Plagiarism Checker X Originality Report

Similarity Found: 24%

Date: Minggu, Mei 13, 2018
Statistics: 418 words Plagiarized / 1737 Total words
Remarks: Medium Plagiarism Detected - Your Document needs Selective Improvement.
----

1 Genetic algorithm to solve the problems of lectures and practicums scheduling M F Syahputra1, R Apriani2, Sawaluddin3, D Abdullah4, W Albra5, M Heikal6, A Abdurrahman7 , M Khaddafi8 1,2,3 Department of Information Technology, Faculty of Computer Science and Information Technology, University of Sumatera Utara, Medan, Indonesia 4Department of Informatics, Universitas Malikussaleh, Aceh, Indonesia 5,6,7 Department of Management, Universitas Malikussaleh, Aceh, Indonesia 8Department of Accounting, Universitas Malikussaleh, Aceh, Indonesia Email : nca_fadly@usu.ac.id | dahlan@unimal.ac.id | wahyuddin@unimal.ac.id | mohdheikal@unimal.ac.id | apridar@unimal.ac.id | khaddafi@unimal.ac.id Abstract. Generally, the scheduling process is done manually.

However, this method has a low accuracy level, along with possibilities that a scheduled process collides with another scheduled process. When doing theory class and practicum timetable scheduling process, there are numerous problems, such as lecturer teaching schedule collision, schedule collision with another schedule, practicum lesson schedules that collides with theory class, and the number of classrooms available.

In this research, genetic algorithm is implemented to perform theory class and practicum timetable scheduling process. The algorithm will be used to process the data containing lists of lecturers, courses, and class rooms, obtained from information technology department at University of Sumatera Utara.

The result of scheduling process using genetic algorithm is the most optimal timetable that conforms to available time slots, class rooms, courses, and lecturer schedules. 1. Introduction Generally, the theory class and practicum class scheduling process in university is done manually. This method of scheduling has a low accuracy rate, depending on number of courses scheduled and class rooms available.

Lee [1] stated that scheduling process correlates with time and resources allocation, which will ensure the entire task to be run smoothly. Problems found in timetable scheduling process correlates with the number of classes and the number of students enrolling for a course. The schedule collision can occur when there are students re-enroll for the previous course, or enroll for the more advanced course.

A scheduling process should fulfill each requirement in order to ensure that every scheduled task can be run smoothly [2]. In this case, class timetable scheduling process should meet the schedule of lecturers and students, the availability of class rooms, and the number of courses available, to ensure Therefore, a method has to be implemented to perform theory class and practicum class scheduling process.

2 In this research, genetic algorithm will be implemented to perform theory class and practicum class scheduling process. Genetic algorithm is a heuristic search algorithm based on natural selection process mechanism, which is known as biological evolution process [3]. The algorithm has an optimal search method, and has an ability to perform search for the best solution on numerous solutions available.

The implementation of genetic algorithm in this process should result in the more optimal theory class and practicum class timetable, which fulfills the requirements of zero schedule collisions. Previously we developed application to schedule diet for diabetes mellitus patient [4], where in other area of scheduling, used on container scheduling [5] and manufacturing [6] . 2.

Problem identification Generally, the manual scheduling process on theory class and practicum class timetable in the university has a low accuracy rate, caused by possibilities of schedule collisions between number of classrooms and courses available, has to be implemented for performing the scheduling process on theory class and practicum class timetable. 3. Methodology The general architecture of the system developed for this research is shown by Figure 1.

The processes involved in this research include initialization of population, evaluate fitness function, selection, crossover, and mutation. The best result obtained from this process will be shown as final timetable schedule. Figure 1. General architecture When using genetic algorithm, several constraints have to be defined in order to enable the algorithm to work.

In this research, the constraints are divided to two types, namely hard constraints and soft constraints. Hard constraints defined in this research are described as below: a. Every course can only be held in a class room; b. A lecturer can only teach for a course in one schedule; c. A student can only attend a course for one schedule; d.

Theory class and practicum class should be placed in adjacent schedule slot; e. The number of available class rooms has to meet the number of students attending for the course; f. Each chromosome has to contain course name, lecturer, time slot, class name, and day.

