ENHANCED KNOWLEDGE IN SCIENCES AND TECHNOLOGY
e-ISSN: 2773-6385

EKST

Vol. 6 No. 1 (2026) 501-511
https://publisher.uthm.edu.my/periodicals/index.php/ekst

Tourist Route Optimisation in Melaka and Penang Using Nearest Neighbour and GRASP Algorithms

Yung Qing Xiu[1], Wan Nor Munirah Ariffin[1]*

[1] *Department of Mathematics and Statistics, Faculty of Applied Sciences and Technology, UTHM Kampus Cawangan Pagoh, Hab Pendidikan Tinggi Pagoh, KM1, Jalan Panchor, 84600 Pagoh, Muar, Johor, MALAYSIA.*

*Corresponding Author: normunirah@uthm.edu.my
DOI: https://doi.org/10.30880/ekst.2026.06.01.056

Article Info

Received: 4 January 2026
Accepted: 6 June 2026
Available online: 9 July 2026

Keywords

Tourist Route Optimisation, Travelling Salesman Problem, Graph Theory, Nearest Neighbour, GRASP, Melaka, Penang

Abstract

Efficient tourist route planning is crucial for enhancing travel experience and mobility in heritage cities. However, tourists often face difficulties in determining an optimal visiting sequence for multiple attractions which results in inefficient travel routes and increased travel distance. This research proposes a graph-based route optimisation framework for tourist attractions in Melaka and Penang, Malaysia by using the Nearest Neighbour and GRASP algorithms. Major attractions are modelled as nodes while travel distances obtained from the Google Maps Distance Matrix API are represented as weighted edges. Two heuristic approaches, which are the Nearest Neighbour (NN) algorithm and the Greedy Randomised Adaptive Search Procedure (GRASP), are implemented and compared in terms of total travel distance and computation time. Empirical results show that NN produces solutions with negligible computation time but longer routes. In contrast, GRASP consistently yields shorter routes with an average improvement of approximately 6.00–7.00% at an acceptable higher computational effort. The findings demonstrate that GRASP is more suitable for practical tourism route optimisation, where solution quality is prioritised. Future work may incorporate real-time traffic conditions, individual tourist preferences, and dynamic environmental factors further to enhance the practicality and adaptability of the proposed approach.

1. Introduction

In countries rich in historical and cultural heritage, tourism plays a crucial role in driving economic growth, preserving cultural heritage, and fostering regional development. In 2023, the tourism sector contributed 15.1% to Malaysia's national GDP and provided job opportunities for approximately 21.4% of the total labour force [1]. The tourism sector plays a significant role in the national development. Every year, many domestic and international visitors are attracted by heritage cities like Melaka and George Town (Penang). These two cities possess exceptional universal value, a multicultural trading history, and well-preserved urban landscapes, which is why they were jointly designated as UNESCO World Heritage Sites in 2008 [2]. These cities have highly concentrated tourist activities, and challenges such as mobility, congestion, and itinerary planning arise.

Researchers have demonstrated that it is essential to enhance tourist experiences while also supporting local economies and sustainable development. [3] found that the satisfaction of Melaka visitors was strongly influenced by convenience, variety of attractions, and historical significance. Additionally, [4] emphasised that optimised routes can enhance local economic input and inclusiveness. [5] further highlighted how Penang's cruise tourism is affected by the environment and the importance of preserving the heritage and ecosystem in Penang. [6] also emphasised that enhancing connections between heritage and natural sites is crucial, allowing pedestrian traffic to be distributed more evenly and thereby improving the tourist experience.

Usually, tourists prioritise visiting many attractions within a limited time. They want the shortest travel distance, the shortest travel time, and lowest associated costs. However, many of them rely on conventional navigation tools or static travel itineraries that they get from guidebooks and online blogs. Although popular navigation applications such as Google Maps do provide them with accurate point-to-point directions, there is no optimal sequencing for multiple destinations in a single trip [7]. Consequently, issues such as inefficient routes, unnecessary backtracking, and increased travel time occur, which in turn decrease their overall travel satisfaction [8].

From an operations research perspective, multi-destination route planning problems can be effectively formulated as the Travelling Salesman Problem (TSP). The TSP seeks to determine the shortest possible route that visits each location exactly once and returns to the starting point. Due to its broad applicability, the TSP has been extensively studied in logistics, transportation, and supply chain management, and has more recently gained attention in tourism route optimisation and itinerary planning [9], [10]. A systematic and quantitative framework is produced by TSP-based approaches to optimise tourists' routes. It is done through modelling tourist attractions as nodes and travel distances as weighted edges in a graph. As a result, tourist flows can be balanced, travel distances and times can be reduced, and the overall visitor experience can be improved [6], [10], [11].

