

IMPLEMENTATION OF GENETIC ALGORITHM BASED ON JAVASCRIPT IN OBJECT ROUTING SHORTEST TOUR ON TIMOR ISLAND

Justin Eduardo Simarmata^{1*}, Ferdinandus Mone²

^{1,2}Mathematics Education Study Program, Faculty of Science Education, University of Timor
Jl. Eltari, Km. 09, Kefamenanu, 85613, Indonesia

Corresponding author's e-mail: *justinesimarmata@unimor.ac.id

ABSTRACT

Article History:

Received: 25th November 2022

Revised: 8th February 2023

Accepted: 18th Februari 2023

Keywords:

Genetic Algorithm;

Shortest Route;

Timor Island



This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution-ShareAlike 4.0 International License.

How to cite this article:

J. E. Simarmata and F. Mone, "IMPLEMENTATION OF GENETIC ALGORITHM BASED ON JAVASCRIPT IN OBJECT ROUTING SHORTEST TOUR ON TIMOR ISLAND" *BAREKENG: J. Math. & App.*, vol. 17, iss. 1, pp. 0545-0558, March 2023.

Copyright © 2023 Author(s)

Journal homepage: <https://ojs3.unpatti.ac.id/index.php/barekeng/>

Journal e-mail: barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id

Research Article • **Open Access**

1. INTRODUCTION

The charm of Indonesian tourism is attractive, both for domestic and foreign tourists. Natural wealth and historical heritage in Indonesia are the main attractions for tourists, besides that there are still artificial tourist attractions that are also no less sought after by tourists. Timor Island located in East Nusa Tenggara Province (NTT) offers the natural charm of beaches, mountains, and waterfalls. Timor Island consists of several districts including Kupang City, Kupang Regency, South Central Timor Regency, North Central Timor Regency, Belu Regency, and Malaka Regency. Existing attractions are evenly distributed throughout the territory on the island of Timor. The location of tourist attraction destinations that are varied but far apart is an obstacle for tourists to make their choice of attractions. Especially if tourists are traveling for the first time on the island of Timor, it is difficult to plan a tourist trip.

Tourism is a travel activity carried out by a person or group of people who visit certain places for several purposes such as recreation or studying the uniqueness of the tourist attractions visited. Problems that often arise are the lack of information about the route to tourist attractions, the level of density of the route to be passed and the volume of the route due to limited tourist time, therefore finding the shortest route is important. The search for the shortest route is how to determine the most optimal route, the route with the shortest route at the smallest cost, in its application it is useful to know the shortest route, increase the effectiveness of providing travel information, the need to provide several alternative lanes for road users and driving [1]; [2]. Finding the most optimal route requires a search method to solve the problem [3]. The tourist travel route optimization system that is often used is the Genetic Algorithm because by applying the Genetic Algorithm it can solve problems in a more stable execution time with varying complexity than conventional methods [4]; [5].

Genetic algorithms are one of the soft computing models and a branch of evolutionary algorithms, a method used to solve a search for value in an optimization problem [6]. Genetic algorithms are also defined as algorithms that apply an understanding of natural evolution to problem-solving efforts [7]. Soft Computing is a model approach to computing by imitating human reason and having the ability to reason and learn in an environment full of uncertainty [8]. In genetic algorithms there are three important parameters that must be defined, namely population size, probability of cross-moving, and probability of mutation. Chromosomes are representations of solutions. Genetic operators consisting of crossovers and mutations can be carried out both or only one of them, then the evolutionary operator is carried out through the chromosomal selection process of the parent and offspring (derived generation) to form a new generation that is expected to be better at estimating the optimum solution, the iteration process then continues according to the number of generations that have been set [9]; [10]; [11]; [12].

The approach taken by genetic algorithms is to randomly combine the best solutions selected in a group to get the next generation of the best solution, namely in a condition that maximizes compatibility which will be called fitness [13]; [14]. The chosen generation is the generation that represents improvements in the previous population or group. By performing a repetitive process, this algorithm can get the most appropriate solution to the problem at hand [15]; [16].

To apply genetic algorithms, the solution of the problem is represented into the form of chromosomes. According to [17] there are three important aspects to applying this genetic algorithm so that bias works well, these aspects are as follows: definition of fitness functions, definition and implementation of genetic representations, definition and implementation of genetic operations.

