The Impact of Mutation Probability on Genetic Algorithm Performance in Optimizing Course Scheduling

Rudi Salman[1], Irfandi[2], Suprapto[3], Sayuti Rahman[4], Herdianto[5]
{rudisalman@unimed.ac.id[1], irfandi@unimed.ac.id[2], suprapto@unimed.ac.id[3], sayutirahman01@gmail.com[4], herdianto05@gmail.com[5]}

Department of Electrical Engineering, Universitas Negeri Medan, Medan, North Sumatera[1]
Departement of Physics, Universitas Negeri Medan, Medan, North Sumatera[2]
Departement of Machine Engineering, Universitas Negeri Medan, Medan, North Sumatera[3]
Departement of Information System, Universitas Harapan Medan, Medan, North Sumatera[4]
Department of Information Technology, Universitas Panca Budi Medan, North Sumatera[5]

Abstract. Computation time plays a crucial role in determining the speed of the genetic algorithm (GA). Critical parameters such as population size, crossover probability (Pc), mutation probability (Pm), and selection significantly influence the time required for the GA to find the optimal solution. Among these parameters, Pm is particularly critical as it directly impacts the mutation process of the parent chromosomes, indicating the importance of the parent chromosomes undergoing mutation. Consequently, selecting the correct Pm value is vital to ensuring the algorithm's efficiency of the mutation process.
To examine the effect of Pm on GA performance, a series of simulations were carried out by varying the Pm values from 0.01 to 0.1 while keeping other parameters constant (Pc = 0.85 and population size = 100). The simulations, performed using Matlab R2012b, revealed that a Pm value of 0.06 resulted in the fastest computation time, averaging 0.382 seconds. This suggests that optimizing the scheduling for the electrical engineering program at Universitas Negeri Medan, a Pm value of 0.06, provides the most efficient computational performance.

Keywords: Mutation probability, computing time, optimization, genetic algorithm, scheduling course

1 Introduction

The genetic algorithm (GA) is a search technique inspired by natural selection and genetic principles. It relies on three core elements: population size, crossover probability (Pc), and mutation probability (Pm).The important operator for creating new offspring from the selected parent chromosome pair is Pc [1],[2]. These operators function to produce optimal solutions [3]. In addition, the genetic algorithm iterates over the initial population to produce a new population that can then be optimized [4].
The basic structure of the genetic algorithm is as follows:
1. Population Initialization:

ICIESC 2024, September 17, Medan, Indonesia
Copyright © 2025 EAI
DOI 10.4108/eai.17-9-2024.2352979

The initial step in the genetic algorithm involves generating an initial population. This population comprises a collection of individuals (chromosomes) that represent potential problem-solving solutions. Typically, individuals are initialized to ensure genetic variation within the initial population.

2. Fitness Evaluation:

Every member of the population is assessed with a fitness function, which evaluates how well each individual addresses the encountered problems. A higher fitness score signifies a more effective solution.

3. Selection:

The selection process aims to choose individuals who will be parents (parents) for the next generation. The commonly used selection methods include:

Roulette Wheel Selection: Individuals are chosen probabilistically based on their fitness value. Individuals with higher fitness have a more significant opportunity to be selected.

Tournament Selection: A group of individuals is randomly chosen, and those with the best fitness are selected as parents.

4. Crossover:

Following the selection process, the chosen individuals (parents) are paired to create new individuals. Crossover merges the genetic traits of two parents to generate one or more offspring. Common crossover techniques include:

Single-point Crossover: A random point is selected along the chromosome, and the chromosome parts after that point are exchanged between the two parents.

Multi-point Crossover: Some random points are selected, and the chromosome segments are exchanged between these points.

5. Mutation:

The mutation process adds extra variations by randomly altering one or more genes in the chromosomes of individuals. Mutations play a crucial role in preventing premature convergence by preserving genetic diversity within the population. Typically, mutations occur with a low probability.

