Comparison between genetic algorithm with Differential Evolution in study scheduling

N Lukman*, M Irfan, A Nugraha and J Jumadi
Department of Informatics engineering UIN Sunan Gunung Djati Bandung, Bandung, Indonesia

*n.lukman@uinsgd.ac.id

Abstract. This paper proposed to discuss the complexity of scheduling by comparing two optimization methods between genetic algorithms with differential Evolution. Genetic Algorithms can solve the simplest to complex problems as well. Therefore, the Genetic algorithm is precisely applied to the scheduling of subjects. Then another appropriate optimization method for completing optimization is the Differential Evolution (DE) algorithm. DE algorithm is a fast and effective search algorithm in solving numerical and finding optimal global solutions. The steps of the two algorithms are initialization, participation, mutation, crossover, and selection. The scheduling system produces non-optimal schedules for teacher conflicts and empty slot schedules. After the genetic algorithm and differential evolution are applied, an analysis of the results of the subject scheduling is then performed by comparing the fitness values and the execution speed of the two algorithms. the genetic algorithm found only 2 perfect schedules out of 10 experiments, whereas in the implementation of differential algorithms, there are 7 perfect schedules out of 10 experiments. Thus, it can be concluded that by determining the value of the producing parameters 5, generation 50, mutation 0.6, and crossover 0.2, the differential evolution produces better output or conformity values using genetics.

1. Introduction
Scheduling in an educational institution is a routine work carried out at every beginning of the new school academic year, Whether in school or college [1–4]. In a school institution, arranging a course schedule is a routine job done by the curriculum section of every coming new school academic year.

According to the English great dictionary, Schedule is a division of time-based on the plan of work arrangement, list, table of activities, or plan activities with a detailed implementation time distribution. The scheduling is the process, way, scheduling activities, or enter a schedule [5]. Scheduling is defined as the process of allocating resources to select a set of tasks within a certain period [6]. In detail, it can be explained that scheduling is a function of decision making, determining the most appropriate schedule, or is a theory containing a set of principles, model, techniques in decision making [7]. Scheduling issues often occur in time clashes, and in the process that takes a long time to Among the right ways to solve problems in scheduling are using optimization methods [8]. The optimization method will be able to provide good results. The method of optimization requires a good strategy in deciding to obtain optimum results [4,9].

Of the many optimization methods that can solve various problems optimization is the algorithm of genetics. The genetic algorithm can solve the simplest and most complex problems though. Therefore,
the genetic algorithm can be applied to the scheduling of subjects. The genetic algorithm is a search algorithm based on natural selection mechanisms and natural genetics [9–13]. Genetic algorithms have proved their efficiency in resolving Non-Polynomial problems [6].

Another algorithm included in the optimization algorithm is the Differential Evolution algorithm (DE). Algorithm DE is a search algorithm that has the capability of being the optimum global optimization method effective. Algorithm DE is a fast and effective search algorithm to solve numerical problems and find the optimum global solution [4,14–19]. Because the two methods of the above optimization algorithm can be effective to compile the scheduling, then research into the performance efficiency comparison of both algorithms against a system of subjects’ schedule.

2. Research method
Research activities depicted in Figure 1, Starting from data collection, data initialization, Understand the business process of making schedules, and Implementing it in Genetic and Differential evaluation methods, the latter test and compare both algorithms used.

![Figure 1. Research activity.](image)

2.1. Genetic algorithm
The Genetic algorithm performs optimal solution search, the search process is conducted among several optimal point alternative based on probabilistic function. The initial population is randomly generated while the next population of the evaluation results and the best chromosomal selection. The process of scheduling using a genetic algorithm consists of several processes that must be undertaken namely defining individuals, fitness value, early population generation, Conducting the selection and the cross-marriage process, and mutation of genes to be used [14].

The genetic algorithm in the first stage will generate the initial population, the next step calculates the value of the evaluation function resulting from population generation. If the value of the evaluation function matches the criteria, the process is complete. If not, a selection, mutation, and crossover process are carried out to find a value that meets the criteria.