Besides its practical relevance, the TSP is also considered an NP-hard problem. This means that as the number of destinations increase, the exact solution methods become computationally impractical. Heuristic and metaheuristic algorithms are applied to obtain near-optimal solutions within reasonable computation times, allowing this problem to be solved. Since simple heuristics, such as the Nearest Neighbour (NN) algorithm, are easily implemented and have very low computational cost, they are attractive. However, suboptimal solutions that make greedy decisions based solely on local information are often produced by NN. In contrast, controlled randomness and iterative improvement are introduced by metaheuristic approaches, such as the Greedy Randomised Adaptive Search Procedure (GRASP). Therefore, local optima can be avoided in the search process, and higher-quality solutions can be generated [12], [13].

Motivated by the need for efficient and practical tourist route planning tools, this research proposes a graph-based route optimisation framework for Melaka and Penang using real-world road distance data. The research applies and compares the NN and GRASP algorithms to evaluate their performance in terms of total travel distance and computation time. This research integrates optimisation techniques with real-world tourism applications. The main contributions of this paper are as follows: (i) modelling popular tourist attractions in Melaka and Penang as weighted graphs using realistic road distances, (ii) implementing NN and GRASP algorithms for tourist route optimisation, and (iii) providing a comparative analysis to support data-driven decision-making in smart tourism and urban mobility planning.

2. Methodology

This section describes the data used in the research, the problem modelling approach, and the optimisation procedures used to generate efficient tourist routes for Melaka and Penang. The research methodology involves three phases: the formation of nodes and edges based on tourism attractions, tourism attractions route optimisation, and model comparison. The data used in this research consists of the distances between major tourism attractions in Melaka and Penang. These attractions were selected based on their popularity, cultural significance, and inclusion in government tourism listings such as the Tourism Malaysia portal. This crucial data will be systematically gathered from various publicly accessible and authoritative sources, including official tourism board websites, reputable online mapping services, and comprehensive digital travel guides. The route optimisation process is implemented in the Python programming language. Popular libraries such as NetworkX are used to model tourist attractions as nodes and travel distances as weighted edges in a graph. The NumPy and Pandas libraries are utilised for data manipulation and distance matrix construction. Heuristic and metaheuristic algorithms, specifically the Nearest Neighbour (NN) and the Greedy Randomised Adaptive Search Procedure (GRASP), are coded using Python functions to generate and iteratively improve route solutions. Additionally, visualisation of the optimised routes and comparative analysis between algorithms are facilitated by Matplotlib and Seaborn libraries. To illustrate the overall research workflow, a flowchart is also included.

2.1 Data Description

The dataset used in this research consists of major tourist attractions in Melaka and Penang. A total of 19 attractions were selected for Melaka and 18 attractions for Penang based on their popularity and relevance to tourists. Each attraction is represented by its geographical location, identified using latitude and longitude coordinates.

Travel distance data between each pair of attractions was obtained using the Google Maps Distance Matrix API, which provides realistic road-based distances rather than straight-line distances. This ensures that the optimisation results reflect actual travelling conditions. The collected distance data were organised into distance matrices for each city and subsequently used as input for the optimisation algorithms. The 19 key attractions from Melaka are listed in Table 1 while the 18 key attractions from Penang are listed in Table 2.

Table 1 Tourist Attraction Places in Melaka

Tourists Attractions Places in Melaka
A Famosa
Baba Nyonya Heritage Museum
Cheng Hoon Teng Temple
Dataran Pahlawan Melaka Megamall
Hang Tuah Centre
Jonker Street
Klebang Beach
Maritime Museum
Melaka River Cruise
Melaka Straits Mosque
Portuguese Settlement
Red Square (Dutch Square)
St Paul's Hill
Submarine Museum
Sultanate Palace Museum
Taming Sari Tower
The Shore Sky Tower
Villa Sentosa (Malay Living Museum)
Zoo Melaka

