Implementation Of Genetic Algorithms For Scheduling The Memorization Of Al-Qur'an At The Markaz Al-Qur'an

Muamar Humaidi Mubin, Budi Arifitama, Yaddarabullah

Abstract: The planning and scheduling stages are the most decisive stages in the success of memorizing the Qur'an. The Qur'an memorization scheduling system compiled at Markaz Al-Qur'an at this time is still not computerized. So it is less effective to do, because there is still a conflicting state of the input data, such as the conflict of availability of time, class, room and the amount of memorization material that is held with the availability of existing religious teachers. This is because scheduling is the stage of dependency between activities as a whole, which is one of the success factors in the implementation of memorization of the Qur'an so that it is completed on time. This thesis applies genetic algorithms to solve optimization problems in scheduling the memorization of the Qur'an. Genetic algorithms represent candidate scheduling solutions into random chromosomes, then evaluate using the fitness function and then do a selection. The selection method used in this study is the roulette wheel selection method, then crossing and mutation are carried out. Each generation, chromosomes are evaluated based on fitness values. After a few generations the genetic algorithm will produce the best chromosome, which is the optimal solution. The advantages of Genetic Algorithms compared to other metaheuristic search optimization are the relatively faster execution time and the minimum use of memory to reach the candidate solution with the best fitness value. The results of the implementation of the Genetic Algorithm as an approach in optimizing Qur'an memorization scheduling results in the achievement of optimal fitness values and is able to produce a schedule of recommendations for activities in memorizing the Qur'an which can be an alternative scheduling decision for memorizing the Qur'an.

Index Terms: Scheduling the Memorization of Qur'an, Genetic Algorithm, Fitness, Selection, Mutation, Crossover.

1 INTRODUCTION

Scheduling in the world of education is very important to note. Basically, teaching and learning activities will run smoothly if the preparation of schedules in accordance with the needs, and conditions in an educational institution. The scheduling process at Markaz Alquran especially regarding managing the memorization of the Qur'an memorization is not computerized. In addition, the current scheduling still takes a long time and data input collision due to many factors that must be considered. Such as the amount of memorization material held, the number of rooms available, the number of religious teachers, as well as the needs of students and religious teachers who support memorization material are also taken into consideration in scheduling. One of the weaknesses in memorizing the Qur'an is to set a time schedule, both in memorizing, repeating memorization, submitting new memorization and when to repeat new memorization. So that the students do not have definite measurement tools to complete memorization at the right time. To overcome these problems, we need a solution by developing a scheduling application as a solution to these problems. The several methods that can be used in solving scheduling problems include Tabu Search (Silaban, Purjani & Nursubiyantoro, 2014), Particle Swarm Optimization (Wati & Rochman, 2013), Genetic Algorithm methods (Setyaningsih, 2014), and the Ant Colony method Optimization (Ashari, Muslim & Alamsyah, 2016). Recent research states that genetic algorithms are a good and effective method of solving scheduling problems (Hari, Putra & Hamdani, 2018). Genetic algorithm is one of the optimization methods that is easily implemented in various case studies, both simple cases and complex cases because it uses the theory of evolution (Priambo, 2015). The use of genetic algorithm methods in this study is expected to help optimize and solve all problems in the process of scheduling the memorization of the Qur'an.