2.2. Differential evolution algorithm
This algorithm uses several population size parameters (NP), mutation (F) operators, Crossover operators (CR), and iteration count (generation). These parameters are specified to be initialized. Then, the process of mutation, crossover, and selection. The calculation process occurs continuously until the number of iterations is determined. When the process is complete, the last schedule is a schedule that has a fitness value greater than every iteration made [15,19].

The Differential Evolution algorithm is the first to include the parameter data value, then generate the population, then process mutation, crossover, selection. When the value meets the criteria, the process is completed. Otherwise, it will re-enter the mutation, crossover, and selection process until it finds a value that meets the criteria.

3. Results and discussion
In this section, it is explained the results of research and at the same time is given the comprehensive discussion about Scheduling using genetic algorithms and differential evolution.
3.1. Implementation of genetic algorithm

The Implementation of the genetic algorithm begins with defining individual, fitness value, initial population generation, conducting selection and the cross-moving process and mutation of genes to be used.

3.1.1. Data initialization and determining the beginning population. The process of encoding input data on schedule is a unification of scheduling aspects itself or generations (teachers, space, time, and subjects) specified by the Admin user, early population generation is then done by initializing the possible solution into several chromosomes. The length of one chromosome is determined based on the problems examined.

The solution to be produced is to determine the time and space for teaching and learning activities. The length of the chromosome is based on the number of genes stating the number of classes is 9 classes.

3.1.2. Selection. The formation of individual structures in a new population is done using the Roulette-wheel selection method. As the name suggests, this method impersonates the game Roulette-wheel where each chromosome occupies a circular piece on the Roulette-wheel proportionally according to the value of the suit and obtained the best fitness value of each calculation test shown in the following table.

| Number of Test | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---------------|---|---|---|---|---|---|---|---|---|----|
| The best fitness value | 1 | 1 | 0.25 | 0.5 | 1 | 0.25 | 0.33 | 0.2 | 0.125 | 0.25 |
| Gen           | 26 | 47 | 40 | 27 | 32 | 44 | 46 | 50 | 26 | 47 |

3.1.3. Crossover. Crossover is used as the individual cutting method randomly and is the merging of the first part of an individual parent 1 with the second part of an individual parent 2. Crossover can be performed only if a random number is raised for an individual less than the probability of a crossover (Pc) specified. Here are illustrations of the crossover algorithm genetic mechanisms:

| Individual Crossover Results |
|-----------------------------|
| M4 | A1c | N5 | H6 | L18 | P9 |
| B1b | K3 | J3 | M4 | B1b | M4 |
| O6 | L18 | J3 | C11 | O6 | K3 |
| K3 | M4 | L18 | H6 | M4 | J3 |
| L18 | L18 | O6 | E1b | C11 | H6 |

3.1.4. Mutation. The mechanism of mutation is to replace one or several genes from a chromosome into another gene that is randomly arranged. The mutation process will be possible when the mutation value qualifies as a mutation rate. The assumed mutation rate of 0.6. If the individual multiplier results are less than or equal to 0.6. Then the individual fulfills the requirements for mutation. If a chromosome qualifies, the next step is to randomize the location of the gene series. Compared to the result of the multiplier is the 2nd sequence, then the mutation treat is done for the gene with the 2nd column number. The gene sequence that is mutated will be removed and replaced with a new gene sequence.

3.2. Implementation of differential evolution algorithm

3.2.1. Population initializes. The first step in determining the initialization is determining parameters NP = 5, F = 0.6, Cr=0.2, JI=50, and then those parameters will be used in calculations. Once the
parameter is specified, determines the initial population perform the calculation, on this algorithm population size states the number of individuals. The individual in case of scheduling is a single piece of entity full of a schedule in a one-week cycle schedule, in each individual, there is a component that compiles an individual. The component itself consists of teacher data, subject data, spatial data, time data, and day data. Then selected a random schedule consisting of 5 individual pieces based on population size, of which there are 5 schedules of lessons searched randomly. For the generation of early population values are searched randomly as follows:

\[
\text{threshold} + \text{random numbers} \times (\text{Upper limit} – \text{threshold}) \\
\text{where: threshold} = 0 \text{ and Upper Limit} = 24
\]

The calculation is done by calculating each variable from Monday to Saturday so that the chromosome with the best fitness value is produced among the other chromosomes.