Table 2 Tourist Attraction Places in Penang

Tourists Attractions Places in Penang
ESCAPE Penang
Batu Feringghi Beach
Kek Lok Si Temple
Penang Hill
Pinang Peranakan Mansion
Cheong Fatt Tze - The Blue Mansion
Armenian Street
Chew Jetty
Penang National Park (Taman Negara Pulau Pinang)
Street Art Penang
Entopia by Penang Butterfly Farm
Penang Botanic Gardens
THE TOP Penang
Wonderfood Museum Penang
Tropical Spice Garden
George Town
Dhammikarna Burmese Temple
Penang Floating Mosque

2.2 Phase I: Formation of Nodes and Edges based on Tourism Attraction

In the first phase, the selected attractions are mapped as nodes in a graph denoted by $G = (V, E)$, where each node, V, represents a destination and each edge, E, represents the travel distance between two destinations. This mapping is carried out separately for Melaka and Penang.

The steps for transforming tourism locations into a graph format by using the graph theory [14]:

Step 1 Identify popular tourist attractions in Melaka and Penang.
Step 2 Compute distances or durations between attractions.
Step 3 Build a weighted graph, where:
  i. Nodes, V = Tourist Attractions
  ii. Edges, E = Pairwise travel distances

2.3 Phase II: Tourism Attraction Route Optimisation

This phase applies optimisation techniques to determine the most efficient route that visits all selected tourist destinations in each city exactly once before returning to the starting point, as defined by the Travelling Salesman Problem (TSP) model.

The core problem can be modelled as the TSP as shown in the equations (1), (2) and (3):

$$Minimise \sum_{i \in V}^{n} \sum_{j \in V,\, j \neq i}^{n} c_{ij} x_{ij} \tag{1}$$

$$Subject\ to \sum_{i \in V,\, j \neq i}^{n} x_{ij} = 1, i = 1, \ldots, n \tag{2}$$

$$\sum_{j \in V,\, j \neq i}^{n} x_{ij} = 1, j = 1, \ldots, n \tag{3}$$

where c_{ij} be the distance between the attraction i and j and $x_{ij} \epsilon \{0, 1\}$ be a binary decision variable where 1 means the route from i to j is taken.

2.3.1 The Nearest Neighbour Algorithm

The Nearest Neighbour (NN) algorithm is a greedy heuristic which approximates a solution to TSP. It begins at a randomly chosen node, and then successively visits the closest unvisited node until all the nodes have been visited. Ultimately, it returns to the initial node to complete the route. NN may not always be fast and easy to implement, which is why it is a viable choice to use it as a basis of comparison.

Steps for the Nearest Neighbour Algorithm are outlined as below [15]:

Step 1 Select a starting node.
Step 2 From the current node, choose the nearest unvisited node.
Step 3 Mark that node as visited and move to it.
Step 4 Repeat step 2-3 until all nodes are visited.
Step 5 Return to the starting node.
Step 6 Calculate total route distance.

Total distance is calculated by using equation (4):

$$Total\ Distance = \sum_{i=0}^{n-1} W(v_i, v_{i+1}) + W(v_n, v_0) \tag{4}$$

where W is the weight of the edges.

2.3.2 The Greedy Randomised Adaptive Search Procedure Algorithm

The Greedy Randomised Adaptive Search Procedure (GRASP) algorithm is a metaheuristic method that combines elements of greedy construction with randomised choices and local search to generate solutions iteratively [12]. GRASP consists of two main phases: a construction phase and a local search phase, which are iteratively repeated.

Steps for the Greedy Randomised Adaptive Search Procedure algorithm are outlined as below:

Step 1 Initialise best solution = ∞.
Step 2 Repeat for a fixed number of iterations:
  i. Construct a solution using a greedy randomised approach.
  ii. Apply local search to improve the solution.
  iii. If improved, update best solution.

Step 3 Return the best solution.

2.4 Phase III: Model Comparison

The final phase is to compare the proposed model of optimising tourist routes. This is validated by comparing the results of the Nearest Neighbour and the Greedy Randomized Adaptive Search Procedure (GRASP) algorithms in terms of total travel distance and computation time. The comparison will be presented through clear tables and graphs, allowing for a direct visual and statistical assessment of the algorithms' performance. This phase will provide empirical evidence to determine which algorithm offers a more effective solution for optimising tourism routes in the specific contexts of Melaka and Penang.

The step for model comparison:

Step 1 Input results from route optimisation algorithms.
Step 2 Compute total travel distance and computation time.
Step 3 Compare results between both algorithms.
Step 4 Conclude on algorithm effectiveness.

