International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-5C, May 2019

Lecture Scheduling Automation Using Genetic Algorithm

Herman, Lukman Syafie, Dirgahayu Lantara

ABSTRACT--- ***This paper aims produce an academic scheduling system using Genetic Algorithm (GA) to solve the academic schedule. Factors to consider in academic scheduling are the lecture to be held, the available room, the lecturers and the time of the lecturer, the suitability of the credits with the time of the lecture, and perhaps also the time of Friday prayers, and so forth. Genetic Algorithms can provide the best solution for some solutions in dealing with scheduling problems. Based on the test results, the resulting system can automate the scheduling of lectures properly. Determination of parameter values in Genetic Algorithm also gives effect in producing the solution of lecture schedule.***

Keywords: Scheduling; Genetic algorithm; Scheduling automation.

I. INTRODUCTION

Scheduling is the process of allocating resources to run a set of tasks over a period of time. One form of scheduling that is often done is scheduling lectures in a university. Lecture scheduling is one of the most important parts supporting lecturing process. Some factors that must be considered in the scheduling of lectures are lecturers, rooms, and time [1]. In addition to these factors, in lecture scheduling should also be noted the suitability of the credits of the subjects with the length of time lectures. The availability of lecturers' time for teaching and Friday prayer times also needs to be a concern in the preparation of lecture schedules. Because the lecturer usually has a problem of teaching at a certain time, especially a lecturer who has a structural position and remember the location of the study is Informatics Engineering Program University Muslim Indonesia which is an Islamic university. Looking at these factors, it can be ascertained that it will take a long time to do lecturing scheduling if done manually because it requires high precision.

One of the algorithms in computer science that solves the scheduling problem is the Genetic Algorithm (GA). The GA is a powerful optimization algorithm that has the ability to navigate the search space well [2]–[4]. Along with the development of the informatics field, GA is applied to a variety of issues, such as forecasting, optimization, scheduling, and so on. In the soft computing field, GA is widely used to obtain optimal parameter values [5]–[7].

Some researchers have conducted research regarding scheduling using GA, including Reballo and Casella and Pandey [8]. However, both have not shown the results of a study that adjusts between the number of credits of courses and the time allocation that is prepared and also the time of lecturers' willingness to teach. Hariyadi et al, implements GA in academic scheduling. In the study, lecturer time has been considered [1]. In this research, besides the time of lecturers, the lecture credit and the Friday prayer time are also considered in the preparation of the lecture schedule.

The next part of this paper is: Section 2 describes the GA stages in scheduling lectures. Section 3 describes the test results and discussion. Section 4 gives a conclusion to the research that has been done.

II. EXPERIMENTAL DETAILS

The scheduling automation system that is built consists of 3 parts, namely: data collection or input data needed into the system and stored in the database, then the scheduling processing section using Genetic Algorithm, and the last is to display the results of the scheduling process. The description of the scheduling automation system can be seen in Figure 1.

The process of automation scheduling lectures on the system built consists of 7 parts: the process of data collection from the database, the initialization process, the fitness value evaluation process, the selection process based on the fitness value, the crossover process, the mutation process, and the last is storing the scheduling result on the base data.

Initialization is the determination of the form or component of each individual compiler. In the case of scheduling, the individual is an entity full of schedules within a week. Individuals consist of several genes, in which the number of individual genes depends on the number of classes in a week. Associated with the class, the means for its formers are classes, subjects, and lecturers of course subjects. The individual forms and genes can be seen in Figure 2.

Manuscript published on 30 May 2019.
***** Correspondence Author (s)
Herman, Faculty of Computer Science, Universitas Muslim Indonesia, Makassar, Indonesia. (email: herman@umi.ac.id)
Lukman Syafie, Faculty of Computer Science, Universitas Muslim Indonesia, Makassar, Indonesia.
Dirgahayu Lantara, Faculty of Industrial Engineering, Universitas Muslim Indonesia.