The variables and parameters used in genetic algorithms are:

1. The fitness function (goal function) owned by each individual to determine the degree of conformity of the individual to the criteria to be achieved.
2. The population numbers the number of individuals involved in each generation.
3. Probability of crossovers in a generation.
4. The probability of occurrence of mutations in each individual.
5. The number of generations to be formed that determines the duration of application of genetic algorithms.

In general, the structure of a genetic algorithm according to [18] can be defined by the following steps:

1. Generating the initial population

This early population is randomly generated so that an initial solution is obtained. The population itself consists of a number of chromosomes that represent the desired solution.

2. Forming a new generation

The formation of a new generation uses reproduction/selection, crossover and mutation operators. This process is carried out repeatedly so that a sufficient number of chromosomes are obtained to form a new generation where this new generation is a representation of a new solution. This new generation is known as offspring.

3. Solution evaluation

In each generation, chromosomes will go through an evaluation process using a measuring instrument called fitness. The fitness value of a chromosome describes the quality of the chromosomes in that population. This process will evaluate each population by calculating the fitness value of each chromosome and evaluating it until the quitting criteria are met. If the quitting criteria have not been met, a new generation will be formed again by repeating step 2. Some of the criteria for stopping are often used, including: stopping in a certain generation, stopping after several consecutive generations obtained the highest fitness value has not changed, stopping in n generations does not get a higher fitness value.

The search for the shortest route for Timor Island tourism trips using genetic algorithms can be implemented to overcome problems, because genetic algorithms are one of the soft computing models that are often used in solving optimization problems, using the concept of biological evolution, an output is produced in the form of a combination of travel route routes [19]. Implementation of genetic algorithms on the search for the shortest route by means of creating software. In the software created, several combinations of tourist destination locations are inputted, then software testing is carried out with varied genetic algorithm parameters and the best chromosomes will be known. In the creation of this system, a genetic algorithm method was used to optimize routes whose database was obtained from the Google Maps API [20]. Google Maps is a web-based mapping service provided by Google for free. Google also provides the Google Maps API service which is a technological development of google that is used to embed Google Map in an application that is not created by Google [21]. Google Maps API is a library in the form of javascript that has a server-side nature, namely maps stored on Google servers can be used by users useful for modifying maps in Google Maps as needed [22]. Therefore, it is necessary to "Search for the Shortest Route of Tourist Locations on the Island of Timor Using a Javascript-Based Genetic Algorithm" which is expected to make it easier for tourists to find optimal tourist routes. This study aims to design and build a system that has information about the destinations of Timor Island tourist attractions and has the function of optimizing tourist travel routes using the Genetic Algorithm method.

2. RESEARCH METHODS

Basically, a study aims to find, develop, or study a knowledge. Finding can be interpreted as an attempt to get something, in an attempt to fill in the gaps or shortcomings. Research methods are stages that researchers go through starting from the formulation of the problem to the conclusion, namely forming a systematic flow. This research method is used as a research guideline in the implementation of research so that the results achieved do not deviate from predetermined goals. At this stage, a review of the system to be studied is carried out to observe and research more deeply and analyze the problems that exist in the running system. The methodology used in this study consisted of the following steps:

1. Conduct literature studies on various references related to the research conducted.
2. Identifying the needs of the system to built includes conducting field surveys to the locations of tourist attractions in the Timor Island area so that they can sight the condition of the tourist attractions.
3. Designing a tourist route determination system.
The system design stage aims to obtain an initial picture in the form of steps to create an application system for finding the shortest route for Timor Island attractions.
4. Implement a system of determining the shortest tourist route based on the design of the system.

