Genetic Algorithm for Web-Based Food Stand Assignment Scheduling

Sintya Oktaviani[1], Farica Perdana Putri[2], Dennis Gunawan[3]
Department of Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia
sintyaoktaviani96@gmail.com
farica@umn.ac.id
dennis.gunawan@umn.ac.id

Received on October 5th, 2018
Accepted on December 21st, 2108

***Abstract*— Scheduling is a hard problem due to much considerations in many goals. Combination of goals in this scheduling cause the problem hard to solve even when using mathematical techniques. Optimization is a method which aim to achieve the best result with the least cost as possible. Optimization for large scale problem usually done with more modern technic, such as metaheuristic. Genetic Algorithm belongs to a larger system called Evolutionary Algorithm which is often used for solving the best value in optimization problem. Hence, this food stand assignment scheduling is build using Genetic Algorithm with population size of 50, uniform crossover with crossover rate of 0.25, mutation rate of 0.0125, and roulette wheel selection. An interview was conducted with three coordinators of fund and consumtion that results in three constraints used in building this system. Testing is done for three events and achieve mean fitness that is 87.967%, 89.609%, and 85.001% for FesTIval, TechnoFest, and DISCO, respectively.**

***Index Terms*—Genetic Algorithm, Roulette Wheel Selection, Scheduling, Stand, Uniform Crossover**

I. Introduction

In the process of scheduling, the use of tools such as Personal Digital Assistant, computer, smartphone, or even a notebook is inevitable [1]. Those tools are highly recommended for creating schedule. Scheduling problem is a troublesome problem due to consideration of many goals [2]. Combination of goals in scheduling cause the problem hard to solve even when using mathematical techniques. There are some constraints while creating a schedule, such as processing time and waiting time. A schedule counts as optimal when it can cover all those constraints.

Optimization is a method which goal is to achieve maximal result with as minimum cost as possible [2]. Optimization can be done with various methods. However, for great scale cases, generally modern technique such as metaheuristic is used. Metaheuristic is capable of solving complex problem by trailing some solutions and create a smaller solution, hence an optimal solution might be found.

Komang Setemen has successfully implemented Genetic Algorithm in lecture scheduling system [3]. Scheduling with small dataset which is 53 to 88 subjects is done within 5.000 generation and results in zero for its fitness value, which means there is no collision between schedules. Scheduling for a larger dataset which is 141 courses, Genetic Algorithm is capable of generating a no collision schedule within 10.000 generations. On the other hand, research by Anita Qoiriah proved that final exam scheduling problem is able to be solved with population size of 50, crossover rate of 0.6, and mutation rate of 0.001 [4]. Another research by Marbun compares Genetic Algorithm and Particle Swarm Optimization in solving lecture scheduling problem [5]. It shows that Genetic Algorithm results in the best fitness value which is 1.0, while Particle Swarm Optimization only results in fitness value of 0.111.

In this paper, we investigate Genetic Algorithm for web-based food stand assignment scheduling in three events: FesTIval, TechnoFest, and DISCO. Ferdinand who was coordinator of funding division of Festival 2016 event said that lots of time needed for creating a schedule for food stand duty. Lecture schedule that is very various among committees make it even harder to create a schedule.

The structure of this paper is as follows. In Section II, Genetic Algorithm is discussed. In Section III, we describe the design of our scheduling system. Implementation and testing of Genetic Algorithm in our system described in Section IV. In Section V, conclusion and future works are discussed.

II. Genetic Algorithm

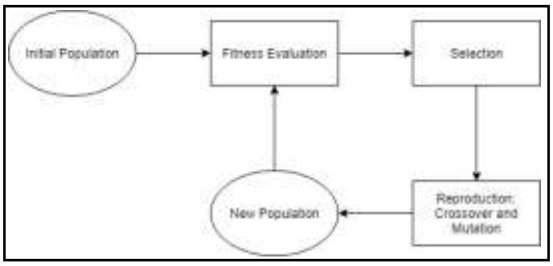

Figure 1. Genetic Algorithm Cycle

Genetic Algorithm belongs to Evolutionary Algorithm which is often used for obtaining a value in an optimization problem [6]. This algorithm is similar to the process of genetic that is in organism or natural selection principle, which is only strong organism can survive in certain environment. Genetic Algorithm cycle first introduced by David Goldberg. Figure 1 shows the Genetic Algorithm cycle [2].

There are several important components in Genetic Algorithm, those are:

1. Chromosome

   One chromosome or individual represents one solution [2]. Generally, encoding is representing some *n* solution using binary number or known as binary encoding.

