https://doi.org/10.31871/IJNTR.9.12.4

International Journal of New Technology and Research (IJNTR)
ISSN: 2454-4116, Volume-9, Issue-12, December 2023 Pages 01-07

Implementation of Held-Karp Algorithm to Calculate Shortest Route Path

Yulianto Bayu Prasetyo, Moh. Ali Romli

Abstract- **Shipping goods has become a part of the current economic growth in society. In the digital era and the advancement of e-commerce, the demand for shipping services is increasing. However, the process of selecting an efficient and optimal delivery route remains a complex challenge. Currently, some couriers and delivery service providers still rely on manual methods or inaccurate estimates, resulting in inefficient deliveries and higher costs. Based on previous research, the Held-Karp algorithm produces better results than the Iterative Deepening Search algorithm in finding optimal delivery routes. Therefore, in this study, an Android application was developed using Google Maps services and integrating the Held-Karp algorithm. The Held-Karp algorithm is used to optimize the delivery route, considering the most efficient distance and delivery sequence. The results will be displayed in an interactive map, facilitating couriers in performing their delivery tasks. By using this application, it is expected to improve the efficiency of goods delivery, reduce delivery time, and optimize the utilization of courier resources. The development of this application requires knowledge of the Held-Karp algorithm, Android-based programming, and integration with map services such as the Google Maps API.**

***Index Terms*— Held-Karp Algorithm, Traveling Salesman Problem, Android, Google Maps.**

I. Introduction

The selection of delivery routes is a problem for couriers in the distribution of goods, one of which is determining the order of delivery of goods to the address of each recipient of goods. This problem can be influenced by two main factors, namely travelling time and distance. Couriers must consider the time required and the distance travelled from the distribution centre to the recipient's address. To optimize the distribution process, courier companies often rely on efficient routing methods and the latest technology, which helps them determine the most effective delivery sequence to ensure the goods can arrive on time and in good condition at the recipient's hands. Therefore, a method is needed as one of the solutions to overcome this problem. The problem is commonly called the Traveling Salesman Problem (TSP). TSP is a problem to find the shortest travel route after visiting all location points, the courier can find the address with the shortest travel route after visiting all points optimally [1].

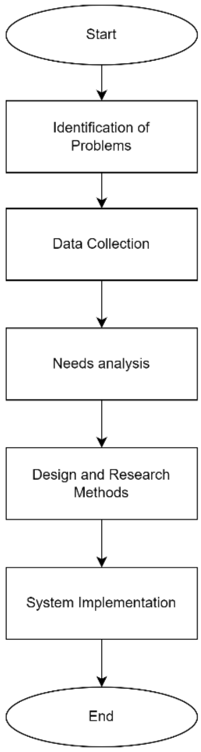
With this problem, a way to solve the problem is needed, one of which can use a suitable algorithm, namely the Held-Karp algorithm. The Held-Karp algorithm also commonly called the Bellman Held-Karp algorithm, is a dynamic programming method proposed in 1962 independently by Bellman and Karp to solve the salesman's journey problem [2]. used as a solution to solve the TSP, which is used as the determination of the shortest route. By implementing this algorithm as a method of solving the problem, couriers can improve the efficiency of distributing goods, reduce travel time, and ensure faster and more timely delivery to customers.

The Held-Karp algorithm will be used in this research to perform the shortest route search and is expected to help couriers to plan their journeys more efficiently. With the optimal solution generated by this algorithm, couriers can avoid long and inefficient routes, save travelling time, and ensure the goods arrive on time in the hands of customers. In addition, the use of the Held-Karp algorithm is also expected to reduce shipping operational costs, thus providing economic benefits to courier companies. Thus, the algorithm is expected to bring about significant improvements in freight management and efficiency.

Yulianto Bayu Prasetyo, University of Technology Yogyakarta. Yogyakarta Indonesia.
Moh. Ali Romli, University of Technology Yogyakarta. Yogyakarta Indonesia.

II. Methodology

This research was conducted using several methods described in Figure 1.



A. *Identification of Problems*

The first step taken in the research is to identify the problem, which will later be used as research material. Identify research objectives and constraints that will be passed in the research. The problem obtained is a problem regarding distance optimization, on the route taken by a courier to each point, where each point is visited only once.

