Analysis of Artificial Bee Colony algorithm for optimizing lecture schedule based on willingness of teaching submission

C N Alam¹*, D Maylawati², M R Al-fayumi¹ and M A Ramdhani¹

¹ Departement of Informatics, UIN Sunan Gunung Djati Bandung, Jalan A H Nasution 105 Bandung 40614, Indonesia
² Departement of Informatics, Sekolah Tinggi Teknologi Garut, Jalan Mayor Syamsu No 1 Tarogong Kidul Kabupaten Garut 44151, Indonesia

*cecep@uinsgd.ac.id

Abstract. Scheduling is one of important activity, especially in universities. Many activities of lecturers make a lecturers submit a willingness of teaching schedule. This allows greater opportunities for conflicting schedules. In this research, we use Artificial Bee Colony (ABC) algorithm for optimizing lecture schedule based on willingness of teaching schedule submission. With ABC algorithm is expected to reduce conflicting schedules and optimize the schedule according to the willingness of lecturer to teach. In this paper explain the analysis and pattern (encoded) of ABC algorithm that have ready to be implemented. The result of analysis shows that ABC can be applied for optimizing schedule. Then, we describe about several scenarios of experiment which can be used for testing the performance or accuracy of ABC algorithm for optimizing schedule based on willingness teaching submission

1. Introduction
Schedule management becomes complicated if managed manually. However, the technology enables the schedule can be created automatically. Implementation of information technology has been proven to solve many problems accurate, effective, and efficient [1] [2] [3] [4]. Especially, at the university that has its own complexities in scheduling. Not infrequently conflicting schedules made as it relates to time, class, and even lecturers who available. Lecturers also have a lot of activity which causes lecturer submit their individual schedules. This increases the possibility of conflicting schedules higher.

Technological devices are designed to enhance a quality of human’s life [5], one of those which are enable efficiency and effectiveness in business process is information systems. Information systems (IS) is a combination of information technology utilizations and human activity upon a set of agreed procedure [6], generally is used to support management and operation [7]. IS is an organized data process [8], IS has a high level of flexibilities to develop and scalable [9]. Refers to several research, an information system has a high capability in decision making, the system has an accurate data accessibility and efficient run-time [10], high accuracy [11], and to support a proper decision [12], low cost [13], extended accessibility [14], intensify user knowledge [15], increase productivity [16], provide a better data and information [17], and in the certain cases are potentially used as data storage [18].

Many algorithm for optimizing case, such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Differential Evolution Algorithm, Ant System Algorithm, and so on. However, Artificial Bee Colony (ABC) algorithm is one of probabilistic optimization algorithm that can be used to create optimal
schedule [19] [20] [21] [22]. Comparing with GA, PSO, and Differential Evolution algorithm, ABC algorithm is better and more efficient for optimizing case with multivariable [23] [24] [25]. Besides that, ABC is interesting algorithm for research because unique and has not been many used for creating an optimal schedule yet [19], especially lecture schedule. All this time, GA is an algorithm that often for scheduling [26] [27] [28] [29] [30] [31] or another algorithm for optimal scheduling case. ABC is an algorithm that is adapted from Monte Carlo basic concept with random and repeated sample to get best solution, where modelling behaviour of bee in search the food sources [19] [21] [22] [23] [24] [32].

In this works, we use ABC for creating optimal lecture schedule with several input parameter such as lecturers, courses, and classes with the value position. The willingness of teaching schedule submission from lecturers is a challenge how the lecture schedule do not conflict and can satisfy lecturer schedule submission. The optimal schedule will be the best food sources in ABC algorithm. In the upcoming section, we describe the concept of ABC algorithm in Section II. Analysis and Encoding of ABC for schedule optimizing based on willingness teaching submission in Section III. The experiment scenarios of ABC algorithm for optimizing Lecture Schedule based on willingness of teaching submission in Section IV. And finally, conclusion of this research stated in Section V.

