IOP Conference Series: Materials
Science and Engineering

PAPER • OPEN ACCESS

Application of genetic algorithm concept on course scheduling

To cite this article: R Ansari and N Saubari 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **821** 012043

View the article online for updates and enhancements.

IOP Conf. Series: Materials Science and Engineering **821** (2020) 012043 doi:10.1088/1757-899X/821/1/012043

Application of genetic algorithm concept on course scheduling

R Ansari[1, *] and N Saubari[2]

[1]Departement of Informatics Engineering, Faculty of Engineering, Universitas Muhammadiyah Banjarmasin, Banjarmasin, Indonesian

[*]Email: rudy@umbjm.ac.id

Abstract. Genetic Algorithms have parameters such as crossover probabilities and mutation probabilities used based on entering population numbers and number of generations. Of the two entries, nine rules were obtained which would produce crossover probabilities and mutation probabilities. One problem that can be solved using a genetic algorithm is the scheduling of courses. In the preparation of course scheduling, it takes quite a long time and needs a very high accuracy. Therefore, the purpose of this study is to implement genetic algorithms on lecture scheduling problems. So that the accuracy and speed in determining the class schedule can be fulfilled. The test results show that applying a genetic algorithm can obtain the course schedule without any collision in one iteration process.

1. Introduction

In the process of lecturing activities, scheduling lecture activities in a university is a complicated matter and often has difficulty in allocating courses, lecturers and rooms to be used for lectures, to make sure that there is no conflict with the schedule of lectures, lecturers and other rooms in one lecture schedule period. Besides, changes in the schedule of lecturers often make it difficult in allocating time and searching for empty rooms that will be used to schedule temporary changes. In this case, the planned lecture activities should be carried out according to a predetermined schedule if it can be managed properly [1, 2].

The problems that are felt when preparing the lecture schedule are the difficulty in allocating courses with the lecturer and his room so that there is no clash and the number of college participants who adjust the capacity of the classroom. Besides, there is also a change in the temporary lecture schedule because the lecturer is unable to attend, often having difficulty in finding empty rooms for surrogate courses [3], [4]. Based on these problems, it is necessary to apply the algorithm in the preparation of the lecture schedule to facilitate the schedule creation process.

2. Literature review

2.1. Scheduling

Scheduling is the activity of allocating resources or machines that exist to run a set of tasks within a certain period of time. According to Ambuhi, scheduling has 2 meanings, namely:

a. Scheduling as a function of decision making. Scheduling is a process for determining a schedule
b. Scheduling is a theory. Scheduling is a collection of principles, models, techniques and logical conclusions in decision making. Scheduling is part of the shop floor control

According to Ambuhi, if scheduling is constant, then the proper work order will reduce the flow time and average work in the process. Decisions made in scheduling include sequencing, starting time and working time, a sequence of operations for routing (routing). Scheduling problems often arise if there is a set of tasks simultaneously, while the equipment is limited [3].

A more general definition of scheduling is to assign a set of events (lectures) with a limited set of sources from time to time to meet the defined constraints (constraints). These constraints can be categorized as hard constraints and soft constraints, where hard constraints have a higher priority than soft constraints. There are two limitations in the preparation of lecture schedules, namely: hard constraints (must be fulfilled) and soft constraints (attempted to be fulfilled). Hard constraints are boundaries that must be applied to scheduling courses and must be fulfilled. A solution can only be said to be valid if the solution has no violated hard constraint [5, 6].

Common hard constraints in scheduling courses are as follows:

a. A lecturer can only teach courses for one location at a certain time.
b. A student can only attend classes for one location at a time.

In contrast to hard constraints, soft constraints are constraints that cannot always be fulfilled in the process of forming schedules, but even if they do not have to be fulfilled, the resulting schedule must try as much as possible to meet the soft constraint provisions. Examples of soft constraints in scheduling courses are:

a. Lecturers can ask for the specific teaching time they want.
b. Placement of the schedule for the time requested by the lecturer is adjusted to the priorities of the lecturer[7-9].

2.2. Genetic algorithm

The genetic algorithm is a heuristic search algorithm based on the mechanism of biological evolution. In this algorithm, search techniques are carried out at the same time on some possible solutions, in other words, this algorithm can help provide the best and fastest solution to a problem that has many possible solutions known as the population. Individuals in one population are called chromosomes. This chromosome is a solution that is still in the form of a symbol. The initial population is built randomly, while the next population is the result of the evolution of chromosomes through an iteration called generation [10-13].

Components of Genetic Algorithms. There are 6 main components contained in the genetic algorithm, namely:

1. Coding Techniques
2. Initialization Procedure (generate initial population)
3. Evaluation Function
4. Selection
5. Genetic Operators

3. Research method

The steps in the process of making a scheduling using genetic algorithms can be seen in Figure 1. The chromosome used in producing the initial individual in this system is as follows:

Schedule chromosomes contain information:

1) list of courses (m),
2) room (r),
3) time (w),

The schedule chromosome array structure on the new chromosome is: [m, r, w]. The list of course genes includes data on course codes, names of courses, credits, lecturers and study program names. The room gene includes room name and room capacity. While for the time gene includes hours and days.

