.

5. RESULTS AND DISCUSSION

Timetable is generated using prediction algorithm with the combination of genetic algorithm and particle swarm optimization algorithm meeting all defined hard constraints and most of the soft constraints. Subjects are allocated consecutively without any clashes. Most of the slots are allotted with maximum priority to subject placement as represented in generated output in Fig4.

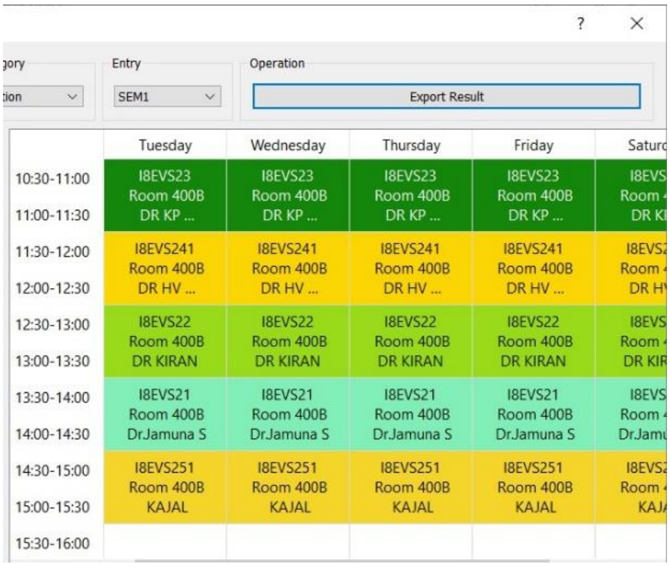
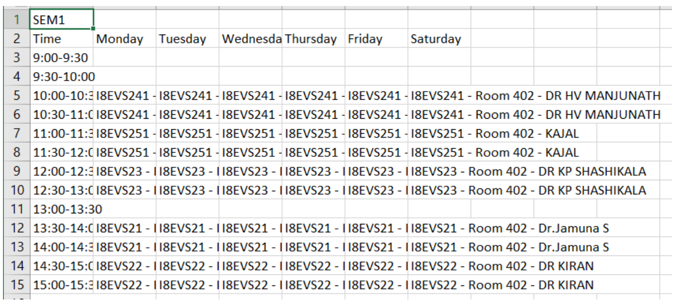


Fig4-Generated timetable

Generated timetable is exported in CSV format as shown in Fig 5.



1	SEM1						
2	Time	Monday	Tuesday	Wednesda	Thursday	Friday	Saturday
3	9:00-9:30						
4	9:30-10:00						
5	10:00-10:3	I8EVS241 -	I8EVS241 -	I8EVS241 -	I8EVS241 -	I8EVS241 -	I8EVS241 - Room 402 - DR HV MANJUNATH
6	10:30-11:0	I8EVS241 -	I8EVS241 -	I8EVS241 -	I8EVS241 -	I8EVS241 -	I8EVS241 - Room 402 - DR HV MANJUNATH
7	11:00-11:3	I8EVS251 -	I8EVS251 -	I8EVS251 -	I8EVS251 -	I8EVS251 -	I8EVS251 - Room 402 - KAJAL
8	11:30-12:0	I8EVS251 -	I8EVS251 -	I8EVS251 -	I8EVS251 -	I8EVS251 -	I8EVS251 - Room 402 - KAJAL
9	12:00-12:3	I8EVS23 - I	I8EVS23 - I	I8EVS23 - I	I8EVS23 - I	I8EVS23 - I	I8EVS23 - Room 402 - DR KP SHASHIKALA
10	12:30-13:0	I8EVS23 - I	I8EVS23 - I	I8EVS23 - I	I8EVS23 - I	I8EVS23 - I	I8EVS23 - Room 402 - DR KP SHASHIKALA
11	13:00-13:30						
12	13:30-14:0	I8EVS21 - I	I8EVS21 - I	I8EVS21 - I	I8EVS21 - I	I8EVS21 - I	I8EVS21 - Room 402 - Dr.Jamuna S
13	14:00-14:3	I8EVS21 - I	I8EVS21 - I	I8EVS21 - I	I8EVS21 - I	I8EVS21 - I	I8EVS21 - Room 402 - Dr.Jamuna S
14	14:30-15:0	I8EVS22 - I	I8EVS22 - I	I8EVS22 - I	I8EVS22 - I	I8EVS22 - I	I8EVS22 - Room 402 - DR KIRAN
15	15:00-15:3	I8EVS22 - I	I8EVS22 - I	I8EVS22 - I	I8EVS22 - I	I8EVS22 - I	I8EVS22 - Room 402 - DR KIRAN