2. **Artificial Bee Colony algorithm**

Artificial Bee Colony (ABC) algorithm is part of swarm intelligent algorithm that is adapted from behaviour of animal colony such as Swarm Optimization and Ant Colony algorithm [20] [24] [25] [32]. ABC is adapted from behaviour of bee in search the food source which has three main component, among others [24] [25] [32]:

- **Food sources**, the value or quality of a food source is determined by the distance to the honeycomb, the number of foods, and easily get the food.
- **Employed foragers**, the bees are in charge of storing information from food sources which have been found. Either an information about the distance or direction from the hive, the level of profitability, or wealth of information from a food sources to be distributed. Therefore, the number of employed bees should be equivalent to the amount of the food sources that have been found. In sharing the information the food sources, employed bees do a dance that is called Waggle Dance in Dancing Room which is located at the centre of a beehive with onlooker bees as watching and selecting food sources to be exploited.
- **Unemployed foragers**, the bees were tasked with finding food sources that can be exploited. There are two types of unemployed foragers: Scouts, a bee agent in charge for position food sources in the environment around the nest at random to a certain extent.
- **Onlookers**, a bee agent whose job is to select and exploit food sources that the information is kept by employed bee.

Generally, the algorithm of ABC is as follow [10] [11] [18]:

```
Initialization Phase REPEAT
Employed Bees Phase
Onlooker Bees Phase
Scout Bees Phase
Memorize the best solution achieved so far
UNTIL (Cycle = Maximum Cycle Number/MCN)
//Initialization Phase
Initializing the population from solution
//Evaluating the population
//Employed Bees, Onlooker Bess, and Scout Bees Phase
a. Generate new solution (Vi) in the neighbourhood (Xi) for employed bees using the following equation:
```
Vij = Xij + φij (Xij - Xkj)
Where Xi is a solution that was obtained randomly and \( \phi \) is a random number in range (-a, a). Evaluation of new solution Vi.

b. Calculate the probability of the value of \( p_i \) for solution \( X_i \) using the fitness value with the following equation:

\[
P = \frac{f_{sol}}{\sum_{n=1}^{n} f_{sol}}
\]

c. Generate new solutions for onlookers to \( V_i \) from the solution \( X_i \) that is selected by \( p_i \)

//Save the best food sources that have been obtained
//Do repetitions to the number of cycle until MCN

3. Analysis and encoding of Artificial Bee Colony for schedule optimizing based on willingness teaching submission

In this research, food source which is set of solution will be filled by all couple of lecturers, courses, classes, and position value. For example, Table 1 describes the teaching willingness submission from lecturers and will be encoded with format as follow: lecturer_code+courses+class+credit_hours, for example DS.AI.A_3 for Dian Sa’adillah who teach Artificial Intelligent (AI) course in class A and credit hours is 2. The example from Table 1 has several constraints, for understanding the ABC algorithm easily, such as schedule only on Monday and Tuesday from 7 o’clock until 12 o’clock and two classroom available. This section will describe how to make couple encoded and how ABC algorithm works to create the optimize schedule based on willingness teaching submission. Table 1 provides the example with 4 lecturers with each courses, classes, credit hour of course, and time (day and hour, with 60 minutes per credit value) of willingness teaching submission.

| Lecturer | Courses | Class | Credit | Day     | Hour   |
|----------|---------|-------|--------|---------|--------|
| DS       | Data    | A     | 2      | Monday  | 7-8    |
|          | Base    | B     |        | Monday  | 9-10   |
| SWE      | A       |       |        | Tuesday | 9-10   |
| GT       | Network | A     | 2      | Monday  | 11-12  |
|          | Cryptography | B | 2     | Tuesday | 7-8    |
|          |         | A     | 2      | Tuesday | 9-10   |
| AL       | AI      | B     | 3      | Monday  | 7-9    |
| DN       | Logic   | B     | 3      | Tuesday | 10-12  |

The cells in table 1 that is coloured is a conflict schedule because several lecturers submit on the same day and the same hours. ABC algorithm will be found the best schedule with no conflict and as much as possible win accordance with the proposed schedule. So, ABC algorithm for case in table 1 will be run as the following processes (description of symbols is available in Appendix in table 12):

- Input parameters among others encode of lecturers, courses, sum of classes (2 class), schedule submission, and encode format for lecturer+courses+class+credit
- Initiate d=1, then count time slots for each lecture based on credit hours. If there are lecturers who have not been counted, then d=d+1 and back to process number 3
- Fill the parameters of ABC, among others sum of population (n), limit value (for this case limit value is 2), and MCN value (for this case is 1 for once iteration)
Then, ABC need to know slots that has been filled. So, we provide list of willingness teaching slots, described in Table 2, where 1 is for according to submission schedule and for not according to submission schedule. Colouring cells show the conflict schedule or not accordance with schedule submission.