2 LITERATURE REVIEW

This section discusses about various related works already done in data mining clustering techniques and also agriculture dataset related work. Research conducted in 2013 with the title "Comparison of Performance of Genetic Algorithms, Differential Evolution and Optimization of Particle Flocks to Benchmark Functions." and Particle Swarm Optimization based on each measurement with predetermined parameter sizes proving that the GA is proven to be better than DE and PSO. Because GA can get the highest amount of fitness minimum and be better than the other two methods. Then DE performance is faster than PSO, but based on the minimum and average fitness values, PSO is better than DE. a study entitled "A Comparative Study of Crossover Operators for Genetic Algorithms to Solve Job Shop Scheduling Problems." By (Jorge Mendes, 2013), discussed about using the performance of crossover operators (one-point crossover, twopoint crossover, uniform crossover, uniform crossover) and flat crossover in genetic algorithms to overcome scheduling problems. The results of this study state that one-point crossover operators have the best results from other crossover operators, followed by uniform crossover operators. a research entitled "Project Scheduling System Using Genetic Algorithms." . This project scheduling system enables data input operations, performs project scheduling processes automatically and creates optimal schedules. In 2014 a study entitled "Scheduling of the Final Semester Examination with Genetic Algorithms (Case Study of Informatics Engineering Department)." existing boundaries (there is no scheduling conflict) including other data courses, course participants, exam space and time. a study entitled "Resource Allocation Using Metaheuristic Search." The results of this study conclude that the Genetic Algorithm has better high performance and can find it faster to complete Project Management related to Resources and Scheduling Problems with other methods. In 2015 a study entitled "Automatic Scheduling System Using Genetic Algorithms at the Faculty of Science and Technology." based on 99.26%. In 2016 a study entitled "Comparison of Performance of Genetic Algorithms and Ant Colony Optimization in Course Scheduling Optimizing"
by (Imam Ahmad Ashari, Much Muslim & Alamsyah, 2016), explains the better performance comparison between genetic algorithms and ant colony algorithms optimization in completing course schedules and getting the best solutions when computing is done. This study concludes that genetic algorithms get the best solution faster and use less memory compared to the ant colony optimization algorithm when doing computational processes. In 2016 a study entitled "On The Performance Of Different Mutation Operators Of A Subpopulation-Based Genetic Algorithm For Multi-Robot Task Allocation Problems." By (Chun Liu & Andreas Kroll. 2016), explains the comparative performance of mutation operators (random swap, insertion, inversion and displacement) on genetic algorithms in completing Multi-Robot Task Allocation. The results of this study state that the Random Mutation operator has a faster execution time to get the best performance results compared to other Mutation operators. A study entitled "Evaluation of Genetic Algorithm (GA) Performance with Chromosome Improvement Strategies Multi-Choice Case Study Multi-Dimensional Knapsack Problem." Multi-Choice Multi-Dimensional Knapsack Problem using Random Selection, Two point Crossover and Swap Mutation. The results of this study stated that the GA Method with chromosome improvement strategy is able to solve Multi-Choice Multi-Dimensional Knapsack Problems optimally. In 2017 a study entitled "Implementation of Genetic Algorithms for Scheduling ICT Training Instructors at UIN Sunan Kalijaga." (Niki Min Hidayati Robbi & Nurochman, 2017), describes the implementation of genetic algorithms for scheduling ICT training instructors using the Roulette Wheel Selection selection method. Based on the research that has been done, the conclusion obtained is that the genetic algorithm was successfully implemented for scheduling ICT training instructors with a crossover probability parameter (Pc) 0.4, probability (Pm) 0.1, and a total population of 30 individuals. The best fitness value produced is 0.9523 with 1 error value in the class division constraint which has a weight of 0.05. In 2018 a study entitled "Genetic Algorithms for Optimizing the e-KTP Service Satisfaction Classification." (Castaka Agus Sugianto & Tri Herdiawan Apandi, 2018), analyzed the Optimization of e-KTP Service Satisfaction Classification by combining the Naïve Bayes and Genetic Algorithms methods with the selection of the Genetic algorithm. Roulette Wheel Selection has a better accuracy value with an accuracy value of 85.64% compared to the default scheme selection with an accuracy value of 83.03. A study entitled "Scheduling Optimization Using Genetic Algorithm Methods in Annuqayah Vocational High Schools." (Nirwana Haidar Hari, Fauzan Prasetyo Eka Putra & Hamdani, 2018), explains the application of genetic algorithms in optimizing school schedule scheduling in which the Algorithm Approach is Algorithm Approach Genetics in solving school scheduling problems is successful because it is able to find the right combination of scheduling with maximum fitness value, minimum error value (found error value 0), and optimal solutions. Based on the results of previous studies, there are several parts that become the material and references of this research, namely Genetic Algorithms. The research will use the Genetic Algorithm method by using the Roulette Wheel Selection in the selection process, One-Point Crossover in the cross-move process and Random Mutation in the mutation process. So that the scheduling system for memorizing the Qur’an becomes bound and directed.

3 METHODS AND MATERIALS
This section describes the methods and materials needed for this research work. In section A explains about Genetic algorithm. Genetic Algorithm The genetic algorithm begins by defining the optimization variables, the cost function and the cost. A path through the components of the genetic algorithm is shown as a flowchart in figure 1.