3 Meanwhile, soft constraints defined in this research are described as below: a. The lecturer can select the desired time slot for a course; b. Each schedule has to be distributed evenly for each day; c. Each schedule should not collide with another schedule. 3.1. Population initialization The chromosome will be represented by ?? ?? ?? = [114] [8] , where m represents the number of available courses, and n represents the number of available chromosomes, which combines course, lecturer, credit value, class room, day, and time slot into several pairs.

The example of chromosome representation is shown by Table 1. Table 1. Chromosome representation example Chromosome Course ID Lecturer ID Class ID Credits Room ID Day ID Time ID 1 1 26 1 2 6 4 5 2 2 26 2 2 5 3 4 3 3 6 1 2 3 2 2 114 114 15 1 3 2 4 1 From the chromosome representation expressed above, the initial population can be generated by providing the number of initial population.

The population will be generated randomly by combining several individual from the population. 3.2. Evaluate fitness function Fitness function value will be calculated by using (1): (1) where the penalty value is calculated from the aspects described as below: a. There is a certain number of lecturer with multiple schedule within a time slot in one day; b.

There are pairs of theory class and practicum class scheduled within the same time slot in one day; c. There are some courses within a semester scheduled within the same time slot; and d. There are time slots being assigned by multiple courses within the same time slot.

The fitness function value can be calculated by checking each chromosome in an individual for penalty value, and dividing it with the number of chromosomes, which is 114 as described in population initialization process. 3.3. Selection In this research, the roulette wheel selection method is implemented to perform selection between individuals.

Each individual will be sorted based on the biggest fitness value, and the first two individual with highest fitness value will be used to perform crossover process. 3.4. Crossover Crossover is the method to combine each gene in multiple chromosomes to generate a new individual. The process is done by switching each similar gene in chromosomes. 4 3.5. Mutation Mutation is the next step to generate new individual after crossover.

In this research, exchange mutation method is implemented to perform the mutation process. By using exchange mutation method, the value of several genes in each individual will be exchanged in order to generate a new individual. 4. Experiments and results The system built for this research consists of three menus, namely File, which contains forms needed for scheduling process, namely: courses, lecturers and lab assistants, time slots, class rooms, and class groups; “Pjadala(Sedng , which contains parameter form to start scheduling process by usigetialrit; Infrm hiccoain information about the built system. The appearance of home menu in this system is shown by Figure 2. Figure 2.

The appearance of home menu After scheduling process is done, the result is shown by timetable format, as shown by Figure 3. The schedule timetable can be converted to Excel spreadsheet format. Each schedule is represented by course code, along with day and time slot. 5 Figure 3. Result view The result of experiment using genetic algorithm is shown by Table 2.

The final fitness value of each experiment has to reach the value of 1 for obtaining best optimal result. The experiment result shows that the experiment utilizes fewer value of final population, can result in fewer final fitness value, which indicates that the result is not optimal.

Meanwhile, the random function used in this research results in different result obtained when utilizing the same amount of initial and final population, compared with different amount of initial and final population. Table 2. Experiment result No. Initial population Final population Max. Fitness Min. Fitness Average Fitness Final fitness Generati on result Iteration 1 100 500 1 0.10361 0.26806 1 458 228 2 100 500 1 0.10546 0.29074 1 514 256 3 20 200 1 0.22880 0.61459 1 866 432 4 50 200 1 0.22786 0.48364 1 272 135 5 50 200 1 0.21430 0.58106 1 548 273 6 20 120 1 0.26306 0.90015 0.8789 2000 999 7 50 120 1 0.27653 0.90349 0.8663 2002 1000 8 100 120 1 0.54776 0.92468 0.8864 2000 999 9 50 300 1 0.14537 0.36282 1 380 188 10 100 300 1 0.19552 0.38352 1 250 124 11 50 100 1 0.37414 0.96623 0.9764 2002 1000 12 75 200 1 0.25072 0.50389 1 234 116 6 5.

Conclusion In this research, genetic algorithm is used for arranging timetable schedule of theory classes and practicum classes in university. The result of timetable arrangement using genetic algorithm correlates on random function, which will affect the best result obtained from the experiment. The number of final population utilized in each experiment correlates to the final fitness value, which affects the result of the schedule generated by the algorithm.

For the future research, numerous variables can be used in the scheduling process. Also, parameter addition to the scheduling process is recommended to provide a better result. References [1] Lee, H.S.C. 2000. Timetabling Highly Constrained System Via Genetic Algorithm. Department of Mathematics. College of Science,University of the Philippines. Diliman.

