Saudi Journal of Engineering and Technology (SJEAT)
Scholars Middle East Publishers
Dubai, United Arab Emirates
Website: http://scholarsmepub.com/

ISSN 2415-6272 (Print)
ISSN 2415-6264 (Online)

Optimizing Scheduling Criteria for Final Project Using Genetic Algorithms

Alyauma Hajjah[1*], Dwi Oktarina[2], Deny jollyta[1]
[1]Department Technical of Informatics, Sekolah Tinggi Ilmu Komputer Pelita Indonesia, Pekanbaru, Indonesia
[2]Department Informatics of System, Sekolah Tinggi Ilmu Komputer Pelita Indonesia, Pekanbaru, Indonesia

Original Research Article

***Corresponding author**
Alyauma Hajjah

Article History
Received: 05.09.2018
Accepted: 15.09.2018
Published: 30.09.2018

DOI:
10.21276/sjeat.2018.3.9.7

Abstract: Scheduling the final project is a problem that is often faced by a college. Because there are many criteria that can be used as a reference for preparing the schedule. In the process of schedule preparation has several requirements are needed, such us lecturer teaching schedule should not clash with the schedule of final project. The availability of the chief examiner, timer and rooms for test. Therefore, it need an optimum and efficient method in determining the criteria on scheduling the final project. In this optimization process uses the method of genetic algorithm. Genetic Algorithm (GA) is algorithm that in solving the problem using crossover operator and mutation operator to obtain the desired cross bits. It aims to get the optimal attribute value in terms of fitness equation from chromosome. The result of the training show GA can produce the optimal criteria value of scheduling the final project criteria, so that can be used by the college as the preparation of the final project schedule.
Keywords: Optimizing Scheduling Criteria, Genetic Algorithm, Binary.

INTRODUCTION

One of the schedules that is often encountered in college is schedules the final project. Schedules the final project is the process of arranging an implementation schedule which inform a number a number of students, lecturers who attend and time. Every semester in various College faces the same problem that is how to schedule the final project with the constraints of limitations of the chief examiner, setting the time, day and space for the test.

Based on the description above, in the preparation of schedules the final project there many possibilities that can be tried to find a good schedule. Therefore need an optimization method that can be applied to arrange the schedule of the final project.

Optimization problems can be resolved in several ways, one of which uses a heuristic search algorithm [10]. For simplicity cases a common heuristic search algorithm is used such as breadth-first search and depth-first search. For more complicated inputs and requirements, the heuristic search algorithm cannot be used properly to get the solution as desired. In the case of scheduling the final task trial, a better algorithm is needed that is Algorithm which can solve multi-criteria and multi-objective problem. One of the algorithms that can be used is the Genetic Algorithm [4]. A natural selection process based on genetic principles and natural selection of individuals continuously undergoes gene changes to suit their environment, so that strong individuals are selected that can survive as optimal results. Genetic Algorithms can be used for complex optimization problems (hard optimization problems) that traditional optimizations such as scheduling, transportation, computer games, traveling salesman problem (TSP) and other traditional optimizations cannot solve [2, 3, 5, 9, 11].

This study aims to apply genetic algorithm method to optimize scheduling of final task trial. The optimization of the scheduling of the final assignment is good for the head of the examiner, the day, the hour and the space for the overall test, so there is no problem of schedule collision on the side of the chief examiner, day, timer and the rooms for test.

MATERIALS AND METHODS

Genetic Algorithm

In the search for solutions to a problem sometimes complex mathematical equation are needed to provide a definite solution. The optimum solution may be obtained but requires a long and impractical calculation process. To overcome this case, heuristic method can be used, which is a search method based on intuition or empirical rules to get a better solution than the solution that has been achieved before [2]. The heuristic method does not always produce the best solution, but if it is well designed it will produce a solution that is near optimum in a fast time. Genetic Algorithm (GA) is a branch of evolutionary algorithms, which is an optimization technique based on natural genetics. To produce an optimal solution, GA performs a search process among a number of alternative optimal points based on probabilistic functions [1, 8]

The main problem with GA is how to map one problem to one chromosome string. The GA development cycle begins with the creation of a new set of solutions (initialization) consisting of a number of chromosome strings and placed in a population shelter. Then the reproduction process is carried out by selecting individuals who will be bred. The use of genetic operators such as crossing genetic operators such as cross-over individuals selected in individual shelters will produce offspring or new generations. After an evaluation process for population improvement, these new generations will replace the original population set. This cycle will take place repeatedly until reproduction is not produced, or until the optimum criteria are found.