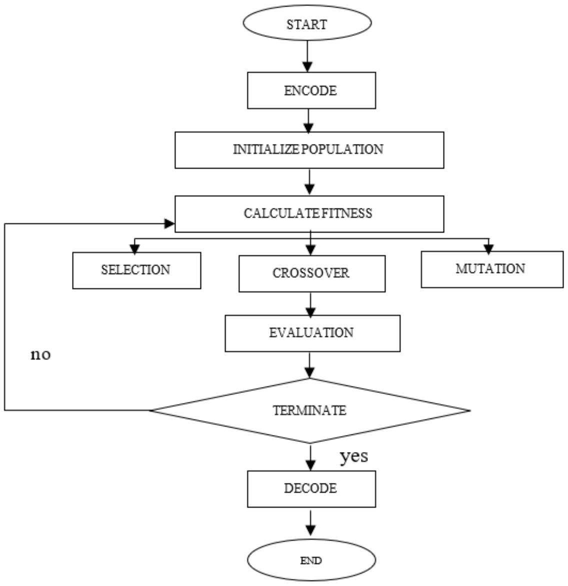

Figure 1. Process of making a scheduling using genetic algorithms

4. Result and discussion

The application of Genetic Algorithms begins with forming the initial population, scheduling limits, soft constraints, GA components. The GA component is divided into the following steps:

4.1. Coding technique

Genes presented here are in the form of arrays of real numbers. The parameters used in each chromosome are a list of courses (m), room (r) and time (w). So that in each one chromosome is filled with data [(m r, w)].

4.2. Initialization of initial population

In the initialization of initial population process, it generates a population randomly, for example chromosomes that are generated as a schedule that will be generated are 4 chromosomes in 1 generation by using 10 experimental data of list of courses (m) and time (w), from the informatics study program.

4.3. Chromosomes evaluation

Chromosome evaluation is a step to calculate the fitness value of a chromosome that was initialized in the previous process. To determine the fitness value, the scheduling value generated by the fitness function will represent how many requirements are violated, so that in the case of scheduling lecture, the smaller the number of violations produced, the better the solution will be. For violations that occur if there is a conflict on lecturers, a value of +5 will be given and conflict on rooms will be given a +1 value. To avoid an infinite fitness value, the total number of all violations will be added by 1. The best standard fitness value for this scheduling is 1 where fitness on the chromosome = 1, means that the optimal solution results obtained were no conflict found on a schedule.
Calculate fitness value with the chromosome formula, (n) = 1 + (BBD + BBR)
Fk1 : 1 + 1 (+5) + 2 (+1) = 8
Fk2 : 1 + 1 (+5) + 1 (+1) = 7
Fk3 : 1 + 2 (+5) + 1 (+1) = 12
Fk4 : 1 + 1 (+5) + 1 (+1) = 7
Total Fitness = Fk1 + Fk2 + Fk3+ Fk4 = 8 + 7 + 12 + 7 = 34

IOP Conf. Series: Materials Science and Engineering **821** (2020) 012043 doi:10.1088/1757-899X/821/1/012043

Description:
BBD = Total conflict on Lecturers
BBR = Total conflict on room
n = Chromosome Sequence
Fk = Chromosome Fitness

4.4. Selection
This selection stage is selecting the best chromosome where the most suitable chromosomes that have a higher probability are more likely to be selected for the next generation. To calculate the probability value of each chromosome can be seen as follows:
Pk1 = Fk1/(Total fitness) = 8/34 = 0,23
Pk2 = Fk2/(Total fitness) = 7/34 = 0,21
Pk3 = Fk3/(Total fitness) = 12/34 = 0,35
Pk4 = Fk4/(Total fitness) = 7/34 = 0,21
Total Probability = Pk1 + Pk2 + Pk3 + Pk4 = 0,23 + 0,21 + 0,35 + 0,21 = 1
Description:
Pk = Chromosome Probability
Fk = Chromosome Fitness
Chromosome 1 = [(1,1,7),(2,3,2),(3,4,1),(4,5,9),(5,2,19),(6,6,1),(7,3,7),(8,3,8), (9,3,7),(10,2,13)].
Chromosome 2 = [(1,1,7),(2,3,2),(3,4,1),(4,5,9),(5,2,19),(6,6,1),(7,3,7),(8,3,8), (9,3,7),(10,2,13)].
Chromosome 3 = [(1,3,1),(2,5,15),(3,3,2),(4,2,16),(5,1,2),(6,4,15),(7,6,1),(8,2,3), (9,5,19),(10,4,13)].
Chromosome 4 = [(1,2,19),(2,1,2),(3,3,1),(4,5,20),(5,6,3),(6,3,1),(7,1,20),(8,4,2), (9,5,15),(10,5,15)].