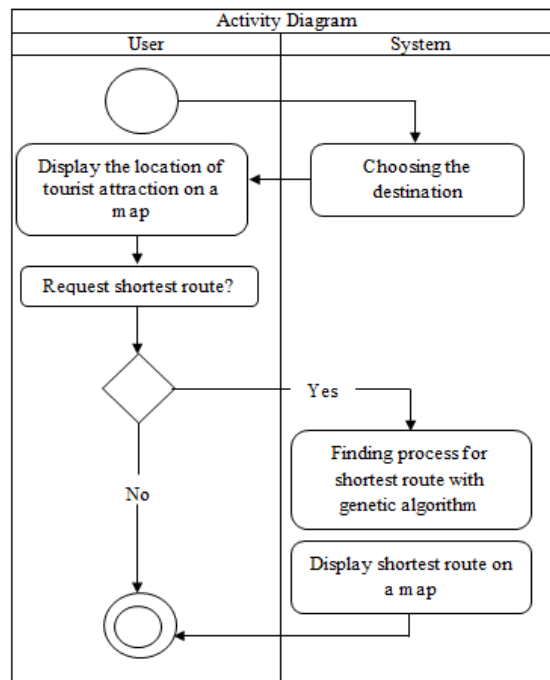


Figure 1. Activity diagram

Figure 1 presents the activity of the diagram. First, the user opens the application. Next, the user fills out the form of the place to be visited. The user selects and displays the location of the destination to be visited. Next, the user presses the button to find the shortest route to be processed using a genetic algorithm. If the results of the data request and the algorithm process are successful, it will obtain the shortest route and be displayed on the monitor screen. Otherwise, it will bring up an error message.

5. Testing the system

Testing the system aims to find errors on system and find out the suitability of the system created with the needs of the user.

System Design

System inputs, processes, and outputs

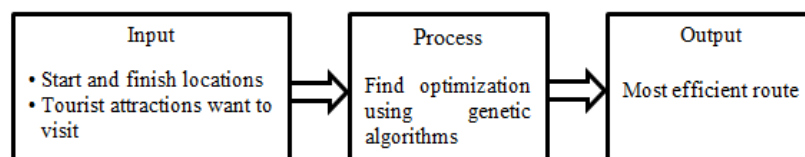


Figure 2. System inputs, processes, and outputs

Figure 2 presents the process flow containing the System Inputs, Processes, and Outputs. Inputs are carried out by the user in the form of initial location selection, destination location, population number, crossover ratio, and mutation ratio. Input is entered into the process by pressing the Find Route button. After the button is pressed, the system performs a process based on genetic algorithms to obtain optimal results. Then, the system produces an output in the form of the shortest route of the tourist attraction to be visited. In this route optimization system, users are expected to enter data on the location of the starting place and the objects they want to visit, then processed by looking for the most efficient route optimization with the Genetic Algorithm. The structure of the genetic algorithm is generally depicted in the following cycle of **Figure 3**.

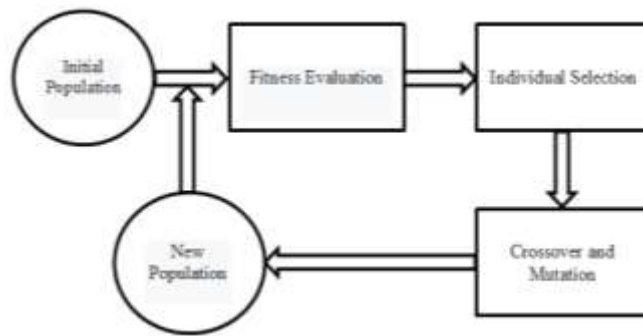


Figure 3. Genetic algorithm cycle

Figure 3. A genetic algorithm is an algorithm that applies an understanding of natural evolution to problem-solving efforts. The approach taken by this algorithm is to randomly combine the various best solutions selected in a group to get the next generation of the best solution, namely in a condition that maximizes compatibility which will be called fitness. The chosen generation is the generation that represents improvements in the previous population or group. By performing a repetitive process, this algorithm can get the most appropriate solution to the faced problem.

Once processed, the user will receive output in the form of the most efficient route for the selected Attractions. The flowchart of the established tourist travel route determination system is as follows.

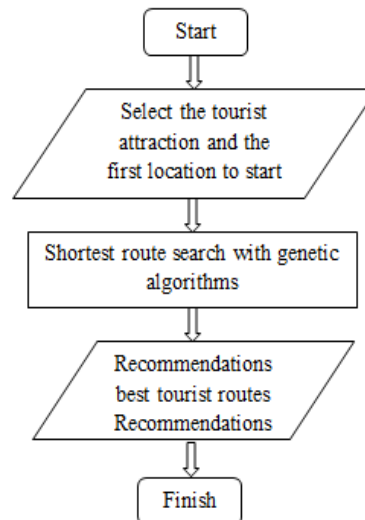


Figure 4. Flowchart of tourist travel route

Figure 4 presents a flowchart of determining the tourist travel route of the system to be created. System creation begins with creating a form for inputting data by the user. Then proceed with connecting with the google map API with the aim of obtaining data on the location and distance of tourist attractions. Furthermore, typing the genetic algorithm script using Javascript by applying the genetic algorithm cycle. Then, test the system that has been designed whether it runs according to the theory of genetic algorithms to obtain optimal results or not. If the test results are not successful, then checking the errors experienced by the system is carried out for repairs until they get the expected results.

