Intelligent decision support systems for determining tour bus route with time windows: A metaheuristic approach

H Hudzaifah¹, A F Rizana², F Ramadhan¹*, and A Imran¹

¹ Department of Industrial Engineering, Institut Teknologi Nasional, Jl. PHH. Mustofa No. 23 Bandung, Indonesia
² Department of Industrial Engineering, Telkom University, Jl. Telekomunikasi No. 1 Bandung, Indonesia

*f.ramadhan@itenas.ac.id

Abstract. The transportation routing problem is a very popular field in operations research and management science area. Many previous researches are still continuing to develop model and algorithm to produce the best transportation route optimization in various research objects. Technology has enabled the development of the intelligent decision support system (I-DSS) to optimize transportation route. Although technology has evolved, there are still few researches that build I-DSS software to minimize transportation distance and fuel consumption, especially using the metaheuristic approach. Thus, this paper aims to develop I-DSS software to minimize fuel consumption and travel distance by considering time-windows constraint using metaheuristic approach. The software can receive input dynamically to evaluate and select the best transportation route. The results of this software have been tested using real historical data from one of tour and travel company in Indonesia. The company provides travel services using a bus to visit several tourist attractions based on the opening hours of the destinations. I-DSS software output shows that the travel distance is 25.64% more efficient than the current real conditions. In addition, the fuel consumption produced by I-DSS is better than the actual conditions, which is 17.35% more efficient.

1. Introduction

The transportation routing problem is a very popular field in operations research and management science area. Many previous researches are still continuing to develop models and algorithms to produce the best transportation route optimization in various research objects. One of popular topic in transportation case is traveling salesman problem (TSP). TSP is a combinatorial optimization problem that has been implemented in many fields to minimize total distance [1]. TSP variation has been introduced in previous researches such as TSP with time-windows [2], multi-period TSP [3], or recent TSP variation called green TSP [4]. There are still a few researches focused on fuel consumption in green TSP, so many aspects can be developed further on this topic.

Besides the development of transportation model, technology has enabled the development of the intelligent decision support system (I-DSS) to optimize transportation area [5,6]. Although technology has evolved, there are still few researches that build I-DSS software to minimize transportation distance and fuel consumption, especially using the metaheuristic approach. Fast and accurate solution method are needed to be used as an optimization engine on I-DSS. Li et al. [7] do some enhancement in metaheuristic approach called record-to-record travel (RRT) algorithm to solve vehicle routing problem.
RRT produced good results compared to other metaheuristic algorithms with fast computational time [7]. Besides that, RRT already implemented in various transportation optimization cases, such as inventory routing problem [8,9], traveling repairman problem [10], or capacitated vehicle routing problem [11].

Therefore, this paper aims to develop I-DSS software for a modified TSP model using RRT algorithm. This proposed I-DSS will be implemented in real transportation optimization problems. One of the tour and travel bus routes in Indonesia has been chosen as a research object of this paper. Tour and travel bus routes have the same characteristic like TSP, but with more complex constraints, such as multi-period time including time-windows for each tourist attraction. This research object is suitable to test the proposed I-DSS performance based on a modified TSP model.

2. Method

The development of proposed I-DSS tour bus route consists of three main steps. First, this paper developed a mathematical formulation for multi-period TSP with time-windows that similar to research object characteristics. Second, a modified RRT algorithm is developed with several local search and structure modifications. Third, the proposed I-DSS software is developed to determine the optimal transportation route solution that can minimize total distance and fuel consumption.

2.1. Intelligent Decision Support System (I-DSS)

A decision support system (DSS) is an information system based on computer processing that can be used for supporting decision making [5]. The intelligent DSS (I-DSS) is automated processing to determine policy related to decision making processes. I-DSS is designed to help the problem owner or decision maker to find a solution based on decision maker requirements in a user-friendly manner [5]. The rapid progress in DSS has led to extraordinary growth in the use of new theories and technologies such as fuzzy logic, artificial intelligence, or optimization techniques [6]. In transportation routing problems, optimization approach is suitable to determine the best distribution or travel route of vehicle that can minimize total travel costs.