**Table 2. Willingness teaching schedule.**

|      | 7 | 8 | 9 | 10 | 11 | 12 |
|------|---|---|---|----|----|----|
| Monday | 1 | 1 | 1 | 1  | 0  | 0  |
| Tuesday| 0 | 0 | 1 | 1  | 0  | 0  |
| DN    | 7 | 8 | 9 | 10 | 11 | 12 |
| Monday | 0 | 0 | 0 | 0  | 0  | 0  |
| Tuesday| 1 | 1 | 1 | 1  | 0  | 0  |
| GT    | 7 | 8 | 9 | 10 | 11 | 12 |
| Monday | 0 | 0 | 1 | 1  | 1  | 1  |
| Tuesday| 1 | 1 | 1 | 1  | 0  | 0  |
| AL    | 7 | 8 | 9 | 10 | 11 | 12 |
| Monday | 1 | 1 | 1 | 1  | 1  | 1  |
| Tuesday| 1 | 1 | 1 | 1  | 1  | 1  |

**Table 3. 1st Position of solution for each encoded couple.**

| Encoded Couple | 1st Position of Solution |
|----------------|--------------------------|
| DN_IN_A_3      | 0.56                     |
| DS_BD_A_2      | 0.79                     |
| DS_BD_B_2      | 0.84                     |
| GT_JAR_A_2     | 0.69                     |
| GT_JAR_B_2     | 0.64                     |
| DS_RPL_A_2     | 0.57                     |
| GT_KR_A_2      | 0.44                     |
| AL_IN_B_3      | 0.81                     |
| AL_LI_B_3      | 0.65                     |

Next, count the value of position of solution for each encoded couple. Table 3 is the first position of solution for each encoded couple.

- Initiate iteration variable is equal with 1, and sol=1
- For each couple that has been created, the next steps among others:
  a. Sum all of teaching submission slots for each lecturer
  b. Create set of H which is contained (jpsg+1)
  c. Choose and sort from the encoded couple that has the biggest position value until the smallest
  d. Scheduled the encoded couple that has the biggest position value
  e. Update the willingness of teaching schedule in Table 2, change all value in the table with 0 and fill 1 for slot that is contained the encoded couple with the biggest position value. And also update the class room availability, fill with 1 if the room already has a schedule and 0 for available class room.
  f. Repeat process a to e until all of encoded couple have been scheduled. The result of willingness of teaching scheduled and class room availability describe in Table 4, Table 5, and Table 6. Coloring cells is schedule that nor accordance with lecturer submission. For example, the slot for the submission schedule of the last encoded couple is not available, so the last encoded couple will be scheduled in still available slot on the same day.
Table 4. Update willingness teaching schedule.

| Room/Day | Monday | Tuesday |
|----------|--------|---------|
| DS       | 7 8 9 10 11 12 | 7 8 9 10 11 12 |
| Monday   | 1 1 1 1 0 1 | 0 0 0 1 1 0 |
| Tuesday  | 0 0 0 1 0 1 | 1 1 0 1 0 1 |
| DN       | 7 8 9 10 11 12 | 7 8 9 10 11 12 |
| Monday   | 0 0 0 1 1 1 | 1 1 0 0 1 0 |
| Tuesday  | 0 0 1 0 1 1 | 0 0 1 1 1 1 |
| GT       | 7 8 9 10 11 12 | 7 8 9 10 11 12 |
| Monday   | 1 1 1 0 0 0 | 0 0 1 1 1 1 |
| Tuesday  | 0 0 0 1 1 1 | 0 0 0 0 0 0 |
| AL       | 7 8 9 10 11 12 | 7 8 9 10 11 12 |
| Monday   | 0 0 0 1 1 1 | 0 0 1 1 1 1 |
| Tuesday  | 0 0 0 0 0 0 | 0 0 0 0 0 0 |

Table 5. Update classroom availability.

| Room/Hour | Monday | Tuesday |
|-----------|--------|---------|
| Room 1    | 1 1 1 1 1 1 | 1 1 0 1 1 1 |
| Room 2    | 1 1 1 1 1 1 | 0 0 1 1 1 1 |

Table 6. Schedule from 1st position of solution.