B. Needs Analysis

Needs analysis is carried out to find out the needs needed in research such as, user needs, software needs, and system needs. Analysis of user needs such as features to be created, and who are the users who use the system. Analyze software needs such as the technology used in system development. Analyze system requirements such as the needs for the development process and when the system is run.

C. Design and Research Methods

The next step is to design the system design and select the method used as a solution to solve the problem. From the previous steps, it can be used as a basis for designing system designs, such as interface design, system flow design and so on, after that doing research and comparisons on what methods can be used, and choosing the method that will be used in this study. And the held-karp algorithm was chosen as the method used to calculate route optimization.

D. System Implementation

The system implementation stage or system creation, after analyzing and designing as the basis for making the system, the next step is to apply it to the system to be created, at this stage testing is also carried out on the application, so that it runs properly.

III. PROPOSED SYSTEM

A. System Architecture

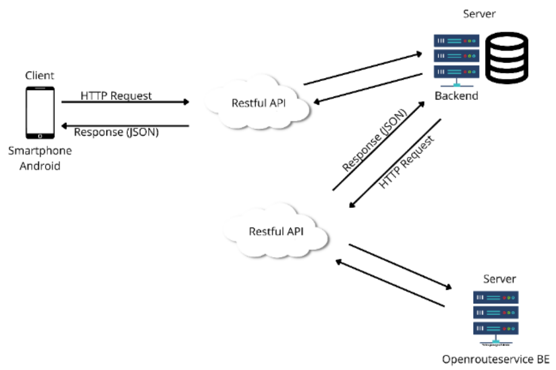


Figure 1. Architecture of System

The system is built to have one interface, namely an android application, which includes features that will be used by users in operating the application. Such as data input and displaying the information needed. This application applies the RESTful API architecture because its application is very easy and suitable for client-server applications. There is also an architecture that was created first, namely SOAP with functions similar to REST, in previous research it was stated that REST has better performance compared to SOAP for testing requests and responses for web services [3]. Another advantage of using RESTful API is that it can be applied to almost all programming languages, so that it can process data in the database without the need to think about the platform or programming language used. JSON (JavaScript Object Notation) is a type of response format provided, JSON data is supported by many popular programming languages at this time, besides JSON there is also another form of format, namely XML, but in previous research on testing and comparison between JSON and XML it was concluded that JSON is faster in the data serialization process and data parsing process than XML and has a smaller data size than XML [4].

B. Optimization Problem

In general, solving the shortest route search problem can be done using two methods, namely the conventional algorithm method and the heuristic method. The conventional algorithm method is applied by means of mathematical calculations as usual, while the heuristic method is applied with artificial intelligence calculations, by determining the knowledge base and calculations [5].