![Fig. 1. Flowchart of a GA](image)

3.1 Initial Population
Initial population initialization is a method for producing initial chromosomes. Chromosome data will provide a solution for each experiment. The number of genes that will be raised in accordance with the amount of memorized eye data there are as many as 30 genes. The parameters used by each gene are data in a table that contains memorization, space and time data. So, each chromosome is filled with gene data in the form of memorization data, space and time. The initialization process is carried out by providing initial values of genes with random values according to the minimum and maximum limits that have been determined.

3.2 Evaluation
The fitness function simply defined is a function which takes a candidate solution to the problem as input and produces as output how “fit” our how “good” the solution is with respect to the problem in consideration. Calculation of fitness value is done repeatedly in a GA and therefore it should be sufficiently fast. A slow computation of the fitness value can adversely affect a GA and make it exceptionally slow. In most cases the fitness function and the objective function are the same as the objective is to either maximize or minimize the given objective function.

3.3 Selection
Selection is the process of selecting parents which mate and recombine to create off-springs for the next generation. Parent selection is very crucial to the convergence rate of the GA as
good parents drive individuals to a better and fitter solutions. However, care should be taken to prevent one extremely fit solution from taking over the entire population in a few generations, as this leads to the solutions being close to one another in the solution space thereby leading to a loss of diversity. Maintaining good diversity in the population is extremely crucial for the success of a GA. This taking up of the entire population by one extremely fit solution is known as premature convergence and is an undesirable condition in a GA. The selection method used in this research is Roulette Wheel Selection.

- Calculate \( S = \text{the sum of a finesses.} \)
- Generate a random number between 0 and \( S \).
- Starting from the top of the population, keep adding the finesses to the partial sum \( P \), till \( P < S \).
- The individual for which \( P \) exceeds \( S \) is the chosen individual.

3.4 Crossover

The crossover operator is analogous to reproduction and biological crossover. In this more than one parent is selected and one or more off-springs are produced using the genetic material of the parents. Crossover is usually applied in a GA with a high probability \( - p_c \). The crossover method used in this study is One-Point Crossover: At the crossing of one point (one-point crossover), the crossing process is done by separating the chromosome strings into two parts, then one part is exchanged with one part of another string that is has been separated in the same way. At this stage the exchange of genetic makeup from chromosomes uses a 75% - 100% probability crossover.

3.5 Mutation

Mutation defined as a small random tweak in the chromosome, to get a new solution. It is used to maintain and introduce diversity in the genetic population and is usually applied with a low probability \( - p_m \). If the probability is very high, the GA gets reduced to a random search. Mutation is the part of the GA which is related to the "exploration" of the search space. It has been observed that mutation is essential to the convergence of the GA while crossover is not.

4 RESULT AND DISCUSSION

4.1 Initial Population

Before making a chromosome, the user is required to determine the value of 4 genetic algorithm parameters, including:

1. The number of chromosomes we want to be raised
2. Number of Generations Generated
3. Crossover Probability \( (P_c) \)
4. Probability of Movements \( (P_m) \)

It is known that the number of chromosomes raised is 10 chromosomes, with a maximum generation of 25 generations, 75% crossover probability and 25% mutation probability. Researchers conducted a relational database management system approach to facilitate the process of calculating genetic algorithms for scheduling Qur'an memorization by adjusting the limitations that exist in scheduling memorization of the Qur'an related to memorization details, details of space and time. Following are the overall results of memorization data that will be used in the process of calculating the Qur'an memorization scheduling using genetic algorithms,

<table>
<thead>
<tr>
<th>No</th>
<th>Source Data</th>
<th>ID</th>
<th>Sum of Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Memorizing Data</td>
<td>A (1-30)</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>Space/Room Data</td>
<td>B (1-6)</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>Time Data</td>
<td>C (1-30)</td>
<td>30</td>
</tr>
</tbody>
</table>

The limit value of variables A and C are integer numbers 1 to 30, while the limits of value B are integer numbers 1 to 6 which will be represented in the form of multidimensional arrays. Then the next step is the random value on each chromosome according to the minimum and maximum limits of each gene, while each gene is represented by 3 variables namely Memorize, space and time. Each gene representation in each chromosome is generated according to the minimum and maximum limits of the data in Figure 2 which are represented in the form of multidimensional arrays.