| Monday | Room | 7 8 9 10 11 12 |
|--------|------|----------------|
| A      | R 01 | DS_BD_A_2      |
|        | R 01 | GT_JAR_A_2     |
| B      | R 02 | DS_BD         |
| Tuesday| Room  | 7 8 9 10 11 12 |
|        | R 01 | GT_JAR_A_2     |
| A      | R 02 | DS_RPL_A_2     |
|        | R 02 | GT_KR_A_2      |
| B      | R 01 | AL_IN_B_3      |
|        | R 02 | DN_IN_A_3      |

- Count fitness value for each encoded couple in the first position of solution. Fitness value is counted with equation \((fa1) = \frac{1}{1+Fi}) and will result new position of solution or second position of solution (describe in Table 7). Do the same process with number 7, then all of encoded couple will be scheduled as like as the willingness teaching submission (describe in Table 8).
- Check whether all of solution have been scheduled run until step number 10 and have been counted the fitness value. If solution has been scheduled (sol = 2), then next run to step number 10. Else, if sol is smaller than n, then sol = sol+1, then the solution is scheduled with the previous steps.
- Fill sol = 1, then do the evaluation with employed bee, first and second solution will be changed by employed bee (described in Table 9) using equation \(Vij = Xij + \phi ij (Xij - Xkj)\), where i is determined sequentially \((i=1, 2, ..., n \text{ where } i=\text{sol})\) and j (in algorithm is y) which is random for
first couple and second solution of position for fourth couple that is chosen randomly. The result of schedule after has evaluated by employed bee for first position of solution describe in Table 10. Then, the result schedule after has evaluated by employed bee for second position of solution describes in Table 11. Encoded couple will be positioned accordance with willingness of teaching schedule. If the slot which is wanted is full then encoded couple will be positioned in the next time slot in the same day and class.

**Table 7. 2nd Position of Solution for each encode couple.**

| Encoded Couple | 2nd Position of Solution |
|---------------|--------------------------|
| DN_IN_A_3     | 0.94                     |
| DS_BD_A_2     | 0.87                     |
| GT_JAR_A_2    | 0.85                     |
| DS_BD_B_2     | 0.79                     |
| AL_IN_B_3     | 0.76                     |
| AL_LI_B_3     | 0.69                     |
| GT_KR_A_2     | 0.68                     |
| GT_JAR_B_2    | 0.59                     |
| DS_RPL_A_2    | 0.58                     |

**Table 8. Schedule from 2nd position of solution.**

| Monday | Room | 7  | 8  | 9  | 10 | 11 | 12 |
|--------|------|----|----|----|----|----|----|
| A      | R 01 | DS_BD_A_2 | GT_JAR_A_2 |
|        | R 02 |               |              |
| B      | R 01 | DS_BD_B_2 |
|        | R 02 | AL_IN_B_3 |
| Tuesday| Room | 7  | 8  | 9  | 10 | 11 | 12 |
| A      | R 01 | DN_IN_A_3 |
|        | R 02 | GT_JAR_B_2 | GT_KR_A_2 | DS_RPL_A_2 |
| B      | R 01 | AL_LI_B_3 |
|        | R 02 |               |

**Table 9. Position of solution evaluation by employed Bee.**

| Encoded Couple | 1st Solution Position | Encoded Couple | 2nd Solution Position |
|---------------|-----------------------|---------------|-----------------------|
| DN_IN_A_3     | 0.62                  | DN_IN_A_3     | 0.94                  |
| DS_BD_A_2     | 0.79                  | DS_BD_A_2     | 0.87                  |
| DS_BD_B_2     | 0.84                  | GT_JAR_A_2    | 0.85                  |
| GT_JAR_A_2    | 0.69                  | DS_BD_B_2     | 0.95                  |
| GT_JAR_B_2    | 0.64                  | AL_IN_B_3     | 0.76                  |
| DS_RPL_A_2    | 0.57                  | AL_LI_B_3     | 0.69                  |
| GT_KR_A_2     | 0.44                  | GT_KR_A_2     | 0.68                  |
| AL_IN_B_3     | 0.81                  | GT_JAR_B_2    | 0.59                  |
| AL_LI_B_3     | 0.65                  | DS_RPL_A_2    | 0.58                  |
Table 10. 1st Position of solution evaluation by employed Bee.