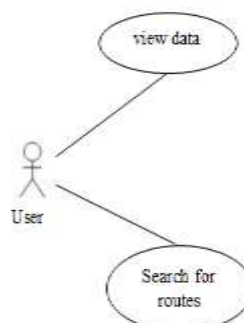


Figure 5. Use case diagram

Figure 5 presents a use case diagram used to understand and know what functions exist in a system, as well as who can use these functions. Use case diagram is a functional modeling to describe the behavior of a system to be built. The figure above shows that the system built has only one role, namely as a user.

3. RESULTS AND DISCUSSION

Implementation of Genetic Algorithms

1) Population initialization function

The population initialization function is used to randomly generate the initial population [23]. The number of genes per individual depends on the number of destination locations chosen by the user. Genes in individuals are symbolized by alphabetic characters.

```

this.initialize = function (chromosomeLength) {
  this.individuals = [];
  for (var i = 0; i < ga.populationSize; i++) {
    var newIndividual = new ga.individual(chromosomeLength);
    newIndividual.initialize();
    this.individuals.push(newIndividual);
  }
}

```

2) Evaluation of Individual Fitness Values

The fitness value calculation function is used to express the value of the destination function [24]. The higher the fitness value, the more optimal the solution produced from the individual.

```

this.getFittestIndex = function () {
  var fittestIndex = 0;
  for (var i = 1; i < this.individuals.length; i++) {
    if (
      this.individuals[i].calcFitness() >
      this.individuals[fittestIndex].calcFitness()
    ) {
      fittestIndex = i;
    }
  }
  return fittestIndex;
};
this.getFittest = function () {
  return this.individuals[this.getFittestIndex()];
};

```

3) Individual Selection With Roulette Wheel

The Roulette Wheel selection process starts by looking for random values from up to 1, then selected based on the probability generated by each individual. Selected individuals will be processed in the crossover process to produce a new generation [25].

```

this.rouletteWheelSelection = function () {
  var totalFitness = 0;
  var fitness = [];
  for (var i = 0; i < this.individuals.length; i++) {
    fitness[i] = this.individuals[i].calcFitness();
    totalFitness = totalFitness + fitness[i];
  }
  var P = [];
  var C = [];
}

```

```

    for (let i = 0; i < this.individuals.length; i++) {
      P[i] = fitness[i] / totalFitness;
      if (i == 0) {
        C[i] = P[i];
      } else if (i == this.individuals.length) {
        C[i] = 1;
      } else {
        C[i] = C[i - 1] + P[i];
      }
    }
    var R = [];
    var rouletteWheelPopulation = new ga.population();
    for (let i = 0; i < this.individuals.length; i++) {
      R[i] = Math.random();
      for (let j = 0; j < C.length; j++) {
        if (R[i] <= C[j]) {
          rouletteWheelPopulation.addIndividual(this.individuals[j]
        );
          break;
        }
      }
    }
    return rouletteWheelPopulation.getFittest();
  }
}

```

4) Crossover Process

This crossover process involves 2 individuals to produce 2 new individuals [26]. The crossover process used is to exchange 2 genes at random positions between 2 individuals who are parents.

```

this.crossover = function (individual, offspringPopulation) {
  var offspringChromosome = [];
  var startPos = Math.floor(this.chromosome.length * Math.random());
  var endPos = Math.floor(this.chromosome.length * Math.random());

  var i = startPos;
  while (i != endPos) {
    offspringChromosome[i] = individual.chromosome[i];
    i++;

    if (i >= this.chromosome.length) {
      i = 0;
    }
  }
  for (parentIndex in individual.chromosome) {
    var node = individual.chromosome[parentIndex];

    var nodeFound = false;
    for (offspringIndex in offspringChromosome) {
      if (offspringChromosome[offspringIndex] == node) {
        nodeFound = true;
        break;
      }
    }
    if (nodeFound == false) {
      for (
        var offspringIndex = 0;
        offspringIndex < individual.chromosome.length;