![Fig. 2. Initialization of Chromosome](image-url)
4.3 Selection
To make a selection using the Roulette Wheel Selection method can be done by calculating the value of the chosen probability of each chromosome. Next figure 4 is the result of calculating the probability of each chromosome being selected.

Then generate random numbers [0-1] for each chromosome, i.e. Rj, with j = 1, 2, ..., n, where later a chromosome will be selected if the random number raised is in the interval of the cumulative probability selection between chromosome. It is known that Kj is the jth chromosome.

1. If Rj < PKi, then choose K1 as the jth chromosome parent
2. If PK (i - 1) < Rj < PKi, then select K1 as the jth parent chromosome.

Next figure 6 is the result of generating random numbers for each chromosome,
4.4 Crossover

The individual selection procedure that will go through the crossover process is to generate random numbers $0 < 1$ as much as population size, a probability value that is less than the probability value of the crossover will be chosen as the parent chromosome (parent) to produce offspring chromosomes. Below this is the random number $R$ as many as the population in Figure 8.

![Fig. 8. Random Crossover Chromosome Numbers](image)

Then the $k$ to chromosome will be chosen as the parent if $R[k] < p_c$, from the random number $R$ above, the chromosomes [0], chromosome [1], chromosome [2], chromosome [7], chromosome [8] and chromosome [9] as shown in Figure 9,

![Fig. 9. Parent Crossover](image)

After selecting the parent, the next process is determining the position of the crossover. This process is done by generating random numbers with a limit of 0 to (the total length of the parent chromosome [i]). Then the parent chromosome will be truncated from the gene to a certain. Below this figure 10 is the position of the gene selected as a cut-point for the crossover process.

![Fig. 10. Cut-Point Crossover Position](image)

Then, the $k$ to chromosome will be chosen as the parent if $R[k] < p_c$, from the random number $R$ above, the chromosomes [0], chromosome [1], chromosome [2], chromosome [7], chromosome [8] and chromosome [9] as shown in Figure 9,

After selecting the parent, the next process is determining the position of the crossover. This process is done by generating random numbers with a limit of 0 to (the total length of the parent chromosome [i]). Then the parent chromosome will be truncated from the gene to a certain. Below this figure 10 is the position of the gene selected as a cut-point for the crossover process.

![Fig. 11. Crossover between Parent](image)

In this mutation process it is very possible the emergence of new chromosomes which had not yet appeared in the initial population. Following is the procedure for selecting genes to be mutated by counting the total number of genes on the chromosome. Next figure 13 is the process of calculating total genes,

\[
\text{Jumlah Gen per Kromosom} = 30 \\
\text{Jumlah Populasi} = 10 \\
\text{Total Gen} = \text{Jumlah Gen per Kromosom} \times \text{Jumlah Populasi} = 30 \times 10 = 300 \\
\]

**Fig. 12. Crossover Chromosomes**

**4.5 Mutation**

In this mutation process it is very possible the emergence of new chromosomes which had not yet appeared in the initial population. Following is the procedure for selecting genes to be mutated by counting the total number of genes on the chromosome. Next figure 13 is the process of calculating total genes,

\[
\text{Jumlah Gen per Kromosom} = 30 \\
\text{Jumlah Populasi} = 10 \\
\text{Total Gen} = \text{Jumlah Gen per Kromosom} \times \text{Jumlah Populasi} = 30 \times 10 = 300 \\
\]

**Fig. 13. Calculation of Total Gene**

Calculate the total mutation. The following figure 14 is the process of calculating total mutations where the mutation probability is 25% of the total gen,

\[
\text{Total Mutasi} = \text{Probabilitas Mutasi} \times \text{Total Gen} = 0.25 \times 300 = 75 \\
\]