   Binary encoding is often used due to its simplicity and easy to trace [7]. It has to be noted that if binary encoding is used, the process of binary decoding have to be done in the last phase of Genetic Algorithm. Genetic Algorithm will prepare population as a set of individual. Each individual represents one solution. This population means that there will be plenty solutions due to various individual in this population. Size of recommended population is 50 [8].
2. Fitness

   Fitness function could tell how optimal a solution is [2]. Each individual in the population will be evaluate by fitness function. Formula 1 is a fitness function that is generally used in Genetic Algorithm.

   $$F(x) = \frac{1}{1+f(x)} \qquad (1)$$

   where *f*(x) will produce a good fitness value. After every solution has been evaluate with the fitness function, some chromosomes will be selected. The chosen chromosomes will be the parent for next generation.
3. Roulette Wheel Selection

   Roulette wheel selection can be imagined as a circle that is split as many as the number of chromosomes [2]. Every area in the circle indicates the chance of every solution which will be picked randomly. Every time the wheel is played, the cursor will point at a specific individual [2]. Hence, individual with higher fitness value has a greater chance to be chosen as a parent.
4. Crossover

   Crossover is an operator in Genetic Algorithm that involves 2 parents to produce a new offspring [2]. Chaturvedi advocates to choose a crossover rate 20 times greater than the mutation rate, which is around 0.25 to 0.95 [8]. There are 3 kind of crossovers, which is one-point, two-point, and uniform crossover. One-point crossover which is inspired from the biology process has a blind spot, which is the offspring combination will be the same as the parent in several cases [7]. Two-point crossover seems to be a good method for multipoint crossover. However, the performance of two-point crossover will drop when population is converging due to the reduction of productivity from crossover. Uniform crossover which can change every bit in chromosome can handle the drawbacks from those two crossover methods. Therefore, uniform crossover is more superior to those two.
5. Mutation

   Mutation might bring out new individual which is not born from crossover [2]. Element that will be mutated is chosen randomly. The higher the mutation rate will result in more fluctuate fitness value [8]. Mutation rate generally ranged from 0.001 to 0.05. The purpose of mutation is to generate a new individual which is different from other individual in current population. Mutation might create a new solution to come out from local optimum [2]. Bit flip mutation is generally used for bit encoding. Bit flip mutation will randomly reverse the bit value [9].

Genetic Algorithm consists of 5 steps:

1. Determine the size of population, crossover rate, and mutation rate. Initiate a random population. Evaluate each chromosome in the population with fitness function.
2. Set iteration t = 1.
3. Do a selection to choose the parent for crossover phase.
4. Crossover phase for the chosen parent.
5. Choose some chromosome in population for mutation phase. Evaluate each chromosome with fitness function. If the solution has not reached optimization, set iteration to t = t + 1. Back to step 3.

III. System Design

A. Data Collection

Interview is conducted to collect data which purpose is to know the real problem deeper and to gather information that have relevance with food stand scheduling. Interviewees comes from 3 students that used to create a schedule for events in scope of Technic and Information Faculty in Universitas Multimedia Nusantara.

Ferdinand, Fisichela Thioanda, and Tommy Miyazaki who was coordinator of funding division of TechnoFest 2016 have a same insight of the most fundamental thing while creating a schedule which is each committee assignment must not collide with their lecture schedule. In one-week period, each committee will be assign 4 to 5 hours for food stand duty and this assignment is continuous. Committees assign as the earliest have to move the food to the stand. Therefore, Ferdinand suggest a male committee for morning assignment.

B. System Model

System model from food stand scheduling is shown in Figure 2. There are 6 components in this system, which is:

1. Committee
   Committee is able to input their lecture schedule and obtain food stand assignment schedule for every event they participate.
2. User Interface
   User interface for displaying the interface of food stand assignment system. Committee and admin are able to interact with system through user interface.
3. Admin
   Admin is capable of creating a new event based on committee demand. Furthermore, admin is also capable of viewing all event’s information.
4. Database
   Database is used to store information of every event, committee, and assignment.
5. Coordinator
   Coordinator is students that is used to create a food assignment schedule. Coordinator’s role is to provide constraints in making a schedule.
6. Generate Schedule
   Generate schedule is the process of creating schedule with the implementation of Genetic Algorithm. This Genetic Algorithm use population size of 50, crossover rate of 0.25, and mutation rate of 0.0125. Generate schedule purpose is to create a schedule based on specific event data and constraints that is given by the coordinator. Schedule that has been generated is stored in database.