3.2.2. Mutation. The mutation process is the first step of the chromosome evolved, in this process is the difference in the value of the values or the difference between two random vectors (random vector r1 and random vector r2). Then both vectors are multiplied by the mutation operation (F) which is the value of input parameters. To perform a mutation calculation that will generate a mutant population that is looking for 3 random individuals who are not equal and are not equal to the selected individual. In this process is calculated by the formula:

\[
Y = \text{random vector r0} + F \times (\text{random vector r1} – \text{random vector r2}).
\]

3.2.3. Crossover. After the mutant population is established, the Next process is the establishment of a trial population. Each individual gene trial comes from a selected individual gene or a mutant population gene. To select a gene from a selected population of a mutant or individual gene, use a random number between 0 and 1 as well as a crossover operator of 0.2. If the selected random number is worth less than 0.2 then the gene value for the trial population is derived from the mutant population gene. However, if the random number value is more than 0.2 then the gene value for the trial population comes from the selected individual gene.

3.2.4. Selection. The selection process is performed between two vectors that are selected vector trials and individual vectors. If the trial vector has a fitness value greater than the selected individual vector, then the trial vector will replace the selected individual vector position in the population in the next generation. If otherwise, the selected individual vector will remain in position in the population.

3.3. Test performance of genetics and differential evolution algorithm.

Testing the performance was done by experimenting 10 times the test, with parameter input user.

| No | Generation | Chromosome | Crossover Rate | Mutation Rate | Fitness | Time/ seconds | Results      |
|----|------------|------------|----------------|---------------|---------|---------------|--------------|
| 1  | 50         | 5          | 0.6            | 0.2           | 0.5     | 28,26         | Not successful |
| 2  | 50         | 5          | 0.6            | 0.2           | 0.25    | 28,02         | Not successful |
| 3  | 50         | 5          | 0.6            | 0.2           | 1       | 35,47         | Successful    |
| 4  | 50         | 5          | 0.6            | 0.2           | 1       | 25,29         | Successful    |
| 5  | 50         | 5          | 0.6            | 0.2           | 0.5     | 28,16         | Not successful |
| 6  | 50         | 5          | 0.6            | 0.2           | 0.2     | 33,31         | Not successful |
| 7  | 50         | 5          | 0.6            | 0.2           | 0.5     | 31,40         | Not successful |
| 8  | 50         | 5          | 0.6            | 0.2           | 0.333   | 27,87         | Not successful |
| 9  | 50         | 5          | 0.6            | 0.2           | 0.25    | 26,42         | Not successful |
| 10 | 50         | 5          | 0.6            | 0.2           | 0.5     | 29,43         | Not successful |

Table 3. Test results of a genetic algorithm.
Table 4. Test results of a differential evolution algorithm.

| No | Generation | Chromosome | Crossover Rate | Mutation Rate | Fitness | Time/ seconds | Results     |
|----|------------|------------|----------------|--------------|---------|---------------|-------------|
| 1  | 50         | 5          | 0.6            | 0.2          | 0.5     | 67,42         | Not successful |
| 2  | 50         | 5          | 0.6            | 0.2          | 1       | 42,85         | Successful   |
| 3  | 50         | 5          | 0.6            | 0.2          | 1       | 64,32         | Successful   |
| 4  | 50         | 5          | 0.6            | 0.2          | 0.5     | 58,90         | Not successful |
| 5  | 50         | 5          | 0.6            | 0.2          | 1       | 65,00         | Successful   |
| 6  | 50         | 5          | 0.6            | 0.2          | 1       | 64,59         | Successful   |
| 7  | 50         | 5          | 0.6            | 0.2          | 0.5     | 60,09         | Not successful |
| 8  | 50         | 5          | 0.6            | 0.2          | 1       | 30,74         | Successful   |
| 9  | 50         | 5          | 0.6            | 0.2          | 0.25    | 26,42         | Not successful |
| 10 | 50         | 5          | 0.6            | 0.2          | 0.5     | 29,43         | Not successful |

Figure 2. Comparison of fitness values genetic algorithm with differential evolution.