Quezon City [2] M.Kar,M.Hn,N p .Gta,“Gtic go: eviw ndaplicatn, International Journal of Information Technology and Knowledge Management, vol. 2, pp. 451-454, 2010. [3] Kusumadewi, Sri. 2003. Artificial Intelligence (Teknik dan Aplikasinya). Yogyakarta: Graha Ilmu. [4] M F Syahputra, V Felicia, R F Rahmat and R Budiarto 2017. Scheduling Diet for Diabetes Mellitus Patients using Genetic Algorithm. Journal of Physics: Conference Series, Volume 801, Number 1.

[5] for job scheduling at automated container terminals using genetic algorithm Computers & Industrial Engineering, vol. 64, pp. 511-523, 2013. [6] problems on flexible manufacturing systems (FMS): a literature survey, classification and analysis Flexible Services and Manufacturing Journal, vol. 26, pp. 408-431, 2014. [7] Information Sciences, vol. 270, pp. 255- 287, 2014.

INTERNET SOURCES:

<1% - https://www.researchgate.net/profile/Dahlan_Abdullah2/publication/291997502_SISTEM_PAKAR_MENDIAGNOSA_GEJALA_KERUSAKAN_MESIN_MOBIL_TOYOTA_MENGGUNAKAN_METODE_CASE_BASED_REASONING/links/56a8958308ae997e22bdad55.pdf?origin=publication_detail
8% - https://www.researchgate.net/profile/Mohd_Heikal2
<1% - https://en.wikipedia.org/wiki/Education_in_India
3% - https://www.researchgate.net/publication/228451910_Genetic_algorithm_Review_and_application
<1% - http://ijcsse.org/published/volume4/issue6/p4-V4I6.pdf

2% - https://www.researchgate.net/publication/232700842_Optimisation_for_job_scheduling_at_automated_container_terminals_using_genetic_algorithm
<1% - http://iopscience.iop.org/issue/1757-899X/308/1
<1% - https://link.springer.com/article/10.1007/s00477-012-0649-y
<1% - https://www.scribd.com/document/378672143/Generating-Floor-Plan-Layouts-with-KDTre-pdf
<1% - http://www.tjprc.org/publishpapers/2-3-1475649692-6.%20IJCNWMC%20-%20A%20STUDY%20OF%20SMART%20ELECTRIC%20VEHICLE%20CHARGING%20PATH%20SYSTEM.pdf
<1% - https://www.wiwi.hu-berlin.de/de/international/incomings/courses
<1% - https://www.scribd.com/document/99899885/Timetabling
<1% - http://www.wseas.us/e-library/conferences/2005prague/papers/493-343.doc
<1% - https://www.hindawi.com/journals/mpe/2015/272947/
<1% - https://kines.rutgers.edu/faq
<1% - https://technet.microsoft.com/en-us/library/bb691120(v=exchg.141).aspx
<1% - http://www.freepatentsonline.com/7181702.html
<1% - https://www.scribd.com/document/194459492/1-Solving-the-0-1-Knapsack-Problem-With-Genetic-Algorithms
<1% - https://www.hindawi.com/journals/jam/2013/643791/
<1% - https://www.sciencedirect.com/science/article/pii/S1000936117302224
<1% - https://patents.google.com/patent/US6287822B1/en
<1% - http://www.cs.bham.ac.uk/~wbl/biblio/gecco2004/LBP044.pdf
<1% - https://usingtechnologybetter.com/category/google/
<1% - https://www.sciencedirect.com/science/article/pii/S0952197617302282
<1% - https://www.sciencedirect.com/science/article/pii/S0957417413002406
<1% - https://www.tomatis.com/en/research-and-resources
<1% - https://www.sciencedirect.com/science/article/pii/S0306261914012604
1% - http://docplayer.net/8010213-Randomly-encryption-using-genetic-algorithm.html
1% - http://iopscience.iop.org/issue/1742-6596/801/1
1% - http://www.tandfonline.com/doi/full/10.1080/0305215X.2015.1012077
1% - http://yadda.icm.edu.pl/yadda/element/bwmeta1.element.baztech-9aebf2b8-26c9-4960-a76f-21e63a63e9be