```



```

        offspringIndex++
    ) {
        if (offspringChromosome[offspringIndex] == undefined) {
            offspringChromosome[offspringIndex] = node;
            break;
        }
    }
}
}
var offspring = new ga.individual(this.chromosomeLength);
offspring.setChromosome(offspringChromosome);
offspringPopulation.addIndividual(offspring);
};

```

5) Mutation Process

The mutation process is carried out on all genes that are imparted in individuals in a generation [27]. The process begins by randomly determining 2 genes positions to be exchanged in one individual.

```

this.mutate = function () {
    this.fitness = null;
    for (index in this.chromosome) {
        if (ga.mutationRate > Math.random()) {
            var randomIndex = Math.floor(Math.random() *
this.chromosomeLength);
            var tempNode = this.chromosome[randomIndex];
            this.chromosome[randomIndex] = this.chromosome[index];
            this.chromosome[index] = tempNode;
        }
    }
};

```

6) Function of displaying maps with Google Maps API

This function displays a map from Google Maps with tourist destination points and start locations selected by the user [28].

```

function initializeMap() {
    map = new google.maps.Map(document.getElementById("map-canvas"), {
        center: { lat: -9.199279567621584, lng: 124.4832042037652 },
        zoom: 9,
        mapTypeId: "hybrid",
        streetViewControl: false,
        mapTypeControl: false,
    });
}

```

The designed system implemented according to the design that has been made. Users of the application that is built prospective tourists who will travel to Timor Island attractions. Applications built based on websites can be visited on the page <http://localhost/tspwisata/>. The pages of the website are shown in the following figure.



Figure 6. Main Page

Figure 6 presents the main page view. On the main page, the user selects the attraction to visit and determines its initial location. Then fill in the Default Location column as the starting point of the location and fill in the Add Location column according to the location you want to visit. If it is appropriate then the user selects the Find Route button. The confirmation page can be seen in **Figure 7**.

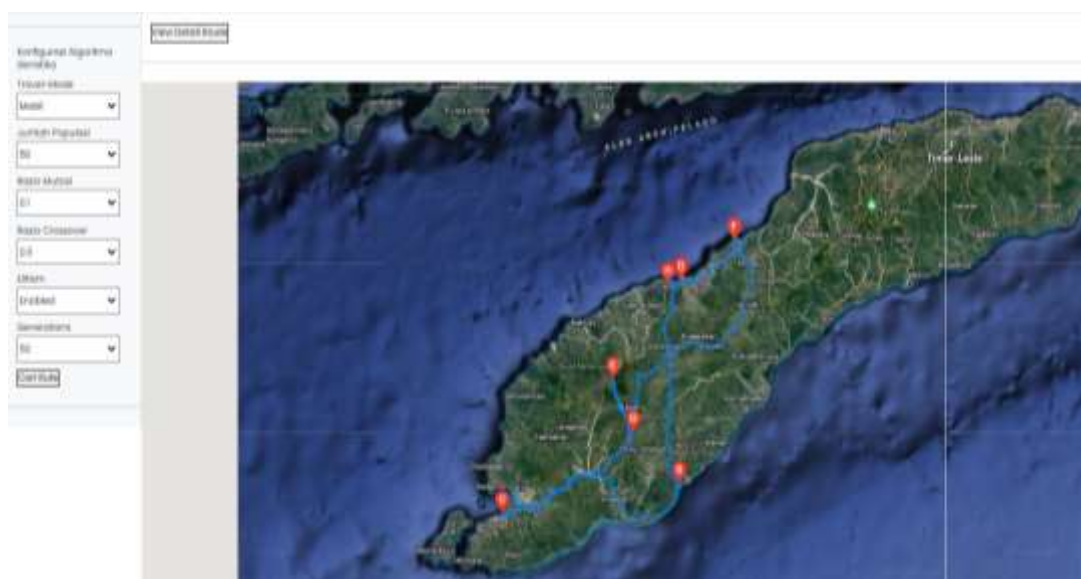


Figure 7. Confirmation Page

Figure 7 presents the appearance of the confirmation page. On the confirmation page, the system displays the route and location that has been inputted by the user. Information on the shortest route with details of the distance traveled can be seen in the following figure.