**Fig. 14. Calculation of Total Mutation**

Then selecting the position of genes to do the mutation process, to choose the position of genes that have mutations can be done by generating random integer numbers between 1 to total gen, 1 to 300 represented in the form of an array [0 - 299]. Next figure 15 is the gene chosen for the mutation process,

\[
118 = [3, 27] \\
215 = [7, 4] \\
87 = [2, 26] \\
243 = [8, 2] \\
246 = [8, 5] \\
66 = [2, 5] \\
52 = [1, 21] \\
34 = [1, 3] \\
101 = [3, 10] \\
176 = [5, 25] \\
85 = [2, 24] \\
5 = [0, 4] \\
113 = [3, 22] \\
11 = [0, 10] \\
242 = [8, 1] \\
253 = [8, 12] \\
180 = [5, 29] \\
142 = [4, 21] \\
228 = [7, 17] \\
110 = [3, 19] \\
238 = [7, 27] \\
182 = [6, 1] \\
161 = [5, 10] \\
197 = [6, 16] \\
213 = [7, 2] \\
142 = [4, 21] \\
186 = [6, 5] \\
48 = [1, 17] \\
113 = [3, 22] \\
157 = [5, 6] \\
96 = [3, 18] \\
139 = [4, 18] \\
97 = [3, 6] \\
128 = [4, 7] \\
283 = [9, 12] \\
55 = [1, 24] \\
188 = [6, 7] \\
213 = [7, 2] \\
254 = [8, 13] \\
192 = [6, 11] \\
267 = [8, 26] \\
57 = [1, 26] \\
96 = [3, 5] \\
48 = [1, 17] \\
60 = [1, 29] \\
127 = [4, 6] \\
34 = [1, 3] \\
25 = [0, 4] \\
19 = [0, 18] \\
91 = [3, 0] \\
1 = [0, 0] \\
73 = [2, 12] \\
26 = [0, 25] \\
183 = [6, 2] \\
255 = [8, 14] \\
22 = [0, 21] \\
26 = [0, 25] \\
58 = [1, 27] \\
19 = [0, 18] \\
289 = [6, 28] \\
256 = [8, 15] \\
40 = [1, 9] \\
203 = [6, 22] \\
58 = [1, 27] \\
19 = [0, 18] \\
289 = [6, 28] \\
256 = [8, 15] \\
40 = [1, 9] \\
203 = [6, 22] \\
58 = [1, 27] \\
19 = [0, 18] \\
289 = [6, 28] \\
256 = [8, 15] \\
40 = [1, 9] 
\]

**Fig. 15. Selected Genes for Mutations**

Then the selected Genes will be replaced with integer random numbers according to the limitation of the representation of the existing genes. The following figure 16 is an example of a gene that will mutate,
Next figure 17 is the result of chromosomes when the mutation process is carried out.

**Fig. 17.** Chromosomes from Mutations

Chromosomes will undergo the same process as in previous generations, namely the process of evaluation, selection, crossover and mutation which will then produce new chromosomes for the next generation. This process will be repeated for a number of generations that have been predetermined. After 3 generations, the best chromosome is the chromosome in the 3rd generation chromosome index [9] with fitness value 1 as shown in Figure 18:

**Fig. 18.** Best Chromosomes

Figure 20 is the result of the fitness calculation of each chromosome in the 3rd generation, where the genetic algorithm has obtained the Qur'an memorization scheduling data that meets the constraint (there is no schedule conflict either memorization, space/room and time) this can be known from the fitness value on the chromosome index to [9] which achieved the best fitness value 1.

**Fig. 19.** 3rd generation chromosomes

**Fig. 20.** Best Chromosome Fitness

Below is Figure 19 which contains detailed information on chromosomes in the 3rd generation:
5. CONCLUSION
Based on the results of scheduling of memorization of Qur'an using Genetic Algorithm, it can be concluded that the Genetic Algorithm Approach in solving the problem of Constraint Scheduling Memorization Al Qur'an uses genetic operator input measuring instrument by distinguishing the number of chromosome generation (10, 50 and 100), generation (25 and 50), 75% crossover value and 25% mutation, concluded that the Qur'an memorization scheduling can be successful because it is able to find the right combination of scheduling with maximum fitness value, and the optimal solution (no collision of Qur'an memorization scheduling).

6 ACKNOWLEDGMENT
On this occasion, the authors would like to express their sincere thanks and highest appreciation to all those who have helped the author: Budi Arifitama, ST., MMSi. as a supervisor I, Yaddarabullah, S.Kom., M.Kom. as the supervisor lecturer II, Mr. Ridwan Nurzaman Al Hafidz as the Coordinator and Teacher of the Markaz Qur'an who has given permission and the opportunity for the writer to research the Markaz Qur'an that he is teaching, Family and Maryam Bint Salim who has provided knowledge and time so that writers can finish this research, as well as the teachers and administrative staff who have helped many writers in conducting this research.

7 REFERENCES