2.2. Multi-period TSP with time-windows

This paper adapts TSP model that was first introduced by Flood [12] (to minimize total distance) with multi-period time addition. Tour and travel bus usually operate on different days, so the transportation route will be different every day. This is suitable to add a multi-period time aspect in the case of transportation problems. All constraints of this paper are similar with Savelsberg [13] that considered time-windows constraint. A tourist attraction as a node destination of bus have the opening hours, so this characteristic is similar with time-windows aspect. The objective function of multi-period TSP is as follows: Let graph \( Z = (V, A) \), where the node set \( V = \{1, 2, ..., n\} \), the arc set \( A = \{(i, j) : i, j \in V\} \), the cost related to arc \((i, j) \in A\) denote \( c_{ij} \geq 0 \), a number of vehicle movements from \( i \) to \( j \) denote \( x_{ij} \), and \( \tau = \{1, 2, ..., T\} \) as a period \( \tau \) where \( t \in \tau \).

\[
\text{minimize } \sum_{t \in \tau} \sum_{i \in V} \sum_{j \notin V} c_{ij} x_{ij} \quad (1)
\]

Besides minimizing the total distance, this paper also aims to find a minimum total fuel consumption based on total distance. Tour and travel bus usually have several types of vehicle that have different fuel consumption. Vehicle type is determined based on number of passengers, so to find a minimum total fuel consumption, the model must add vehicle capacity constraint. The objective function and additional constraints to find a minimum total fuel consumption are as follows:

\[
\text{minimize } \left( \sum_{t \in \tau} \sum_{i \in V} \sum_{j \notin V} c_{ij} x_{ij} \right) \sum_{k \in M} f_k b_k, \quad M = \{1, 2, ..., m\} \quad (2)
\]

\[
p \leq C_k \quad \forall k \in M = \{1, 2, ..., m\} \quad (3)
\]
\[ f_k p, c_k \geq 0, \quad \forall k \in M = \{1, 2, ..., m\} \]  
(4)

\[ b_k \in \{0, 1\}, \quad \forall k \in M = \{1, 2, ..., m\} \]  
(5)

Eq. (2) is a modified objective function that considers fuel consumption for vehicle type \( k \) per unit distance \( f_k \) and binary variable \( b_k \) (equal to 1, if vehicle type \( k \) is used, and 0 otherwise); Eq. (3) is a vehicle type \( k \) capacity limit \( c_k \) based on the number of passengers \( p \); Eq. (4) and (5) are non-negativity and binary variables.

2.3. A modified RRT algorithm

A modified RRT algorithm is another variation of variable neighborhood search (VNS). VNS was first introduced by Mladenovic & Hansen [14] to solve the combinatorial optimization problem. In the last few researches, VNS algorithm has already been implemented in various transportation or supply chain management cases, such as location routing problem [15], inventory routing problem [16], or vehicle routing problem [17]. This approach uses local searches to make systematic changes in the neighborhood. VNS can accept worse solutions on each iteration as long as the solution is feasible. The RRT algorithm is almost similar with VNS, where on each iteration it can accept the worse solution. However, RRT has a tolerance limit (\( \alpha \% \)). The proposed RRT algorithm can be seen in Figure 1.