| Monday | Room  | 7  | 8  | 9  | 10 | 11 | 12 |
|--------|-------|----|----|----|----|----|----|
| A      | R 01  | DS_DB | D_A_2 | GT_JAR |       |    |    |
|        | R 02  |       |       |       |       | GT_KR_A_2 | |
| B      | R 01  |       | AL_IN_B_2 |       |       |    |    |
|        | R 02  | AL_IN_B_3 |       |       |    |    |    |

Table 11. 2nd Position of solution evaluation by employed Bee.

| Monday | Room  | 7  | 8  | 9  | 10 | 11 | 12 |
|--------|-------|----|----|----|----|----|----|
| A      | R 01  | DS_B | D_A_2 | GT_JAR |       |    |    |
|        | R 02  |       |       |       |       |    |    |
| B      | R 01  | DS_BD_B_2 |       |       |    |    |    |
|        | R 02  | AL_IN_B_3 |       |       |    |    |    |

- Check whether all solution have been scheduled, if have not been scheduled yet, fill sol = sol+1, else run to step 12.
- Fill sol = 1, then give the probability value for each solution that has been scheduled based on fitness value with equation /P = f_sol/∑_i=1^n f_u. Check if sol=1 then filled Cumsol=Psol, else Cumsol=Cum (sol-1).
- Fill sol = 1 and m=1, then do evaluation by onlooker bee with equation Vij = Xij + φij (Xij - Xkj). In evaluation with onlooker bee, first and second bee choose the solution and change two of encoded couple from the choosen solution randomly. For example, first and second bee choose first solution. First bee will choose sixth encoded couple and second bee will choose the fifth encoded couple. So, in TABLE XI encoded couple GT_KR_A_2 will be exchanged with DS_RPL_A_2.
- Check if sol=n then do the next step, else do the previous step for do evaluation with onlooker bee. Fill sol=1, then check whether the solution has been evaluated by onlooker bee. If the best solution still equal or smaller than first solution, then CN value will be added by 1 (CN=CN+1), else CN=0.
- Do Scout Bees evaluation for check whether CN value is equal with the limit. Fill y=1, if CN has reach the limit (for this case is 1), then the solution that has reach the limit will be changed with xsol,y=xmin,y+rand(0,1)*(xmax,y-xmin,y) for get new funnity value by scout bees evaluation.
• Check whether sol=n, if sol is smaller than n then fill sol=sol+1. Next, back to evaluation phase by onlooker bees. Save the best solution for lecture scheduling that has the most optimal fitness value with equation Best=Max(fsol, Best).
• Check stop condition, whether iteration value is equal with MCN value (for this example only one iteration)

4. The experiment scenarios
The experiment for testing performance or accuracy of ABC algorithm must be adjusted with real case study or real implementing environment. Several scenarios or test case of experiment that can be done is as follow:

• Create several case study with several of willingness of teaching schedule submission, such as total of lecturers, total of available class, and courses therewith its credit hour. For example, for first case study, total of lecturer is 5 lecturers with 6 types of courses that will result several encoded couple (for example to be 30 encoded couple), and available classes is 10 classroom with 10 time slot accordance with each credit hour. Second case study is added total of lecturers and courses (automatically increase the encoded couple) with the same available class. Third case study, decrease total of available class with the same total of lecture, total of courses, and total of encoded couple as first or second case study. The point is to make a case study with a variety of input parameters and the possibility of cases occur in creating the schedule.
• Do several iteration for each case study. Because every iteration and total of iteration produce the different optimal schedule. It does not rule out the possibility of the total iteration affect the result of schedule. For example first case study in number 1 is done with once iteration, twice iteration, three times iteration, and so on. Then, compare the result schedule of each iteration for each case study. It shall be make the accuracy of ABC algorithm more reliable

5. Conclusion
ABC algorithm is one of optimization algorithm that is adapted from behaviour of bee. Generally, ABC algorithm is divided into three phase, among others employed bee phase, onlooker bee phase, and scout phase. This research describes the analysis how to use ABC algorithm for optimizing lecture schedule based on willingness of teaching schedule submission. For the future works, ABC algorithm must be tested to know the performance or accuracy of ABC algorithm in creating the optimal schedule. And of course ABC algorithm must be implemented accordance with the real case study.