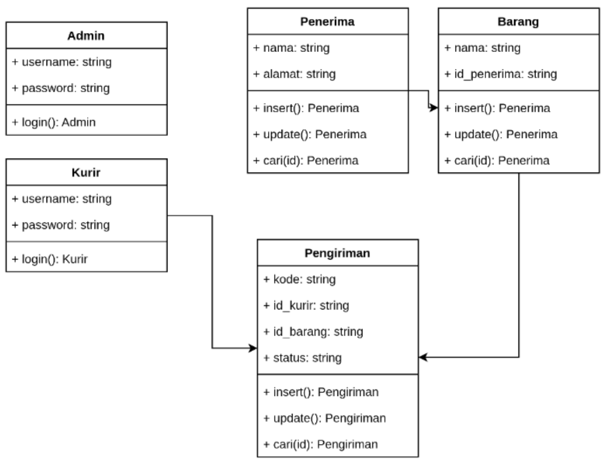
C. Held-Karp Algorithm

The Traveling Salesman Problem (TSP) is a well-known problem in computing minimum-cost Hamiltonian cycles on a given weighted graph. Since this problem is NP-hard, many techniques have been developed not only to solve it optimally, but also to compute upper and lower bounds on its optimal value. Theoretically, TSP can be represented as a complete and weighted Graph G = (V, E), where V is the set of vertices and E is the set of edges [6]. The lower bound procedure proposed by Held and Karp is based on Lagrangean relaxation: at each iteration, the edge weights of the base graph G = (V, E) are modified so that the Hamiltonian cycle cost remains unaffected. Then, valid lower bounds for the TSP can be efficiently obtained from minimum spanning 1-trees, i.e., spanning trees with one additional edge, relying on the observation that Hamiltonian cycles are a special case of spanning 1-trees. [7]. TSP can be classified into Symmetric Traveling Salesman Problem (STSP), Asymmetric Traveling Salesman Problem (ATSP), and Multi Traveling Salesman Problem (MTSP) [8]. The Held-Karp algorithm is generally used to solve the problem of finding the closest path, for example in the Traveling Salesman Problem (TSP) or on tourist trips. The Held-Karp algorithm has a complexity of O(2nn2) [9].Based on experiments conducted in previous research, it is concluded that the Held-Karp method produces better routes compared to the Iterative Deepening Search method [10].

D. Class Diagram

Class diagram is one of the parts of UML. This diagram is used to describe classes, attributes, and methods. This diagram makes it easier to describe the system.

https://doi.org/10.31871/IJNTR.9.12.4
International Journal of New Technology and Research (IJNTR)
ISSN: 2454-4116, Volume-9, Issue-12, December 2023 Pages 01-07



IV. Implementation

After conducting research on the system creation process, the next step is to implement the held-karp algorithm to calculate the shortest route distance optimization, which is then applied to the system.

A. Held-Karp Algorithm

In calculations using the held-karp algorithm, the first thing is to compare the distance, time, or cost of each point, according to the optimization requirements to be calculated. In this research we will calculate optimization at distance, for example as in the table below.

Table 1. Distance Table

No	Start Location	DestinationLocation	Distance
1	A	B	2000 M
2	A	C	9000 M
3	A	D	10000 M
4	B	A	1000 M
5	B	C	6000 M
6	B	D	4000 M
7	C	A	15000 M
8	C	B	7000 M
9	C	D	8000 M
10	D	A	6000 M
11	D	B	3000 M
12	D	D	12000 M

After getting the comparison of each distance, the next step is to model the distance data into a distance comparison matrix. To carry out calculations, a matrix can be created as below:

$$C = \begin{pmatrix} 0 & 2000 & 9000 & 10000 \\ 1000 & 0 & 6000 & 4000 \\ 15000 & 7000 & 0 & 8000 \\ 6000 & 3000 & 12000 & 0 \end{pmatrix}$$

Functions description:

- g(x, S) - starting from 1, path min cost ends at vertex x, passing vertices in set S exactly once
- c_{xy} - edge cost ends at x from y
- p(x, S) - the second-to-last vertex to x from set S. Used for constructing the TSP path back at the end.

k = 0, null set:

Set Ø:
g(2, Ø) = c21 = 1000
g(3, Ø) = c31 = 15000
g(4, Ø) = c41 = 6000

k = 1, consider sets of 1 element:

Set {2}:
g(3,{2})
= c32 + g(2, Ø)
= c32 + c21
= 7000 + 1000
= 8000 p(3,{2}) = 2
g(4,{2})
= c42 + g(2, Ø)
= c42 + c21
= 3000 + 1000
= 4000 p(4,{2}) = 2

Set {3}:
g(2,{3})
= c23 + g(3, Ø)
= c23 + c31
= 6000 + 15000
= 21000 p(2,{3}) = 3
g(4,{3})
= c43 + g(3, Ø)
= c43 + c31
= 12000 + 15000
= 27000 p(4,{3}) = 3

Set {4}:
g(2,{4})
= c24 + g(4, Ø)
= c24 + c41
= 4000 + 6000
= 10000 p(2,{4}) = 4
g(3,{4})
= c34 + g(4, Ø)
= c34 + c41
= 8000 + 6000
= 14000 p(3,{4}) = 4

k = 2, consider sets of 2 elements:

Set {2,3}: g(4,{2,3})
= min {c42 + g(2,{3}), c43 + g(3,{2})}
= min {(3000+21000), (12000+8000)}
= min {24, 20}
= 20000 p(4,{2,3}) = 3