| The Proposed RRT Algorithm |
|----------------------------|
| 1: procedure RRT |
| 2: step (0) initialization: set a maximum number of uphill iteration \( K_{\text{max}} = 20 \), downhill iteration \( I_{\text{max}} = 60 \), global iteration \( M_{\text{max}} = 120 \), record deviation \( \alpha = 1\% \), and then generate initial solution |
| 3: \( x \) using nearest neighbor \( (x^* \leftarrow x) \) |
| 4: step (1) uphill move: local searches with randomize movements |
| 5: set \( K = 0 \) |
| 6: while \( K \leq K_{\text{max}} \) |
| 7: one-point-move(1-0), if the solution \( x \leq (1 + \alpha)x^* \) then accept the solution, \( K++ \) |
| 8: set \( K = 0 \) |
| 9: while \( K \leq K_{\text{max}} \) |
| 10: one-point-move(1-1), if the solution \( x \leq (1 + \alpha)x^* \) then accept the solution, \( K++ \) |
| 11: step (2) downhill move: local searches with systematic movements |
| 12: while \( I \leq I_{\text{max}} \) |
| 13: one-point-move(1-0) \( \rightarrow \) one-point-move(1-1), if the solution \( x \) better than \( x^* \) then \( x^* \leftarrow x \), |
| 14: back to step (2); else \( I++ \) |
| 15: step (3) global iteration: updating the global iteration |
| 16: if \( M \leq M_{\text{max}} \) then \( M++ \) and back to step (1); else go to step (4) |
| 17: step (4) report: the best solution obtained, \( x^* \) |

**Figure 1.** The proposed RRT metaheuristic algorithm

Based on Figure 1, there are five main steps in RRT algorithm. Step (0) is the initialization step to set the parameter value and the nearest neighbor algorithm for initial solution. Step (1) is local searches with randomize movements (uphill move). The procedure of movement starts with choosing a node randomly and move to other position. An example of uphill move can be seen in Figure 2. Different from VNS, RRT can accept a worse solution on each iteration with tolerance criteria. If the solution \( x \leq (1 + \alpha)x^* \) then update the solution. Step (2) is a downhill move that focused on systematic movement for almost all possible changes. An example of downhill move can be seen in Figure 3. Step (3) is updating the global iteration until \( M_{\text{max}} \). Step (4) is the report of the best solution.
2.4. The proposed I-DSS software implementation

3.4GHz with 8GB RAM computer is used to develop I-DSS software using Visual Studio 2017 software. Rapid application development is used to develop I-DSS software. System requirement, software specification, and performance test is done by several iteration with interviewing and testing by tour and travel company user (routing operator). The user interface of I-DSS software can be seen in Figure 4 and Figure 5.

3. Results and discussion

The results of this proposed I-DSS software tested using real historical data (tourist attraction, distance, opening hours, estimation of fuel consumption cost for each vehicle type, num. of passenger) from one of tour and travel company in Bali, Indonesia. The company provides travel services using a bus to visit
several tourist attractions based on the distance and opening hours of the destinations. There are several tourist attractions in Bali, such as Desa Sade, Pantai Kuta, Tanjung Aan, Sukarare, M. Hotel, Bukit Malimbu, Gili Tour, Narmada. The vehicle starts and finish at Bali international airport. This research compares three similar tour and travel company (similar tour package) with the output from proposed I-DSS software. The results comparison can be seen in Table 1. Based on Table 1, Figure 6, and Figure 7, the proposed I-DSS produce a good solution compared to other companies output in real conditions (total distance 188.8 km, with total fuel consumption cost Rp. 135,096). The proposed I-DSS has implications for the fast-computational time to determine the best route on the tour and travel company that can minimize total travel costs. Besides that, various combination related to the route constraints can be included in the software depend on real case situations.

Table 1. The results comparison.

| Company ID | Tour bus route | Total distance (km) | Total fuel consumption (Rp) |
|------------|----------------|---------------------|----------------------------|
| Comp. 1    | Day 1: Airport – Sukarare – Pantai Kuta – Tanjung Aan – Desa Sade – M. Hotel Day 2: M. Hotel – Narmada – Bukit Malimbu – Gili Tour – Bukit Malimbu – M. Hotel | 217.6 | 148,980 |
| Comp. 2    | Day 1: Airport – Bukit Malimbu – Gili Tour – Narmada – M. Hotel Day 2: M. Hotel – Desa Sade – Sukarare – Pantai Kuta – Tanjung Aan – Airport | 237.2 | 158,535 |
| Comp. 3    | Day 1: Airport – Desa Sade – Pantai Kuta – Tanjung Aan – Sukarare – M. Hotel Day 2: M. Hotel – Bukit Malimbu – Gili Tour – Narmada – M. Hotel | 202.5 | 141,618 |
| I-DSS      | Day 1: Airport – Desa Sade – Pantai Kuta – Tanjung Aan – Sukarare – M. Hotel Day 2: M. Hotel – Bukit Malimbu – Gili Tour – Narmada – Airport | 188.8 | 135,096 |