4. Conclusion
In the implementation of the genetic algorithm only found 2 perfect schedules of 10 experiments, while in the implementation of the differential evolution algorithm, there are 7 perfect schedules of 10 experiments. Thus, it can be concluded that with the value of the population parameters 5, Generation 50, mutation 0.6, and crossover 0.2 then the differential evolution algorithm produces the output or fitness value better than the genetic algorithm.

References
[1] Irfan M and Lukman N 2012 Scheduling Lecturesoftware Using Web-Based Framework Codeigniter (a Case Study: Faculty of Science and Technology Un Sunan Gunung Djatibandung) Ed. Juli
[2] Hijriana N 2015 Application of Genetic Algorithm Method for Nurse Scheduling Problems Info Tek. 16 61–74
[3] Janata A, Janata A and Haerani E 2015 Outsourcing Scheduling System Using Genetic Algorithms (Case Study: PT. Syarikatama) J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf. 1 17–24
[4] Kusmarna I, Wardhani L K and Safrizal M 2015 Scheduling Application Using Particle Swarm Optimization (PSO) Algorithm J. Tek. Inform. 8
[5] Indonesia K B B 2012 Meaning of the word time table KBBI
[6] Desiana E 2016 Performance Algoritma Genetika (GA) pada Penjadwalan Mata Pelajaran InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)
[7] Damayanti L and Cholissodin I 2018 Optimization of Thesis Guidance Scheduling Using Genetic Algorithms (Case Study: Faculty of Computer Science, Universitas Brawijaya) J. Pengemb. Teknol. Inf. dan Ilmu Komput.
[8] Li G, Zhou H, Jing X, Tian G and Li L 2017 An intelligent wheel position searching algorithm for cutting tool grooves with diverse machining precision requirements Int. J. Mach. Tools Manuf. 122 149–60
[9] Hay’s R N 2017 Combination of Firefly Algorithm-TabuSearch for Resolving Traveling Salesman Problem J. Online Inform. 2 42
[10] Makhdoomi S and Askarzadeh A 2020 Optimizing operation of a photovoltaic/diesel generator hybrid energy system with pumped hydro storage by a modified crow search algorithm J. Energy Storage 27 101040
[11] Mirjalili S 2019 Genetic algorithm Studies in Computational Intelligence
[12] Goldberg D E 1989 Genetic Algorithms in Search, Optimization, and Machine Learning. Choice Rev. Online
[13] Kumar M, Husian M, Upreti N and Gupta D 2010 Genetic Algorithm: Review and Application Int. J. Inf. Technol.
[14] Eko Wiratno S, Rudi N and Santosa B 2012 Algoritma Differential Evolution untuk Penjadwalan Flow Shop Banyak Mesin dengan Multi Obyektif J. Tek. Ind.
[15] Kasus S, Paris P T and Mandiri J 2019 Differential Evolution (DE) Algorithm in Optimizing Nestle Product Distribution Routes Semin. dan Konf. Nas. IDEC 2–3
[16] Nugraha A, Nugraha A A and Mauludiyanto A 2015 Optimasi Peletakan Base Transceiver Station Di Kabupaten Mojokerto Menggunakan Algoritma Differential Evolution J. Tek. ITS 4 A42–7
[17] Ardia D, Boudt K, Carl P, Mullen K M and Peterson B G 2011 Differential evolution with deoptim R J.
[18] Das S and Suganthan P N 2011 Differential evolution: A survey of the state-of-the-art IEEE Trans. Evol. Comput.
[19] Price K V 2013 Differential Evolution Intell. Syst. Ref. Libr.