Acknowledgement
We say many thanks to Research and Publication Center of UIN Sunan Gunung Djati Bandung Indonesia for supporting publication of our research

References
[1] S Maylawati, W Darmalaksana and M A Ramdhani 2018 Systematic Design of Expert System Using Unified Modelling Language IOP Conference Series: Materials Science and Engineering 288(1) p. 012047
[2] D S Maylawati, M A Ramdhani, W B Zulfikar, I Taufik and W Darmalaksana Expert System for Predicting the Early Pregnancy with Disorders using Artificial Neural Network in 5th International Conference on Cyber and IT Service Management (Denpasar)
[3] C Slamet, R Andrian, D S Maylawati, Suhendar, W Darmalaksana and M A Ramdhani 2018 Web Scraping and Naïve Bayes Classification for Job Search Engine IOP Conference Series: Materials Science and Engineering 288(2017) p. 012038
[4] C Slamet, A Rahman, M A Ramdhani and W Darmalaksana 2016 Clustering the Verses of the Holy Qur'an using K-Means Algorithm Asian Journal of Information Technology 15(24) pp. 5159-5162
[5] M A Ramdhani, H Aulawi, A Ilkhwana and Y Mauluddin 2017 Model of green technology
adaptation in small and medium-sized tannery industry J. Eng. Appl. Sci. 12(4) pp. 954–962

[6] A Pamoragung, K Suryadi and M A Ramdhani Enhancing the implementation of e-Government in Indonesia through the high-quality of virtual community and knowledge portal in Proceedings of the European Conference on e-Government ECEG 2006 pp. 341–348

[7] M A Ramdhani 2013 Metodologi Penelitian untuk Riset Teknologi Informasi (Bandung: UIN Sunan Gunung Djati Bandung)

D S Maylawati, W Darmalaksana and M A Ramdhani 2018 Systematic Design of Expert System Using Unified Modelling Language IOP Conf. Ser. Mater. Sci. Eng. 288(1) p. 12047

[8] H Aulawi, M A Ramdhani, C Slamet, H Ainissyifa and W Darmalaksana 2017 Functional Need Analysis of Knowledge Portal Design in Higher Education Institution Int. Soft Comput. 12(2) pp. 132–141

[9] C Slamet, A Rahman, A Sutedi, W Darmalaksana, M A Ramdhani and D S Maylawati 2018 Social Media-Based Identifier for Natural Disaster IOP Conf. Ser. Mater. Sci. Eng. 288(1) p. 12039

[10] Slamet C, Andrian R, Maylawati, D S A, Darmalaksana W and Ramdhani M A 2018 Web Scraping and Naïve Bayes Classification for Job Search Engine In IOP Conference Series: Materials Science and Engineering 288(1) (IOP Publishing) p. 012038

[11] Y A Gerhana, W B Zulfikar, A H Ramdani and M A Ramdhani 2018 Implementation of Nearest Neighbor using HSV to Identify Skin Disease IOP Conf. Ser. Mater. Sci. Eng. 288(1) p. 012153

[12] A Rahman, C Slamet, W Darmalaksana, Y A Gerhana and M A Ramdhani 2018 Expert System for Deciding a Solution of Mechanical Failure in a Car using Case-based Reasoning IOP Conf. Ser. Mater. Sci. Eng. 288(1) p. 12011

[13] C Slamet, A Rahman, M A Ramdhani, and W Darmalaksana 2016 Clustering the Verses of the Holy Qur’an using K-Means Algorithm Asian J. Inf. Technol. 15(24) pp. 5159–5162

[14] D S Maylawati, M A Ramdhani, W B Zulfikar, I Taufik, and W Darmalaksana 2017 Expert system for predicting the early pregnancy with disorders using artificial neural network 2017 5th Int. Conf. Cyber IT Serv. Manag. CITSM 2017.

[15] W B Zulfikar, Jumadi, P K Prasetyo and M A Ramdhani 2018 Implementation of Mamdani Fuzzy Method in Employee Promotion System IOP Conf. Ser. Mater. Sci. Eng. 288(1) p. 12147

[16] D S Maylawati, M A Ramdhani, A Rahman and W Darmalaksana 2018 Incremental technique with set of frequent word item sets for mining large Indonesian text data 2017 5th Int. Conf. Cyber IT Serv. Manag. CITSM 2017 pp. 1–6