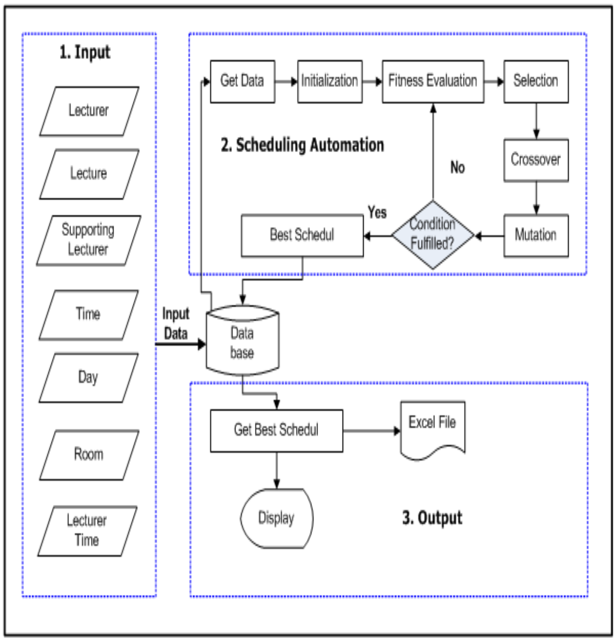


Fig. 1: The flow of the lecture scheduling automation system

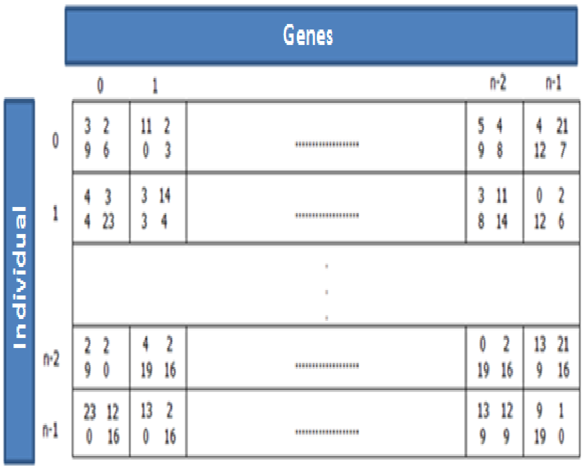


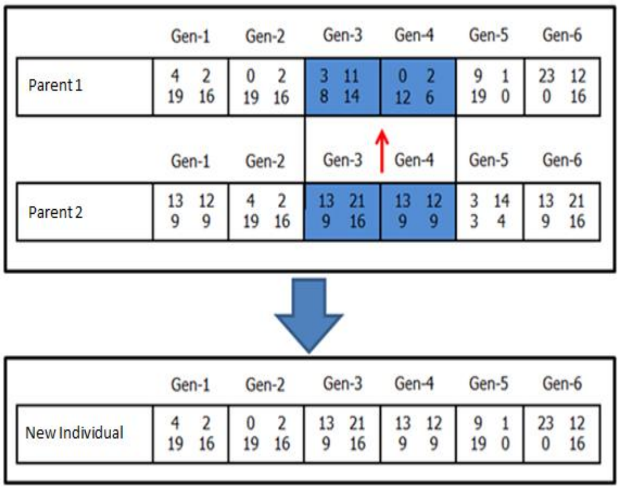
Fig. 2: Individuals and gene constituents

In Figure 2 the gene constituent consists of 4 values. It represents the hours, days, rooms, and lecturers of course subjects. Especially for the hour, adjusted for the number of credits from the course. Once the individuals are formed in the population, the next is to evaluate the fitness value. The fitness equation can be expressed as (1).

$$\text{Fitness} = \frac{1}{1+num\_crash} \quad (1)$$

The more lecture schedules that hit someone, the individual fitness value will be smaller. Lecturer time in fitness calculation. If the gene time is equal to the time the lecturer does not want it, then it is considered as a scheduling collision, as well as Friday prayers. In this study, Friday prayer time starts from 12:00 to 13:00.

The selection process is done by doing the friction based on the highest fitness value. Individuals are in the top rankings and have a greater chance of being selected in the next generation. Based on the selection results, then selected individuals will be used in the crossover. From the crossover process, generate new individuals for the population in the next generation. Figure 3 shows the crossover process. Furthermore, the mutation process is done by replacing the contents of one gene. The Genetic Algorithm process will continue to recur according to the number of generations specified.