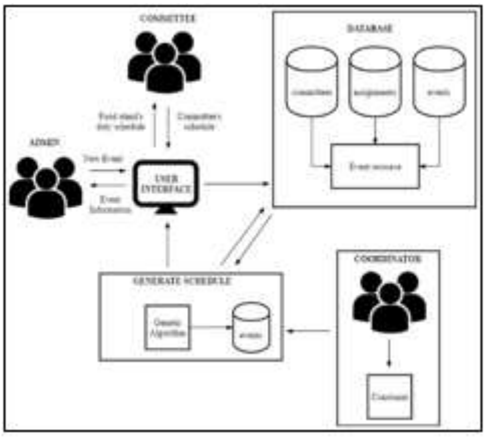

Figure 2. System Model

C. Sitemap

There are 2 sitemaps in this system. Admin sitemap is shown in Figure 3.

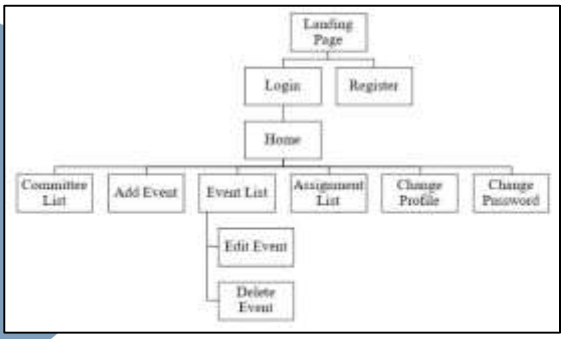

Figure 3. Admin Sitemap

When the first-time user enters the system, user will be direct to Landing Page. User can choose to login or register. User that has login successfully and marked as admin will be able to see every committee, events, and assignments, create new event, change event detail, and change password.

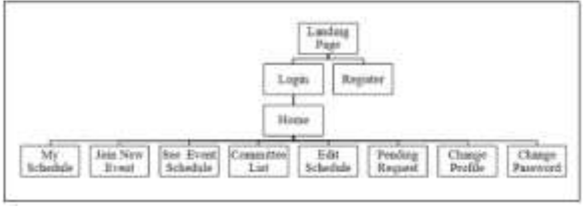

Figure 4. Committee Sitemap

Committee sitemap is shown in Figure 4. User that has successfully login and is marked as committee will be able to see their schedule, join new event, view their event, change their profile, and change their password. Besides that, user might also view every committee in any event they participate in, edit schedule, and view pending request.

D. Flowchart

Process flow of Genetic Algorithm is shown in Figure 5. Genetic Algorithm starts with initialization of every data that is needed, including the starting population. Iteration will start from 1. Fitness will be calculated as reference for Probability Distribution Function (PDF) calculation. Then, there is selection phase for obtaining a better chromosome to be included in crossover phase. After crossover phase, there is mutation phase. Mutation phase is done for some chromosome that is chosen randomly. Then, fitness will be calculated as reference for next PDF calculation. Iteration counter will increase by one by now. If solution is not optimal yet, then there will be a loop from selection phase. An optimal solution will be store to events table in database and success message will be shown.

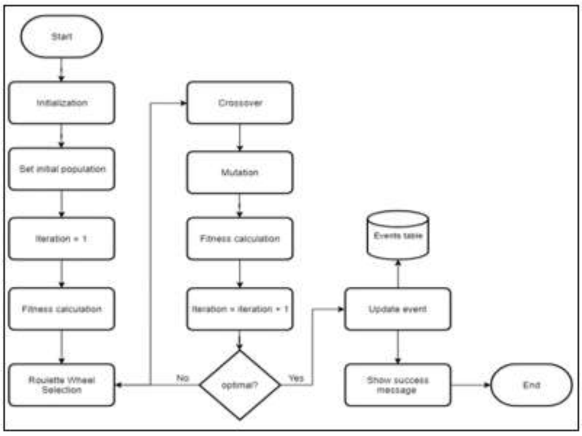

Figure 5. Generate Schedule Flowchart

IV. Implementation and Testing

A. Implementation

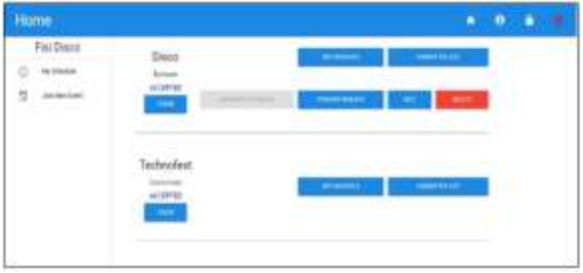

Figure 6. Home Committee Interface

The interface of home committee is shown in Figure 6. In this page, committee is capable of viewing every events they participate. Committee that is place in funding division has more privilege, which is to generate a new schedule, accept or reject committee, edit current schedule, and delete event.