[17] A Taofik, N Ismail, Y A Gerhana, K Komarujmanan and M A Ramdhani 2018 Design of Smart System to Detect Ripeness of Tomato and Chili with New Approach in Data Acquisition in IOP Conference Series: Materials Science and Engineering 288(1) p. 12018

[18] H Yoviana and D Suryadi 2011 Perancangan Sistem Penjadwalan Kuliah di Fakultas Teknologi UNPAR dengan Menggunakan Algoritma Artificial Bee Colony (Bandung: Parahyangan University)

[19] A Hasad 2011 Algoritma Optimasi dan Aplikasinya (Bogor: Graduate School, Department of Computer Science Bogor Agricultural University)

[20] L Wang, G Zhou, Y Xu, S Wang and M Liu 2012 An effective artificial bee colony algorithm for the flexible job-shop scheduling problem The International Journal of Advance Manufacturing Technology 60(1) pp. 303-315

[21] R Zhang, S Song and C Wu 2013 A hybrid artificial bee colony algorithm for the job shop scheduling problem International Journal of Production Economics 141(1) pp. 167 178

[22] B Akay and D Karaboga 2012 A modified Artificial Bee Colony algorithm for real-parameter optimization Information Sciences 192 pp. 120-142

[23] D Karaboga and B Akay 2009 A comparative study of Artificial Bee Colony algorithm Applied Mathematics and Computation 214(1) pp. 108-132

[24] D Karaboga, B Gorkelmi, C Ozturk and N Karaboga 2014 A Comprehensive Survey: Artificial
Bee Colony (ABC) Algorithm and Application Artificial Intelligent Review 42(1) pp. 21-57
[25] L Bambrick 1997 Lecture Timetabling Using Genetic Algorithms (Department of Electrical and Computer Engineering The University of Queensland St. Lucia)
[26] R Cheng, M Gen and Y Tsujimura 1996 Tutorial Survey of job- shop scheduling problems using genetic algorithms Computer and Industrial Engineering 30(4) pp. 983-997
[27] Y Putra 2009 Aplikasi Penjadwalan Perkuliahan menggunakan Algoritma Genetika (Riau Pekanbaru: UIN Sultan Syarif Kasim)
[28] R Qing-dao-er-ji and Y Wang 2012 A new hybrid genetic algorithm for job shop scheduling problem Computer and Operations Research 39(10) pp. 2291-2299
[29] V V Peteghem and M Vanhoucke 2010 A genetic algorithm for the preemptive and non-preemptive multi-mode resource- constrained project scheduling problem European Journal of Operational Research 201(2) pp. 409-418
[30] G Zhang, L Gao and Y Shi 2011 An effective genetic algorithm for the flexible job-shop scheduling problem Expert System with Applications 38(4) pp. 3563-3573
[31] D Karaboga and B Basturk 2007 A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm Journal of Global Optimization 39(3) pp. 459-471

Appendix

Table 12. Description of symbols of ABC algorithm.

| Notation | Information |
|----------|-------------|
| Best     | Result of lecture schedule that has the best fitness value |
| CN       | Cycle number that is used to calculate the initial solution that does not changed in the end iteration and compared with the limit |
| Cumsol   | Cumulative value from Psol (sol=1,2,3,...,n) |
| d        | Lecturers |
| fsol     | Fitness value for best solution in one solitation |
| F        | Penalty value for sol solution |
| h        | Day (h = 1, 2, 3, 4, 5) |
| H        | Empty set that filled by y (1,2,3,...,(jpsg+1)) |
| j        | Hour (j = 1, 2, 3, 4,..., 11) |
| jpsg     | Sum of couple of lectures, courses, and classes |
| k        | Random value (k = 1, 2, 3,..., n; k ≠ sol) |
| m        | Variable that is used for help onlooker bees choose evaluated solution (m = 1, 2, 3,..., n) |
| l        | Limit |
| MCN      | Maximum Cycle Number |
| n        | Sum of population of solution |
| o        | Day value for counting slots of teaching willingness submission (o = 1, 2, 3, 4, 5) |
| p        | Hour value for counting slots of teaching willingness submission (p = 1, 2, 3, 4, 5) |
| Psol     | Probability value for solution |
| r        | Classroom which is available |
| rand     | Initial position value for each couple in solution |
| sol      | Counter the sum of solution that have been iterated (sol = 1,2,3,...,n) |
| t        | Checking willingness slot corresponding with course credit (t=j….(j+course credit)-1). |