III. RESULT AND DISCUSSION

In conducting the test, 2 exploratory forms were performed, based on Genetic Algorithm parameter values and Genetic Algorithm test results. In the exploration test of Genetic Algorithm value values, firstly 2 tests were conducted. The first test is to determine the population number of 10, the number of generations of 10, the probability of crossover is 0.70, and the mutation probability of 0.40.

The first test result did not find the solution. Then proceed with the second test. In the second, test the population number is changed to 100. The results of the second test get a solution which means getting the best lecture schedule.

Based on the first and second tests, it can be deduced that the number of populations gives effect in generating the solution. Then the next test is continued by exploring the part of the probability value of crossover and mutation. A total of 10 tests in this exploration. The test results are shown in Table 1. In the second test, a schedule solution was obtained, but in the third test using the same parameter values, did not get a solution.

Table 1: The results of crossover and mutation exploration testing

Crossover	Mutation	Population	Generation	Result
0.10	0.10	100	10	NS
0.20	0.20	100	10	S
0.20	0.20	100	10	NS
0.40	0.40	100	10	S
0.50	0.50	100	10	S
0.10	0.10	100	20	S
0.20	0.20	100	20	S
0.30	0.30	100	20	S
0.40	0.40	100	20	S
0.50	0.50	100	20	S

The difference in results obtained in the second and third tests using the same parameter values, probably caused by the initial individuals generated randomly because the number of generations used is 10 means only 10 times iterations to get the solution. Then the problem is also the selection process. When considered the selection process described in the previous chapter, it is clear that the lack of methods used cause of the lack of the best individuals who have a high fitness.

Next is to test the results of Genetic Algorithm. The test results aim to see the results of the Genetic Algorithm in lecture scheduling. There are 3 parts of the attention of this result is the suitability between the credits with the time range, the suitability between the schedule of lectures with the Friday prayer time and the time lecturers are available.

In Figure 4, it appears that the suitability of the lecture credit with the time range is appropriate. It can be seen that the courses that have 3 credits get the time range of 50 minutes x 3. In Figure 5 it shows that there is no schedule that collides with Friday prayer time. This is as expected. Next test the lecturers time are available to teach. First is to determine the unwilling time of the lecturer. For example, the lecturers on behalf of Herman and Tasrif Hasanuddin are unwilling to teach on Wednesdays at 10:00 to 13:00 and Thursday from 08:00 to 12:00. The test results are shown in Figure,

#	Day	Sess.	Time	Lecture	Credit	Sem.	Class	Lecturer	Room
1	Monday	(2-3)	08.50-10.30	Struktur Data	2	2	A2	Herman, S.Kom.,M.Cs	U3.11
2	Monday	(2-4)	08.50-11.20	Sistem Informasi Manajemen	3	2	B4	Lilis Nurhayati, S.Kom., M.Eng	U3.05
3	Monday	(3-5)	09.40-12.10	Sistem Digital	3	2	A3	Tasrif Hasanuddin, S.T.,M.Cs	U3.06
4	Monday	(4-6)	10.30-13.00	Sistem Informasi Manajemen	3	2	B2	Nia Kurniati, S.Kom., M.Kom	U3.04
5	Monday	(4-6)	10.30-13.00	Sistem Digital	3	2	A6	Dolly Indra, S.Kom., M.Si	U3.02
6	Monday	(4-6)	10.30-13.00	Sistem Digital	3	2	A1	Farniwati Fattah, S.T.,M.T	U3.01
7	Monday	(8-9)	13.50-15.30	Struktur Data	2	2	B1	Lutfi Budi Ilmawan,S.Kom.,M.Cs	U3.03
8	Monday	(9-11)	14.40-17.10	Rekayasa Perangkat Lunak	3	4	C1	Muh. Aliyazied Mude, S.Kom., M.Kom	U3.09
9	Tuesday	(2-4)	08.50-11.20	Sistem Informasi Manajemen	3	2	A6	Syahrul Mubarak, S.Kom	U3.12
10	Tuesday	(2-4)	08.50-11.20	Sistem Informasi Manajemen	3	2	B1	Nia Kurniati, S.Kom., M.Kom	U3.01