Figure 8. Shortest route

Figure 8 presents the results of the genetic algorithm process carried out by the system and obtaining optimum results. From the picture above, you can see in detail the pairing between the two locations along with the distance between the two locations. The route from the place of origin to all the destinations exactly once then back to the place of origin is the summation of all the distances from the pair between locations.

Application Output Test Results

- 1) Attraction Location Selection
- 2) This page contains a selection of attractions on Timor Island and a map of the Timor Island to determine the start location. On this page the user input the data of the destination attraction and the start location. The location selection of attractions can be seen in the following figure.

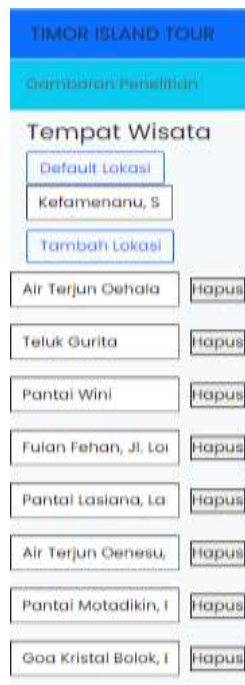


Figure 9. Attraction location selection

Figure 9 presents the data inputted by the user can be captured by google map and the user can choose the location in question.

3) Confirmation of Selected Attraction Location

This page contains information from inputs that have been selected by the user before. If the user wants changes can select "hapus" and then change the desired destination. After that, the user can click the button "cari rute" as seen in the following figure.

Konfigurasi Algoritma Genetika

Travel-Mode

Jumlah Populasi

Rasio Mutasi

Rasio Crossover

Elitism

Generations

Figure 10. Shortest route search

Figure 10 shows that the user inputs which will be the parameters used in the genetic algorithm process.

4) Recommended routes to attractions

This page contains a map of recommended tourist travel routes with detailed distances from each selected place.



Figure 11. Recommended routes to tourist sites

Number of Locations: 9

Generations: 50

Total Estimated Distance: 771.89 km

Figure 11 shows the results of the route display on the google map in the form of location markers and route recommendations to tourist locations.

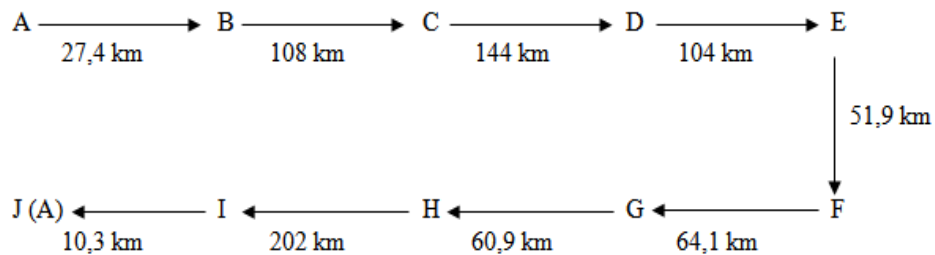


Figure 12. Recommended tourist travel routes

Based on **Figure 12** shows that from the inputted location a shortest route is obtained with the city symbol and the travel flow. The shortest route obtained is from Goa Kristal Bolok, Kupang Regency (A) to Lasiana Beach, Kupang City (B), head to Oehala Waterfall, South Central Timor Regency (C), head to Motadikin Beach, Malaka Regency (D), head to Teluk Gurita, Belu Regency (E), head to Padang Fulan Fehan, Belu Regency (F), head to Wini Manamas Beach, North Central Timor Regency (G), head to Sasi Kefamenanu Church, North Central Timor Regency (H), head to Oenesu Waterfall, Kupang Regency (I), and finally back to Goa Kristal Bolok, Kupang Regency (J).

4. CONCLUSIONS

Based on research stages have been carried out, it can be concluded that the travel route search application has been built can provide convenience in choosing tourist travel routes on Timor Island. Route optimization applications can help the process of creating routes from the initial location of departure and the attractions to be visited. This system is also used to assist domestic tourists in getting to know the attractions of Timor Island for solutions in the search for the shortest route.