4.5. Crossover
At this stage, a crossover is done by crossing two chromosomes. This technique uses the one cut point method, which is to randomly choose a position of the parent chromosome then exchange sub-chromosomes. In this process, the probability of a crossover (Pc) will control the crossover operator in each generation in the crossover population. the probability of crossover (pc) used is 75% or 0.75. The greater the value (Pc) the faster the new chromosome will appear in the population, but if (Pc) is too large then the chromosomes that are candidates for the best solution may disappear faster in the next generation. If the random number generated is smaller than the crossover (Pc) probability value, a crossover will be done.
Generates random numbers in the range of (0 - 1).
Chromosome 1 = 0,65
Chromosome 2 = 0,81
Chromosome 3 = 0,50
Chromosome 4 = 0,69

Thus, the selected parent will be crossed, namely chromosome 1, chromosome 3 and chromosome 4. the rules of crossing are done randomly (random) following the parents who have been previously selected. The process of this crossing can be seen as follows
Chromosome 1 >< Chromosome 3
Chromosome 3 >< Chromosome 4
Chromosome 4 >< Chromosome 1

4.6. Mutation
From the calculation result, the best fitness value for the new population in one generation is obtained, chromosome 4 as the best schedule because it has a fitness value that is close to 1. The following is the result of schedule output from the best chromosome, chromosome 4, can be seen in the following table (Table 1):

Table 1. Schedule output chromosome

No	Day	Time	Course Codes	Courses	Credits	Lecturer	Class Room	Capacity
1	Senin	08.50-10.30	UMB0002	Pend. Kewarganegaraan	2	Harpani	FT7	30
2	Senin	08.00-10.30	INF6242	Pemrog. Perangkat Mobile	3	Wahyu Ridhoni, M.Kom	Lab Dasar	30
3	Senin	10.30-13.50	INF4223	Sistem Operasi	3	Kamarudin, M.Kom	Lab Dasar	30
4	Senin	08.00-10.30	INF4221	Metode Numerik	3	Mukhaimy Gazali, M.Si	Lab Jaringan	40
5	Senin	08.00-10.30	INF4222	Sistem Pendukung Keputusan	3	Rudy Ansari, M.Kom	FT1	60
6	Selasa	10.30-13.50	INF4220	Interaksi Manusia & Komputer	3	Nahdi Saubari, M.Kom	FT1	60
7	Rabu	03.00-15.30	INF4219	Ilmu Kealaman Dasar	3	Windarsyah, M.Kom	FT2	30
8	Kamis	08.50-11.20	INF2209	Algoritma & Struktur Data	3	Munsyi. S.Kom, M.T	FT1	60

5. Conclusion and recommendation

5.1. Conclusion

Scheduling at Informatics Study Program, Muhammadiyah Banjarmasin University, has implemented the GA method, from the results obtained there is still a conflict of rooms from chromosome 4, this is because the number of initial chromosomes and generation generated is small. If the initial population and generation are increased, then there will be more variations on chromosome and will lead to the most optimal solution, namely the chromosome with a fitness value = 1, which indicates that there is no conflict of schedules.

5.2. Recommendation

There should be experiments using other techniques in selection operators, crossover and mutation. This is to compare the use of techniques for each operator so the most optimal technique is found.

References

[1] Nugraha T A, Putra K T and Hayati N 2017 university course timetabling with genetic algorithm: a case study *J. Electr. Technol. UMY* vol **1** no 2 pp 100–5.
[2] Mauren A O, Gudo C O and Oanda I O 2011 University expansion in Kenya and issues of quality education: Challenges and opportunities *Int. J. Bus. Soc. Sci.* vol **2** no 20 pp 203–15.
[3] Ambühl C, Mastrolilli M, Mutsanas N and Svensson O 2011 On the approximability of single-machine scheduling with precedence constraints *Math. Oper. Res.* vol **36** no 4 pp 653–69.
[4] Ouelhadj D and Petrovic S 2009 A survey of dynamic scheduling in manufacturing systems *J. Sched.* vol **12** no 4 pp 417–31.
[5] Xu Y, Zeng, Liu Q and Wang X 2014 A genetic algorithm based multilevel association rules mining for big datasets *Math. Probl. Eng.*vol **2014** pp 1–9.
[6] Jouya M and Khayati S 2018 *Review local search algorithms in artificial intelligence* vol **4** no 2 pp 81–6.
[7] Science C and Engineering S 2017 *Automatic Time Table Generator 1* vol **7** no 5 pp 204–11.
[8] Duan K, Fong S, Siu S W I, Song W and Id S S G 2018 *SS symmetry Adaptive Incremental*

Genetic Algorithm for Task Scheduling in Cloud Environments pp 1–13.

[9] Sani H M 2016 *Solving Timetabling problems using Genetic Algorithm Technique* vol **134** no 15 pp 33–8.

[10] Shinde S and Gurav S 2018 *Automatic Timetable Generation using Genetic Algorithm* vol **9** no 4 pp 19–21.

[11] Yadav S L and Sohal A 2017 Comparative study of different selection techniques in genetic algorithm *Int. J. Sci. Math. Educ.*vol **6** no 3 pp 251–54.

[12] Salvi A, Khanvilkar O and Balkhande B W 2016 *Automatic Time-Table Generation System using Genetic Algorithm* vol **5** no 3 pp 824–26.

[13] Sari Y, Alkaff M and Pramunendar R A 2018 Iris recognition based on distance similarity and PCA *AIP Conf. Proc.* vol **1977**.