Fig. 4: Schedule of results of Genetic Algorithm

Friday	(1-2)	08.00-09.30	Struktur Data
Friday	(8-10)	13.50-16.20	Sistem Informasi Manajemen
Friday	(9-10)	14.40-16.20	Struktur Data
Friday	(9-11)	14.40-17.10	Sistem Digital

Fig. 5: The lecture schedule considers the Friday prayer time

Thrusday	(2-4)	08.50-11.20	Sistem Digital
Thrusday	(2-4)	08.50-11.20	Sistem Digital
Thrusday	(5-6)	11.20-13.00	Struktur Data
Thrusday	(5-6)	11.20-13.00	Syariah
Thrusday	(7-8)	13.00-14.40	Struktur Data
Thrusday	(9-10)	14.40-16.20	Struktur Data
Wednesday	(2-4)	08.50-11.20	Sistem Digital
Wednesday	(3-5)	09.40-12.10	Sistem Informasi Manajemen
Wednesday	(4-5)	10.30-12.10	Struktur Data
Wednesday	(4-6)	10.30-13.00	Sistem Digital
Wednesday	(7-8)	13.00-14.40	Struktur Data
Wednesday	(7-9)	13.00-15.30	Sistem Informasi Manajemen
Wednesday	(8-10)	13.50-16.20	Sistem Digital

Fig. 6: Schedule of lecture considering the time of the lecturer

IV. CONCLUSION

Based on the test results, the number of population and the number of generations can have an effect on the Genetic

Algorithm in getting the best lecture schedule solution. The results of lecture scheduling using Genetic Algorithm can provide a lecture schedule that adjusts between the number of credits with the time range and the fit between the Friday prayer time with the lecture time as well as the time lecturers available for teaching.

V. ACKNOWLEDGEMENT

The authors would like to thank the parties who have helped so that this research can be completed. The first to the Institute of Research and Development Resources (LP2S) University Muslim Indonesia Makassar which finance this research and then to the Study Program of Informatics Engineering Faculty of Computer Science which provides data

REFERENCES

1. H. P. Hariyadi, T. Widiyaningtyas, M. Z. Arifin, and S. Sendari, “Implementation of Genetic Algorithm to academic scheduling system,” in 2016 IEEE Region 10 Conference, 2016, pp. 2013–2016.

2. H. M. Pandey, "Solving lecture time tabling problem using GA," in 2016 6th International Conference-Cloud System and Big Data Engineering, 2016, pp. 45–50.
3. K. Shashank and M. Balachandra, "Adopting genetic algorithm to develop a neural network for recognition of network intrusion," Int. J. Eng. Technol., vol. 7, no. 4, pp. 5869–5873, 2018.
4. S. Marrouchi, M. Ben Hessine, and S. Chebbi, "New Strategy Based on Combined Use of Genetic Algorithm and Gradient to Solve the UC Problem: Theoretical Investigation and Comparative Study," Int. J. Eng. Technol., vol. 7, no. 3.13, pp. 44–50, 2018.
5. S. Parera, H. T. Sukmana, and L. K. Wardhani, "Application of genetic algorithm for class scheduling (Case study: Faculty of science and technology UIN Jakarta)," in 2016 4th International Conference on Cyber and IT Service Management, 2016, pp. 1–5.
6. Haviluddin and R. Alfred, "A genetic-based backpropagation neural network for forecasting in time-series data," in Proceedings - 2015 International Conference on Science in Information Technology: Big Data Spectrum for Future Information Economy, ICSITech 2015, 2016.
7. Haviluddin, R. Alfred, J. H. Obit, M. H. A. Hijazi, and A. A. A. Ibrahim, "A performance comparison of statistical and machine learning techniques in learning time series data," Adv. Sci. Lett., 2015.
8. F. J. Reballo and I. R. S. Casella, "Mobile application for residential energy consumption scheduling employing ga," in 2016 IEEE International Symposium on Consumer Electronics, 2016, pp. 89–90.

Retrieval Number: E12100585C19/2019©BEIESP
DOI:10.35940/ijeat.E1210.0585C19
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)