The genetic algorithm scheme can be summarized as follows [6, 7]:

1. The Initial Population. This stage is the beginning of the Genetic Algorithm process, the initial population is to determine the number of samples to be population studies.
2. Codification. The coding technique is a technique that states the initial population as a potential solution of a problem to the chromosome (3) as a key issue when using the Genetic Algorithm. The Student in Scheduling are encoded as a Chromosome. The criteria that scheduling are coded as genes to be arranged in binary form.
3. Fitness Function. The process of determining the fitness value is done to give the fitness value to each chromosome, the chromosome with high fitness value will survive, while the low fitness value will die. Fitness Value can be calculated using the following equation [6].

$$Fitnes\ (pos) = \frac{(2 - SP) + 2(SP - 1)(Pos - 1)}{N - 1} \quad (1)$$

   Value interval $SP \in [1, 2]$
4. Parent Selection. The parent selection is used to get a pair of chromosomes to be used as a parent, to get a random value as much as the number of chromosomes is generated, if the random value of a chromosome is smaller than the probability of the plotted (pc) then the chromosome will be chosen as the parent.
5. Crossover. This process is done after getting the parent chromosome, then the parent chromosome is crossed with each other for get the off-spring.
6. Mutation. This process is carried out to replace the genes lost due to the selection process. This process will mutate genes by generating random numbers as many as the number of genes, random numbers smaller than the chance of mutation will be mutated. If the gene is 0 then it will be mutated to 1, otherwise if the gene with a value of 1 will be mutated to 0.

New Population. This stage is the stage where the initial population that has passed all stages will be made a new population, then the new population will be reprocessed as well as the initial population until the fitness value of all chromosomes are equal.

EXPERIMENT

The search for optimal values for each criteria of Schedules the final project, starting with codification in the first generation population. Codification performed on the data that has been prepared. These data consist of 24 student at STIKOM college were coded as individual chromosome S1-S24. Each individual has four criteria and each them has 4 sub criteria as shown in table-1. Genetic Algorithm process performed on all students.

Table-1: Student Code

Number	Student Code
1	S1
2	S2
3	S3
⋮	⋮
23	S23
24	S24

Table-2: Codification of Schedule criteria

Student Code	X1=Criteria 1				X2= Criteria 2			
	G1	G2	G3	G4	G1	G2	G3	G4
Student Code	X3=Criteria 3				X4=Criteria 4			
	G1	G2	G3	G4	G1	G2	G3	G4

Genetic Algorithm began by assigning binary bits to each gene of a chromosome. Following the calculation of the fitness value of each chromosome using equation (1).

$$F(1) = \frac{(2-1.5)+2(1.5-1)(1-1)}{24-1}$$
$$= 0.021739$$
$$F(2) = \frac{(2-1.5)+2(1.5-1)(2-1)}{24-1}$$
$$= 0.065217$$
$$\vdots$$
$$F(24) = \frac{(2-1.5)+2(1.5-1)(24-1)}{24-1}$$
$$= 1.021739$$

Then the selection stage, this Selection is used to select individuals as parents. This is called a crossover. There are many ways that can be used to selecting the parents, one of them is Roullettewheel. This method puts each individual in a proportion of circles according to the value of fitness. Individuals with higher fitness scores occupy a larger part of the circle.

The next step is to do chromosome selection, the steps to be taken in chromosome selection are as follows [8]:

1. Calculate the relative fitness value (pk) of each chromosome.
   P1 = F1 / Total of Fitness Value = 0.026 / 12.565 = 0.002.
   P2 = F2 / Total of Fitness Value = 0.079 / 10.565 = 0.005.
   Determination conducted up to the individual to 24. The results can be seen in Table 3.
2. Calculate the cumulative fitness (qk) of each chromosome.

$$Q_1 = P_1 = 0.002$$
$$Q_2 = Q_1 + P_2 = 0.002 + 0.005$$
$$= 0.007$$
$$\vdots$$
$$Q_{24} = Q_{23} + P_{24} = 0.919 + 0.081$$
$$= 1.000$$

The results can be seen in Table-3.

3. Generating random numbers as the number of individuals. Random values are generated by the interval [0 1].

Selection of individuals is done by making a cumulative interval fitness value of each individual. An individual will be selected if the random numbers are generated in their cumulative interval.