6. Replacement:

The existing population is replaced with a new one of offspring and possibly a few top individuals from the previous generation (elitism). This cycle continues until termination criteria are met, which may include a specific number of generations or achieving a desired fitness level.

Certain operations contribute to the effectiveness of the genetic algorithm in identifying optimal solutions. These include generating the initial population, calculating the fitness value for each chromosome, selection, crossover, and mutation. Mutation is one of the key processes in evaluating the algorithm [5]. The mutation process changes genes in one chromosome to increase diversity, which is determined by the value of mutation probability [6]. Each chromosome's fitness value will be assessed to gauge its quality. As the process is repeated, chromosomes with the highest fitness values will emerge. These chromosomes are more likely to reproduce and generate new chromosomes in the next generation. Consequently, future generations will outperform their predecessors, illustrating the effectiveness of the genetic algorithm.

Course scheduling entails assigning resources at specific times while adhering to certain constraints. One of the initial tasks at the beginning of the semester is to develop a course timetable. However, scheduling classrooms can lead to various issues, such as conflicts that

prevent lecturers from teaching due to overlapping class schedules and instances where multiple courses are assigned to the same room at the same time, among other challenges.
To create the desired course schedule, efforts will be made to optimize the timetable using a genetic search algorithm. The mutation of genetic operators will enhance the diversity of individuals in the subsequent population, resulting in new individuals that are superior to their parent chromosomes. The genetic algorithm relies on this operator to find optimal values. Mutations improve genetic diversity within a chromosome, differentiating it from the parent chromosome and enhancing its quality [7]. The impact of the Pm value on GA performance is assessed by varying Pm. Changes in the value of Pm will be observed to affect computational time. This computing time determines GA performance. Therefore, the selection of the appropriate Pm value will significantly determine the performance of the GA significantly[8].

2 Method

Figure 1 shows a flowchart demonstrating the process of applying genetic algorithms in scheduling simulations. The initial step involves generating a starting population based on a graph diagram. The second step is to assess the fitness of each population member. In the third step, parent chromosomes are selected. The fourth step involves crossover and mutation of the parent chromosomes. Lastly, the fifth step evaluates the child's chromosome against the desired criteria. If it meets the criteria, the process is complete; if not, the procedure returns to the initial step.

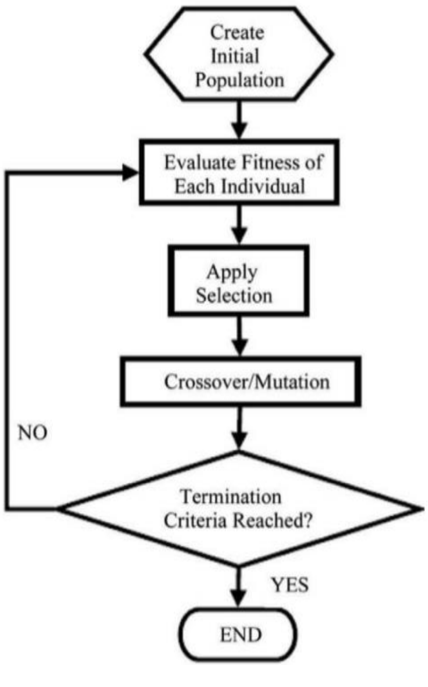

Fig.1. Flowchart of Genetic Algorithm

3 Results and Discussion

Refer to **Figure 2**, the simulation results showing the effect of Mutation Probability (Pm) on the performance of the Genetic Algorithm in scheduling optimization were obtained by adjusting the mutation probability (Pm) to 0.06, with a crossover probability of 0.85 and a population size of 100. This configuration yielded a computation time of 0.382 seconds, representing the fastest time achieved.