Set {2,4}: g(3,{2,4})
= min {c32 + g(2,{4}), c34 + g(4,{2})}

= min {(7000+10000), (8+4)}
= min {17000, 12000}
= 12000   p(3,{2,4}) = 4

Set {3,4}: g(2,{3,4})
= min {c23 + g(3,{4}), c24 + g(4,{3})}
= min {(6000+14000), (4000+27000)}
= min {20000, 31000}
= 20000   p(2,{3,4}) = 3

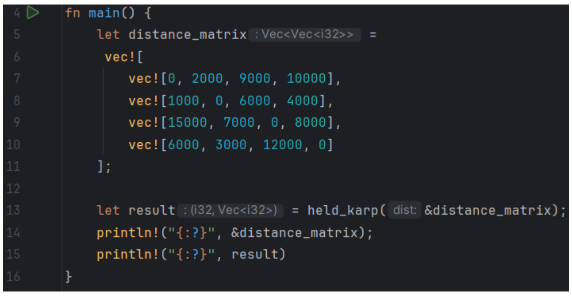
Length of an optimal tour:

f = g(1,{2,3,4})
= min {c12 + g(2,{3,4}), c13 + g(3,{2,4}), c14 + g(4,{2,3})}
= min {(2000 + 20000), (9000 + 12000), (10000 + 20000)}
= min {22000, 21000, 30000}
= 21000

Predecessor of node 1: p(1,{2,3,4}) = 3
Predecessor of node 3: p(3, {2,4}) = 4
Predecessor of node 4: p(4, {2}) = 2
Predecessor of node 2: p(2, {}) = 1

Optimal TSP tour is given by 1 → 2→ 4 → 3→ 1.[11]

The following is an example of the calculation results of the program pieces used in the system, with the same matrix as the manual calculation that has been done.

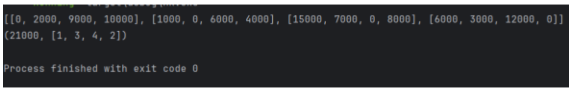


```
fn main() {
    let distance_matrix :Vec<Vec<i32>> =
     vec![
        vec![0, 2000, 9000, 10000],
        vec![1000, 0, 6000, 4000],
        vec![15000, 7000, 0, 8000],
        vec![6000, 3000, 12000, 0]
    ];

    let result :(i32, Vec<i32>) = held_karp( dist: &distance_matrix);
    println!("{:?}", &distance_matrix);
    println!("{:?}", result)
}
```

Figure 2. Example Simple Held-Karp Program

And the result is the same as the manual calculation, which is in the figure below



```
[[0, 2000, 9000, 10000], [1000, 0, 6000, 4000], [15000, 7000, 0, 8000], [6000, 3000, 12000, 0]]
(21000, [1, 3, 4, 2])

Process finished with exit code 0
```

Figure 3. Result Program

Where 21000M is the cost or total distance traveled and[1, 3, 4, 2] is the optimal route.

B. System Implementation Results

The following is a page view that is divided into several parts, which are important in the Application for selecting goods delivery routes, including:

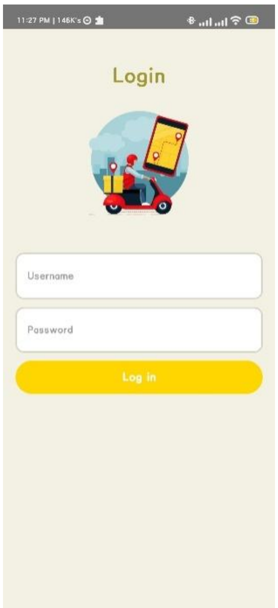


Figure 4. Login Page

The login page serves to distinguish access rights between couriers and admins. After logging in, it will enter the menu page according to its role or access rights.

Figure 5. Menu

After an account with a courier role will be directed to the dashboard menu page, which only has two options, namely delivery and a button to exit, made simple which aims to be easy to use.

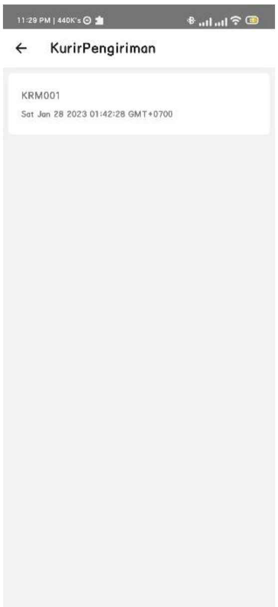


Figure 6. List

The delivery list page contains deliveries that will be assigned to couriers, which displays a list of completed or unfinished deliveries.