Fig 5-Generated timetable in CSV format

Results show that generated timetable is free from clashes and prediction algorithm shows improvised results with stable output and improved global optimization.

6. CONCLUSION

The outcomes have shown that the framework can give substantial arrangements that can be utilized. There are still situations that would require the administrator to change a few information to make an ideal arrangement. The framework was likewise intended to be straightforward and direct. This disposes of any turmoil brought about by dispersed UI controls and makes utilization of the product completely used. The straightforwardness of the framework and presentation of configurable calculation's objective and execution decreased the requirement for such a lot of limitations as arrangements are made powerfully. Automated timetable generation enhances the quality of the education institutes and tends to greatly reduce time and manual efforts. The Prediction Algorithm uses the advantages of the two Artificial Intelligence algorithms to generate more valid timetable output aiming to provide maximum allotment.

7. FUTURE WORK

Automated timetable generation can be further improved by using it in a management system to further automate the notification system in the campus. The generation can be done using a mobile application or can be generated in a mini hardware computer like Raspberry Pi and further can be used to notify the faculties about the generated timetable. Timetable management system has been carried out using RFID and Biometric, but to improve the security latest machine learning algorithms can be used for real time face detection and recognition of faculties and for sending notification based on the stored database.

REFERENCES

[1] E. Psarra and D. Apostolou, “Timetable scheduling using a hybrid particle swarm optimization with local search approach,” in 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA), 2019.

[2] A. Aminu et al., “Design and implementation of an automatic examination timetable generation and invigilation scheduling system using genetic algorithm,” in 2nd International Conference on Applied Engineering (ICAE), 2019.

[3] T. W. Ekanayake, P. Subasinghe, S. Ragel, A. Gamage, and S. Attanayaka, “Intelligent timetable scheduler: A comparison of genetic, graph coloring, heuristic and iterated local search algorithms,” in 2019 International Conference on Advancements in Computing (ICAC), 2019.

[4] F. D. Wihartiko, H. Wijayanti, and F. Virgantari, “Performance comparison of genetic algorithms and particle swarm optimization for model integer

programming bus timetabling problem," IOP Conf. Ser. Mater. Sci. Eng., vol. 332, p. 012020, 2018.

[5] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview," Soft Comput., vol. 22, no. 2, pp. 387–408, 2018.

[6] K. Y. Junn, J. H. Obit, and R. Alfred, "The study of genetic algorithm approach to solving university course timetabling problem," in Lecture Notes in Electrical Engineering, Singapore: Springer Singapore, 2018, pp. 454–463.

[7] T. Elsaka, "Autonomous generation of conflict-free examination timetable using constraint satisfaction modelling," in 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), 2017.

[8] R. E. Febrita and W. F. Mahmudy, "Modified genetic algorithm for high school time-table scheduling with fuzzy time window," in 2017 International Conference on Sustainable Information Engineering and Technology (SIET), 2017.

[9] J. Soyemi, J. Akinode, and S. Oloruntoba, "Electronic lecture time-table scheduler using genetic algorithm," in 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology (DASC/PiCom/DataCom/CyberSciTech), 2017.

[10] A. Azlan and N. M. Hussin, "Implementing graph coloring heuristic in construction phase of curriculum-based course timetabling problem," in 2013 IEEE Symposium on Computers & Informatics (ISCI), 2013.

[11] K. Nguyen, D. Nguyen, K. Trieu, and N. Tran, "Automating a real-world university timetabling problem with Tabu search algorithm," in 2010 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2010.