See schedule interface is shown in Figure 7. In this page, committee is capable of viewing the food assignment schedule of a specific event. The name of themselves will be emphasized.

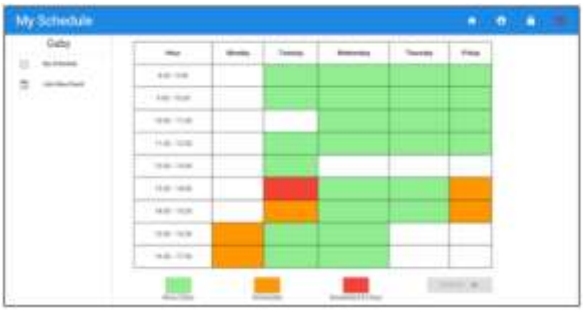

Figure 7. See Schedule Interface

My schedule interface is shown in Figure 8. In this page, committee is capable to see and change their own lecture schedule. There are 3 colors that have different meaning. Green indicates there is lecture schedule, orange indicates there is a food stand assignment, and red means there is a collision between both.

Figure 8. My Schedule Interface

B. System Testing

System testing is done by comparing the fitness value which is generated with different penalty point and iteration. Genetic Algorithm in this system uses population size of 50, uniform crossover with crossover rate of 0.25, mutation rate of 0.0125, and roulette wheel selection. Dataset for testing is from a real past event data which is FesTIval 2016, TechnoFest 2016, and DISCO 2017. Then, a schedule is generated for every case and fitness value that has been generated this way is noted to find its mean. Recapitulation of this testing can be seen in Table 1. The highest fitness mean is achieved by setting the iteration to 30.000, 6 penalty point for schedule collision, 4 penalty point for each hour that made a committee to exceed their max duty hour, and 1 penalty point when a female is assigned to the earliest period. The highest fitness means for three events are 87.967%, 89.609%, and 85.001% for FesTIval, TechnoFest, and DISCO, respectively.

Table 1. Recapitulation of Fitness Value

Event	Collision Penalty	Max Hour Exceeded Penalty	Gender Penalty	Iteration	Mean
FesTIval	6	3	1	10.000	85.771
FesTIval	7	3	1	10.000	85.386
FesTIval	6	4	1	10.000	86.944
FesTIval	6	3	2	10.000	85.765
FesTIval	7	4	1	10.000	86.573
FesTIval	7	3	2	10.000	85.977
FesTIval	6	4	2	10.000	86.386