Figure 7. Detail

The delivery details page contains delivery details such as recipient information and also information about the sequence of routes traveled.

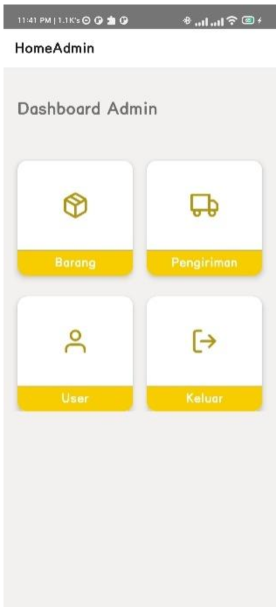


Figure 8. Menu Admin

The admin menu page is a page that will appear if the user who has successfully logged in has the role of admin. This page contains menus that can be done by the admin, such as inputting goods, shipping, and couriers or application users.

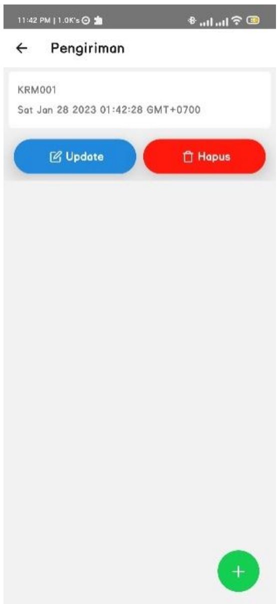


Figure 9. List For Admin

The admin delivery list page functions to add, display, change and delete shipments.

Figure 10. Detail For Admin

The admin delivery details page serves to display delivery details and calculate the optimal route using the held-karp algorithm.

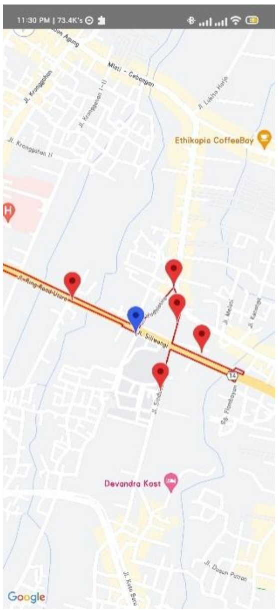


Figure 11. Display Map

The path display page is a map display using Google Maps whose function is to display the path in the form of a red line, as a marker of the route traveled, the blue pin is the starting point.

C. Testing

The next stage is testing the running system to find out that the application is running properly. Black Box Testing is a suitable method used at this stage. Black box testing is an integral part of correctness testing but its ideas are not limited to correctness testing only. Black box testing iscomplementary to white box testing technique and likely to uncover different class of errors then white box method. Black box testing is conducted at various stages of both the software development life cycle and the software testing life cycle, including regression testing, acceptance testing, unit testing, integration testing, and system testing. This testing approach exclusively emphasizes the assessment of the functionality of software applications.[12].

Menu Feature	Expected Outcome	Conclusion
Login	Display and execute login action	Success
Home	displays the menu for app navigation	Success
Detail	Display details in the form of calculation results or data from the database	Success
Calculate Shortest Route	Execute background process for Calculating the sh	Success
Show Route	Display route with google maps	Success
Error Handling	displays an error if something	Success

V. CONCLUSION

From the research that has been done on the system developed, namely the shortest route calculation application using the held-karp algorithm, it can be concluded that the application can perform calculations as it should and is expected, the application can help couriers in calculating routes with the shortest distance so that the delivery process can be more efficient than before.

REFERENCES

[1] A. P. C, A. P. C, B. Pramono, and L. M. B. Aksara, “TRAVELLING SALESMAN PROBLEM (TSP) UNTUK MENENTUKAN RUTE TERPENDEK BAGI KURIR KOTA KENDARI MENGGUNAKAN ALGORITMA GREEDY BERBASIS ANDROID,” *semanTIK : Teknik Informasi*, vol. 3, no. 1, Aug. 2017, doi: 10.55679/semantik.v3i1.2617.