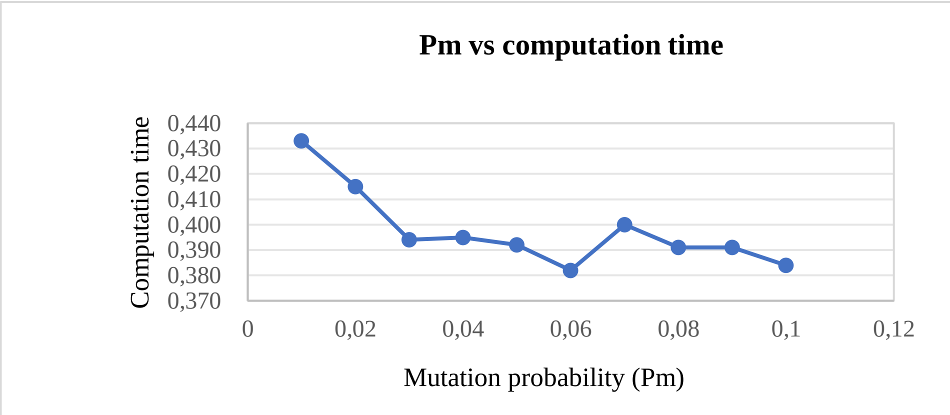

Fig.2. Computing Time with variation of the value of mutation probability

This research examines the computational speed of genetic algorithms by varying one parameter: the mutation probability value. The selected mutation probability influences the percentage of mutations occurring in the parent chromosomes, affecting computing time. Other factors, such as population size and selection method, significantly impact the computing time of genetic algorithms, but they are not addressed in this study.

4 Conclusion

Based on the research findings, it can be concluded that a mutation probability (Pm) of 0.06, a crossover probability of 0.85, and a population size of 100 results in a computation time of 0.382 seconds in the course scheduling optimization process for the Electrical Engineering Study Program at Unimed.

Acknowledgments

Special thanks are extended to the Lembaga Penelitian dan Pengabdian Masyarakat Universitas Negeri Medan for providing the research funding, as outlined in the Decree of the Chairman of LPPM UNIMED, Number 0147/UN33.8/PPKM/PD/2024. This support made completing the research within the allotted time frame possible.

References

[1] Wahyuningsih.,et.al.,Penerapan Algoritma Genetika untuk optimasi Penjadwalan pada MTS Negeri 1 Pangkal Pinang, Jurnal SisFokom., Vol.09,No.3, 2020.

[2] Ma'arif,et.al., Penerapan Metode Algoritma Genetika untuk Optimasi Penjadwalan Mata Kuliah, SemNas. Inov. Teknologi UN PGRI Kediri, pp. 93–97, 2018..

[3] Victor,et.al.,Optimasi Penjadwalan Mata Pelajaran Menggunakan Algoritma Genetika (Studi Kasus :SMKN 7 BALENDAH)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 1, no. 10, pp. 1066–1072,2017.

[4] Puspita,et.al.,Pengembangan Aplikasi Penjadwalan Kegiatan Pelatihan Teknologi Informasi Dan Komunikasi Dengan Algoritma Genetika (Studi Kasus: Bprtik),*J. Online Inform.*, vol. 1, no. 2, pp. 76–81, 2016, doi: 10.15575/join.v1i2.43

[5] Ni Luh Gede Pivin,et.al.,Penerapan Algoritma Genetika Untuk Penjadwalan Mata Pelajaran, *J. Appl. Intell. Syst.*, vol. 1, no. 3, pp. 220–233, 2016.

[6] Toscany,et.al., Pengembangan Sistem Penjadwalan Kuliah Menggunakan Algoritma Genetik (StudiKasus : Pascasarjana Universitas Jambi), *J. Manaj. Sist. Inf.*, vol. 2, no. 2, pp. 379–393, 2017

[7] Saptarini,et.al., Comparing Selection Method in Course Scheduling Using Genetic Algorithm, vol.1, no.1, Icst, pp. 574–578, 2018

[8] Salman,et.al., Pengaruh Probabilitas Crossover Terhadap Kinerja Algoritma Genetika Dalam Optimasi Penjadwalan Matakuliah, Jurnal Teknoif Teknik Informatika InstitutTeknologi Padang - Vol.11No.2, pp. 69-74, 2023