Event	Collision Penalty	Max Hour Exceeded Penalty	Gender Penalty	Iteration	Mean
FesTIval	7	4	2	10.000	86.184
FesTIval	6	3	1	20.000	86.798
FesTIval	7	3	1	20.000	86.796
FesTIval	6	4	1	20.000	87.416
FesTIval	6	3	2	20.000	86.428
FesTIval	7	4	1	20.000	87.312
FesTIval	7	3	2	20.000	86.181
FesTIval	6	4	2	20.000	87.432
FesTIval	7	4	2	20.000	86.103
FesTIval	6	3	1	30.000	87.039
FesTIval	7	3	1	30.000	86.886
FesTIval	6	4	1	30.000	87.967
FesTIval	6	3	2	30.000	86.564
FesTIval	7	4	1	30.000	87.554
FesTIval	7	3	2	30.000	86.147
FesTIval	6	4	2	30.000	87.796
FesTIval	7	4	2	30.000	86.628
DISCO	6	3	1	10.000	82.66
DISCO	7	3	1	10.000	82.328
DISCO	6	4	1	10.000	84.297
DISCO	6	3	2	10.000	83.065
DISCO	7	4	1	10.000	83.48
DISCO	7	3	2	10.000	82.361
DISCO	6	4	2	10.000	84.137
DISCO	7	4	2	10.000	83.175
DISCO	6	3	1	20.000	83.935
DISCO	7	3	1	20.000	82.762
DISCO	6	4	1	20.000	84.414
DISCO	6	3	2	20.000	83.427
DISCO	7	4	1	20.000	84.099
DISCO	7	3	2	20.000	82.718
DISCO	6	4	2	20.000	84.636
DISCO	7	4	2	20.000	83.918
DISCO	6	3	1	30.000	83.366
DISCO	7	3	1	30.000	83.419
DISCO	6	4	1	30.000	85.001
DISCO	6	3	2	30.000	84.177
DISCO	7	4	1	30.000	84.085
DISCO	7	3	2	30.000	83.304
DISCO	6	4	2	30.000	84.705
DISCO	7	4	2	30.000	83.908
TechnoFest	6	3	1	10.000	88.036
TechnoFest	7	3	1	10.000	87.589
TechnoFest	6	4	1	10.000	88.174
TechnoFest	6	3	2	10.000	87.365
TechnoFest	7	4	1	10.000	87.636
TechnoFest	7	3	2	10.000	87.415
TechnoFest	6	4	2	10.000	88.477
TechnoFest	7	4	2	10.000	88.248
TechnoFest	6	3	1	20.000	88.177
TechnoFest	7	3	1	20.000	87.92
TechnoFest	6	4	1	20.000	89.266
TechnoFest	6	3	2	20.000	88.04
TechnoFest	7	4	1	20.000	88.177
TechnoFest	7	3	2	20.000	87.81
TechnoFest	6	4	2	20.000	88.477
TechnoFest	7	4	2	20.000	88.692
TechnoFest	6	3	1	30.000	88.95
TechnoFest	7	3	1	30.000	89.091
TechnoFest	6	4	1	30.000	89.609
TechnoFest	6	3	2	30.000	88.564
TechnoFest	7	4	1	30.000	88.367
TechnoFest	7	3	2	30.000	87.979
TechnoFest	6	4	2	30.000	89.204
TechnoFest	7	4	2	30.000	88.772

V. CONCLUSION AND FUTURE WORKS

Genetic Algorithm has successfully been implemented in this food stand assignment scheduling system. This system generates a schedule for one-week period. The result of interview concludes 3 constraints that is used as references in generating a schedule. Those 3 constraints are food stand assignment schedule should not collide with committee lecture schedule, food stand duty hour should not exceed 5 hours for one-week period, and the earliest period of each day should prioritize a male committee.

The experimental results indicate that the highest mean fitness value can be obtained with penalty point configuration such as, 6 penalty point for schedule collision, 4 penalty point for each hour that made a committee to exceed their max duty hour, and 1 penalty point when a female is assigned to the earliest period. The highest fitness mean for three event is 87.967%, 89.609%, and 85.001% for FesTIval, TechnoFest, and DISCO, respectively.

Based on the research that has been done, here are some ideas for future research.

1. System can be integrated with single sign-on from UMN system, hence system can gather the lecture schedule for each committee automatically.
2. This system can be developing to be mobile application to ease committee in viewing their schedule.

REFERENCES

[1] Kompas. 2009. Agar 24 Jam Lebih Efektif. [Online]. Available on:http://lifestyle.kompas.com/read/2009/12/15/13422113/agar.24.jam.lebih.efektif [8 May 2017].

[2] Sianturi, A. L., "Optimasi Penjadwalan Karyawan Pengawas Pembangunan Kapal Dengan Menggunakan Algoritma Genetika," Depok: Universitas Indonesia, 2012.

[3] Setemen, K., "Implementasi algoritma genetika dalam pengembangan sistem aplikasi penjadwalan kuliah," Singaraja: Universitas Pendidikan Ganesha, 2010.

[4] Qoiriah, A., "Penjadwalan Ujian Akhir Semester dengan Algoritma Genetika (Studi Kasus Jurusan Teknik Informatika UNESA)," Surabaya: Jurnal Manajemen Informatika UNESA, 2014.

[5] Marbun, Y., "Perbandingan Algoritma Genetika dan Particle Swarm Optimization dalam Optimasi Penjadwalan Matakuliah," 2013.

[6] Leow, P., "Genetic Algorithms Demystified: Unravel the Myths and Power of Genetic Algorithms in Machine Learning," 2016.

[7] Man, K. F., Tang, K. S., & Kwong, S., "Genetic Algorithms: Concepts and Applications," IEEE Transactions On Industrial Electronics, vol. 43, no. 5, 1996.

[8] Chaturvedi, D., Soft Computing Techniques and its Applications in Electrical Enginering, Studies in Computational Intelligence (SCI), Berlin: Springer-Verlag Berlin Heidelberg, 2008.

[9] Kramer, O., Genetic Algorithm Essentials, Springer International Publishing, 2017.