[2] I. Gede, S. Rahayuda, N. Putu, L. Santiari, and N. Y. Arso, “Penerapan Bidirectional Search dan Held-Karp pada Penentuan Rute PengirimanProduk,” *JurnalTeknologiInformasi dan IlmuKomputer*, vol. 5, no. 5, pp. 549–558, Oct. 2018, doi: 10.25126/JTIIK.201855881.

[3] M. G. L. Putra and M. I. A. Putera, “ANALISIS PERBANDINGAN METODE SOAP DAN REST YANG DIGUNAKAN PADA FRAMEWORK FLASK UNTUK MEMBANGUN WEB SERVICE,” *Scan :JurnalTeknologiInformasi dan Komunikasi*, vol. 14, no. 2, pp. 1–7, Jun. 2019, doi: 10.33005/SCAN.V14I2.1480.

[4] S. Saryanto, S. Sumarsono, and N. D. Retnowati, “COMPARATIVE ANALYSIS OF XML AND JSON USING PHP APPLICATION PLATFORM WITH REPRESENTATIONAL STATE TRANSFER (REST) ARCHITECTURAL,” *Compiler*, vol. 2, no. 2, 2013, Accessed: Nov. 20, 2023. [Online]. Available: https://ejournals.itda.ac.id/index.php/compiler/article/view/53

[5] A. Program Studi Sistem Informasi, “Pemanfaatan Metode Heuristik Dalam Pencarian Jalur TerpendekDenganAlgoritmaSemut dan AlgoritmaGenetika,” *Inspiration: JurnalTeknologiInformasi dan Komunikasi*, vol. 3, no. 1, Jun. 2013, doi: 10.35585/INSPIR.V3I1.27.

[6] * Rizki, P. Sinaga, R. P. Sinaga, and U. N. Medan, “PerbandinganAlgoritma Cheapest Insertion Heuristic Dan Nearest Neighbor Dalam Menyelesaikan Traveling Salesman Problem,” *Jurnal Riset RumpunMatematika dan IlmuPengetahuanAlam (JURRIMIPA)*, vol. 2, no. 2, pp. 238–247, Jul. 2023, doi: 10.55606/JURRIMIPA.V2I2.1614.

[7] G. Righini, “Efficient optimization of the Held–Karp lower bound,” *Open Journal of Mathematical Optimization*, vol. 2, no. 9, pp. 1–17, 2021, doi: 10.5802/OJMO.11/.

[8] A. Rao *et al.*, “Literature survey on travelling salesman problem using genetic algorithms,” *International Journal of Advanced Research in Eduation Technology (IJARET)*, vol. 2, no. 1, p. 42, 2015.

[9] P. I. PemrogramanDinamis Pada ManajemenPengirimanProdukMenggunakan Metode Held Karp Gede Surya Rahayuda, N. Putu Linda Santiari, and S. STIKOM Bali Jl Raya Puputan, “PenerapanPemrogramanDinamis Pada ManajemenPengirimanProdukMenggunakan Metode Held Karp,” *E-Proceedings KNS&I STIKOM Bali*, pp. 513–518, Aug. 2017, Accessed: Nov. 16, 2023. [Online]. Available: https://www.knsi.stikom-bali.ac.id/index.php/eproceedings/article/view/94

[10] I. G. S. Rahayuda and N. P. L. Santiari, “Implementasi dan Perbandingan Metode Iterative Deepening Search dan Held-Karp pada ManajemenPengirimanProduk,” *Sisfo*, vol. 07, no. 02, Jan. 2018, doi: 10.24089/J.SISFO.2018.01.005.

[11] “Held–Karp algorithm - Wikipedia.” Accessed: Nov. 16, 2023. [Online]. Available: https://en.wikipedia.org/wiki/Held%E2%80%93Karp_algorithm

[12] Mohd. E. Khan, “Different Approaches to Black Box Testing Technique for Finding Errors.” Jul. 21, 2021. Accessed: Dec. 15, 2023. [Online]. Available: https://papers.ssrn.com/abstract=3890672