ACKNOWLEDGMENT

Thank you to the Chancellor of Universitas Timor, Kefamenanu who allowed us to participate in research activity. Researchers would like to thank the Chairman of the UPT LPPM Universitas Timor for providing material support with SK No.031/UN60.6/PP/2022.

REFERENCES

- [1] I. P. G. H. Suputra dan L. A. A. R. Putri, "Sistem Rekomendasi Perjalanan Wisata Berbasis Expert System Dan Algoritma Genetika," *J. Resist. (Rekayasa Sist. Komputer)*, vol. 5, no. 1, hal. 47–54, 2022, doi: 10.31598/jurnalresistor.v5i1.761.
- [2] T. S. Putra dan A. Sahari, "Optimasi Rute Distribusi Koran Radar Sulteng Palu Menggunakan Algoritma Genetika," *J. Ilm. Mat. dan Terap.*, vol. 19, no. 1, hal. 39–48, 2022, doi: 10.22487/2540766X.2022.v19.i1.15807.
- [3] J. E. Simarmata, "Penerapan Algoritma Branch And Bound Pada Persoalan Pedagang Keliling (Travelling Salesman Problem)," *RANGE J. Pendidik. Mat.*, vol. 1, no. 2, hal. 111–121, 2020, doi: 10.32938/jpm.v1i2.366.
- [4] M. Yunus dan R. M. T. Rumlaklak, "Optimasi Waktu Eksekusi Penentuan Rute Menuju Obyek Wisata Di Malang Raya Dengan Algoritma Genetika," *J. Teknol. Inf. Teor. Konsep dan Implementasi*, vol. 9, no. 1, hal. 29–40, 2018.
- [5] A. W. Widodo dan W. F. Mahmudy, "Penerapan Algoritma Genetika Pada Sistem Rekomendasi Wisata Kuliner," *J. Ilm. Kursor*, vol. 5, no. 4, hal. 205–211, 2010.
- [6] F. Mone dan J. E. Simarmata, "Aplikasi Algoritma Genetika Dalam Penjadwalan Mata Kuliah," *Barekeng J. ILMU Mat. DAN Terap.*, vol. 15, no. 4, hal. 615–628, 2021, doi: 10.30598/barekengvol15iss4pp615-628.
- [7] A. P. Putra dan S. Yunita, "Sistem Informasi Penentuan Rute Pengiriman Barang di CV ASA Menggunakan Metode Algoritma Genetika," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 2, no. 1, hal. 35–42, 2021.
- [8] A. Rifa'i, "Prediksi Inflasi Indonesia Berdasarkan Fuzzy Ann Menggunakan Algoritma Genetika," *J. ELTIKOM J. Tek. Elektro, Teknol. Inf. Dan Komput.*, vol. 5, no. 1, hal. 12–24, 2022, doi: 10.31961/eltikom.v5i1.215.
- [9] A. Hannawati, Thiang, dan Eleazar, "Pencarian Rute Optimum Menggunakan Algoritma Genetika," *J. Tek. Elektro*, vol. 2, no. 2, hal. 78–83, 2002.
- [10] W. Zhao, Y. Wang, Z. Zhang, dan H. Wang, "Multicriteria Ship Route Planning Method Based on Improved Particle Swarm