Table-3: P_k, q_k, and Random Numbers

Student Code	Chromosome	pk	qk	Random Value	Interval Fitness Cumulative		Chromosome
S1	1	0.002	0.002	0.079	0.000	0.002	7
S2	2	0.005	0.007	0.355	0.002	0.007	15
S3	3	0.009	0.016	0.789	0.007	0.016	22
S4	4	0.012	0.028	0.627	0.016	0.028	20
S5	5	0.016	0.043	0.517	0.028	0.043	18
S6	6	0.019	0.062	0.630	0.043	0.062	20
S7	7	0.022	0.085	0.266	0.062	0.085	13
S8	8	0.026	0.111	0.559	0.085	0.111	18
S9	9	0.029	0.140	0.228	0.111	0.140	12
S10	10	0.033	0.173	0.121	0.140	0.173	9
S11	11	0.036	0.209	0.672	0.173	0.209	20
S12	12	0.040	0.249	0.070	0.209	0.249	7
S13	13	0.043	0.292	0.853	0.249	0.292	23
S14	14	0.047	0.339	0.974	0.292	0.339	24
S15	15	0.050	0.389	0.735	0.339	0.389	21
S16	16	0.054	0.443	0.499	0.389	0.443	17
S17	17	0.057	0.500	0.566	0.443	0.500	19
S18	18	0.061	0.561	0.349	0.500	0.561	15
S19	19	0.064	0.625	0.820	0.561	0.625	22
S20	20	0.067	0.692	0.933	0.625	0.692	24
S21	21	0.074	0.766	0.661	0.692	0.766	20
S22	22	0.074	0.841	0.138	0.766	0.841	9
S23	23	0.078	0.919	0.172	0.841	0.919	10
S24	24	0.081	1.000	0.519	0.919	1.000	18

Opportunities crossover (pc) is one of parameter that indicates the ratio of off-spring which produced in each generation. Pc used was 0.25 by one - point crossover. Election chromosome to crossover process performed by generation 24 random numbers with interval (0, 1).

Table-4: Random Number

Student Code	Chromosome	Random Value
7	1^{*}	0.044137
15	2^{*}	0.454558
22	3*	0.039226
20	4*	0.672927
18	5*	0.07585
20	6*	0.343488
13	7*	0.168606
18	8*	0.588254
12	9*	0.469868
9	10*	0.254137
20	11*	0.676224
7	12*	0.956165
23	13*	0.612323
24	14*	0.767669
21	15*	0.908977
17	16*	0.341739
19	17*	0.356519
15	18*	0.197512
22	19*	0.87489
24	20*	0.475866
20	21*	0.569435
9	22*	0.527668
10	23*	0.518198
18	24*	0.115388

Table-5: Parent Chromosome and Off-Spring

Parent Chromosome																		
S1	1*	1	0	1	0	1	0	1	1	1	0	0	0	1	0	0	1	0.28261
S2	3*	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	0.93478
S3	5*	1	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0.76087
S4	7*	1	0	1	1	1	0	1	0	1	0	0	1	1	0	0	0	0.54348
S5	18*	1	0	0	1	1	0	1	0	1	0	1	1	1	0	0	0	0.63043
S6	24*	1	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0.76087
Off-Spring																		
S1	1*	1	0	1	0	0	1	0	0	1	0	0	0	1	1	0	0	0.28261
S2	3*	0	0	0	0	1	0	1	1	1	0	0	0	1	0	0	1	0.93478
S3	5*	1	0	0	1	1	0	1	0	1	0	0	1	1	0	0	0	0.76087
S4	7*	1	0	1	0	1	1	0	0	0	0	0	0	0	1	0	0	0.54348
S5	18*	1	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0.63043
S6	24*	1	0	0	1	1	0	1	0	1	0	1	1	1	0	0	0	0.76087

The next process is a mutation, which starts by counting the number of bits in the population where:

Number of bits = pop size * Number of Genes (2)
= 24 * 16
= 384

The mutation (pm) used is 0.05, meaning that there will be a 5% chance of the number of bits to be mutated is 20 bits. To specify the bits must be generated random numbers for each bit in the population (as many as 384 random numbers). The selected bit is a bit whose random number is less than pm. The selected bit will be mutated, 0 to 1 and 1 to 0.

The mutation result is the final population on the whole of the first generation process. This final popolation became the initial population in the next generation.

RESULTS

Experiments show that the training results on the data yield varying values for each assessment criterion. By doing the same step then training stopped at the 14th generation, because all individuals show the same fitness value of 1.02174.

Furthermore, to optimize the bit value of each gene, it will be calculated the criterion value by using the following equation [6].

$$x = r_b + (r_a - r_b)(g_1 x 2^{-1} + g_2 x 2^{-2} + \ldots + g_n x 2^{-n}) \quad (3)$$

Where,

x = variable (criteria)

r_a = upper limit of interval value

r_b = lower limit of interval value

g = gen of variable

2^{-n} = numbers of genin variable

Using equation (3) for get x value (criteria) of first chromosome of first generation.

$$x_1 = 1 + (16 - 1)((1(2^{-1}) + 0(2^{-2}) + 1(2^{-3}) + 0(2^{-4}))$$
$$= 10.275$$

$$x_2 = 1 + (16 - 1)((0(2^{-1}) + 1(2^{-2}) + 0(2^{-3}) + 0(2^{-4}))$$
$$= 4.75$$

$$x_3 = 1 + (16 - 1)((1(2^{-1}) + 0(2^{-2}) + 0(2^{-3}) + 0(2^{-4}))$$
$$= 8.5$$

$$x_4 = 1 + (16 - 1)((1(2^{-1}) + 1(2^{-2}) + 0(2^{-3}) + 0(2^{-4}))$$

$= 12.25$

$$\sum_{i=1}^{4} x_i = 35.875$$

The highest total value of x of each generation, will be decisive to determine the optimal value of x as shown in Table-6.