4. Conclusion

This research shows that the proposed algorithms produce a good and competitive solution and can be implemented in intelligent decision support systems (I-DSS). The results show that the travel distance is 25.64% more efficient and the fuel consumption is better than the actual conditions (17.35% more efficient). The result of this research proves that the algorithms can be used to solve complex TSP models such as multi-period TSP with time-windows in I-DSS software. For future works, several local search algorithms can be added, such as two-point-move (2-0 or 2-2), 2-opt intra-route, OR-opt, perturbation procedure, cross tail movements, or math-heuristic (including dijkstra algorithm) to improve existing solutions.

References

[1] Xu J, Pei L, Zhu R 2018. Application of a genetic algorithm with random crossover and dynamic mutation on the travelling salesman problem 8th International Congress of Information and Communication Technology p 937–45.
[2] Cheng C-B and Mao C-P 2007 A modified ant colony system for solving the travelling salesman problem with time windows Math Comput Model 46(9–10) 1225–35
[3] Pandiri V and Singh A 2018 A hyper-heuristic based artificial bee colony algorithm for k-interconnected multi-depot multi-traveling salesman problem Inf Sci (Ny) 8
[4] Osaba E, Del Ser J, Iglesias A and Bilbao N 2018 Solving the open-asymmetric green traveling salesman problem in a realistic urban environment In: Intelligent Distributed Computing XII p 181–91
[5] Wibowo S and Deng H 2012 Intelligent decision support for effectively evaluating and selecting ships under uncertainty in marine transportation Expert Syst Appl. 39(8) 6911–20
[6] He S, Song R and Chaudhry S S 2014 Service-oriented intelligent group decision support system: Application in transportation management Inf Syst Front 16 939–51
[7] Li F, Golden B and Wasil E 2007 A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem Comput Oper Res. 34(9) 2734–42
[8] Ramadhan F, Imran A and Rizana AF 2018 A record-to-record travel algorithm for multi-product and multi-period inventory routing problem In 5th International Conference on Industrial Engineering and Applications p 274–8
[9] Ramadhan F, Imran A and Rizana AF 2019 Threshold based algorithms for the multi-product multi-period inventory routing problem J Eng Sci Technol 14(5) 2703–15
[10] Ramadhan F and Imran A 2019 A two-phase metaheuristic method for solving travelling repairman problem In 2019 International Conference on Sustainable Engineering and Creative Computing
[11] Imran A and Ramadhan F 2019 Solution of the capacitated vehicle routing problem using variable neighborhood search with threshold In 2019 International Conference on Sustainable Engineering and Creative Computing
[12] Flood M M 1956 The traveling-salesman problem Operations research 4(1) 61-75
[13] Savelsbergh M W P 1985 Local search in routing problems with time windows Ann Oper Res. 4 285–305
[14] Mladenović N and Hansen P 1997 Variable neighborhood search Comput Oper Res. 24(11) 1097–100
[15] Jarboui B, Derbel H, Hanafi S and Mladenovic N Variable neighborhood search for location routing Comput Oper Res. 40(1) 47–57
[16] Mjirda A, Jarboui B, Macedo R and Hanafi S 2012 A variable neighborhood search for the multi-product inventory routing problem Electron Notes Discret Math. 39 91–8
[17] Imran A, Luis M and Okdinawati L 2016 A variable neighborhood search for the heterogeneous fixed fleet vehicle routing problem J Teknol. 78(9) 53–8