- Optimization–Genetic Algorithm,” *J. Mar. Sci. Eng.*, vol. 9, no. 4, hal. 357, 2021, doi: 10.3390/jmse9040357.
- [11] Z. Xue, “Routing Optimization of Sensor Nodes in the Internet of Things Based on Genetic Algorithm,” *IEEE Sens. J.*, vol. 21, no. 22, hal. 25142–25150, 2021, doi: 10.1109/JSEN.2021.3068726.
- [12] C. Madapatha, B. Makki, A. Muhammad, dan E. Dahlman, “On Topology Optimization and Routing in Integrated Access and Backhaul Networks: A Genetic Algorithm-Based Approach,” *IEEE Open J. Commun. Soc.*, vol. 2, hal. 2273–2291, 2021, doi: 10.1109/OJCOMS.2021.3114669.
- [13] R. R. Pratama, “Penyelesaian Travelling Salesman Problem Menggunakan Algoritma Genetika,” *J. Ris. Sist. Inf. Dan Teknol. Inf.*, vol. 2, no. 1, hal. 10–18, 2020, doi: 10.52005/jursistekni.v2i1.35.
- [14] O. Kramer, “Genetic algorithms,” *Genet. algorithm essentials*, vol. Springer, hal. 11–19, 2017.
- [15] L. Xin, P. Xu, dan G. Manyi, “Logistics Distribution Route Optimization Based on Genetic Algorithm,” *Comput. Intell. Neurosci.*, hal. 1–9, 2022, doi: 10.1155/2022/8468438.
- [16] Aswandi, S. Cokrowibowo, dan A. Irianti, “Model Penentuan Rute Terpendek Penjemputan Sampah Menggunakan Metode MTSP dan Algoritma Genetika,” *J. Appl. Comput. Sci. Technol.*, vol. 2, no. 1, hal. 43–48, 2021, doi: 10.52158/jacost.v2i1.168.
- [17] N. D. Rumlaklak dan E. S. Pandie, “Penggunaan Algoritma Genetika Dalam Penentuan Rute Wisata di Kota/Kabupaten Kupang,” in *Seminar Nasional & Konferensi Ilmiah Sistem Informasi, Informatika & Komunikasi*, 2018, hal. 660–667.
- [18] A. Yandra, *Algoritma genetika: teori dan aplikasinya untuk bisnis dan industri*. 2012.
- [19] S. Sivanandam dan S. Deepa, “Genetic Algorithms. In: Introduction to Genetic Algorithms,” vol. 2, no. 1, Springer, Berlin, Heidelberg, 2008, hal. 60–66.
- [20] S. L. Rayendra, R. W. Perdana, dan N. M. Yendrizal, *Kecerdasan Buatan*. 2022.
- [21] A. Triansah, D. Cahyadi, dan I. F. Astuti, “Membangun Aplikasi Web dan Mobile Android Untuk Media Pencarian Kost Menggunakan Phonegap Dan Google Maps API,” *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 10, no. 1, hal. 58–61, 2015, doi: 10.30872/jim.v10i1.21.
- [22] A. B. Prakoso, Y. Ariyanto, dan A. R. T. H. Ririd, “Optimasi Rute Lokasi Wisata Kota Malang Menggunakan Metode Algoritma Genetika,” *J. Inform. Polinema*, vol. 3, no. 3, hal. 48–52, 2017, doi: 10.33795/jip.v3i3.34.
- [23] M. Pant, A. Abraham, dan M. Ali, “Unconventional initialization methods for differential evolution,” *Appl. Math. Comput.*, vol. 219, no. 9, hal. 4474–4494, 2013, doi: 10.1016/j.amc.2012.10.053.
- [24] Y. Huang, W. Li, F. Tian, dan X. Meng, “A fitness landscape ruggedness multiobjective differential evolution algorithm with a reinforcement learning strategy,” *Appl. Soft Comput.*, vol. 96, hal. 106693, 2020, doi: 10.1016/j.asoc.2020.106693.
- [25] M. Yunus dan R. M. T. Rumlaklak, “Optimasi Waktu Eksekusi Penentuan Rute Menuju Obyek Wisata Di Malang Raya Dengan Algoritma Genetika,” *J. Teknol. Inf.*, vol. 9, no. 1, hal. 29–40, 2018.
- [26] M. B. Arif, “Penentuan Jarak Terpendek Pada Jalur Pengiriman Musae Chips Dengan Menggunakan Algoritma Genetika,” *J. Teknol. Inf. dan Terap.*, vol. 6, no. 1, hal. 19–23, 2019, doi: 10.25047/jtit.v6i1.97.
- [27] F. Altiparmak, M. Gen, L. Lin, dan T. Paksoy, “A genetic algorithm approach for multi-objective optimization of supply chain networks,” *Comput. Ind. Eng.*, vol. 51, no. 1, hal. 25–30, 2006, doi: 10.1016/j.cie.2006.07.011.
- [28] R. Rosdania, F. Agus, dan A. H. Kridalaksana, “Sistem Informasi Geografi Batas Wilayah Kampus Universitas Mulawarman Menggunakan Google Maps API,” *J. Inform. Mulawarman*, vol. 10, no. 1, hal. 38–46, 2015.