Table-6: The Highest Value of x of Each Generation

Generation	Chief Examiner	Day	Time	Room	Total Value
1	11.3125	8.5	13.1875	10.375	43.375
2	15.0625	8.5	13.1875	10.375	47.125
3	14.125	9.4375	13.1875	10.375	47.125
4	9.4375	10.375	13.1875	9.4375	42.4375
5	4.75	12.25	14.125	12.25	43.375
6	9.4375	11.3125	8.5	10.375	39.625
7	9.4375	11.3125	9.4375	10.375	40.5625
8	9.4375	11.3125	9.4375	10.375	40.5625
9	15.0625	10.375	13.1875	1.9375	40.5625
10	15.0625	10.375	13.1875	1.9375	40.5625
11	15.0625	10.375	13.1875	1.9375	40.5625
12	14.125	14.125	9.4375	1.9375	39.625
13	14.125	14.125	9.4375	1.9375	39.625
14	13.1875	10.375	13.1875	1.9375	38.6875

In this research, the optimal value shown by 7rd and 8rd generation. The optimal value is taken from the individual that has the best criterion value and total value which is indicated by the total value of 40.5625. This means, every criterion has obtained the most optimal value as follows:

Table-7: The Optimal Value of Scheduling Criteria

Variabel x	Name of Criteria	Optimal Value
x_1	Chief Examiner	9.4375
x_2	Day	11.3125
x_3	Time	9.4375
x_4	Room	10.375

CONCLUSION

The result of this research proved that Genetic Algorithm can achieve the optimize value of scheduling the final test criteria as follow:

Chief Examiner : 9.4375
Day : 11.3125
Time : 9.4375
Room : 10.375

The optimal value is largely determined by the generation of random numbers and gene numbers because this affects the results of training. In the 14th generation this training was stopped because the value of each fitness was the same so that the optimal value of each scheduling criterion had been obtained. The results of this optimal value are expected to be applied by college in the process of making schedules the final project.

ACKNOWLEDGMENT

This work support by the Directorate General of Research and Community Service, Strengthening Ministry of Research, Technology and Higher Education of Republic Indonesia and Sekolah Tinggi Ilmu Komputer Pelita Indonesia Pekanbaru.

REFERENCES

1. Aickelin, U., & Dowsland, K. A. (2000). Exploiting problem structure in a genetic algorithm approach to a nurse rostering problem. *Journal of scheduling*, *3*(3), 139-153.

2. Ashlock, D. (2006). *Evolutionary computation for modeling and optimization*. Springer Science & Business Media.
3. Gen, M., & Cheng, R. (2000). *Genetic algorithms and engineering optimization* (Vol. 7). John Wiley & Sons.
4. Gen, M., & Cheng, R. (1997). *Genetic Algorithm and Engeenering Design*: John Wiley & Son, Inc, New York.
5. Hartmann, S. (2002). A self-adapting genetic algorithm for project scheduling under resource constraints. *Naval Research Logistics (NRL)*, *49*(5), 433-448.
6. Jollyta, D., & Hajjah, A. (2017). Genetic Algorithms to Optimizatize Lecturer Assessment's Criteria. In *IOP Conference Series: Earth and Environmental Science* (Vol. 97, No. 1, p. 012005). IOP Publishing.
7. Jollyta, D. (2017). Genetic Algorithm To Optimize The Value Of Customer Satisfaction For Motorcycle User, *Abstract Book ADRI Sumbar 7th-Part 03.*
8. Kusumadewi, S. (2003). Artificial intelligence (teknik dan aplikasinya). *Yogyakarta: Graha Ilmu*, *278*.
9. Lewis, R., & Paechter, B. (2005, March). Application of the grouping genetic algorithm to university course timetabling. In *European Conference on Evolutionary Computation in Combinatorial Optimization*, 144-153. Springer, Berlin, Heidelberg.
10. Suyanto. (2005). *Algoritma Genetika dalam Matlab*. Penerbit Andi.
11. Wang, Y. Z. (2002). An application of genetic algorithm methods for teacher assignment problems. *Expert Systems with Applications*, *22*(4), 295-302.