The improvement percentage of algorithms is calculated by using equation (5):

$$Improvement\ percentage = \frac{\sum d_{NN} - \sum d_{GRASP}}{\sum d_{NN}} \times 100\% \quad (5)$$

where d is the total distance of the route for attractions.

2.5 Flowchart of Research

The overall workflow of the proposed methodology is summarised in Fig. 1.

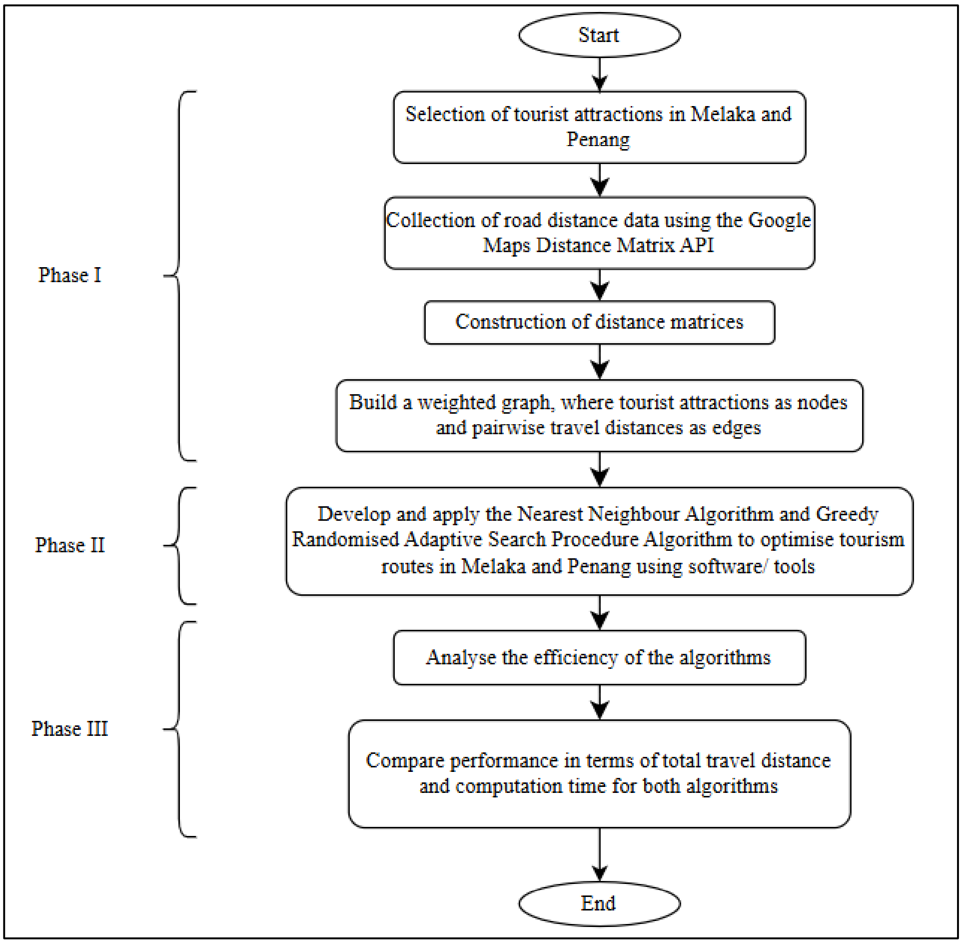

Fig. 1 Flowchart of Research

3. Results and Discussions

The input dataset consists of inter-location distances between major tourist attractions. For pairs of attractions that have multiple possible routes, only the shortest distance is selected to ensure the model represents the most efficient travel connections. When multiple routes exist between two attractions, the shortest distance is specified using the following rule in equation (6):

$$d_{ij}, \text{ Shortest Path} = \min\left\{d_1, d_2, d_3, \ldots, d_n\right\} \tag{6}$$

where d represents distances between tourist attractions.

The path for other locations with multiple paths can also be selected using the same method, while the path will remain for those nodes that only have one path. This ensures that each pair of attractions in the graph is connected using the minimum possible travel distance. The resulting weighted graph forms the foundation for route optimisation in Phase II.

Table 3 summarises the route optimisation results for Melaka. The total distance of the Nearest Neighbour algorithm is calculated by using Equation (4), while the improvement percentage is obtained by using Equation (5). The Nearest Neighbour algorithm achieved a total travel distance of 55.50 km with a computation time of 0.0001 seconds, while the GRASP algorithm produced a shorter route of 51.83 km but required 6.38 seconds to compute. This demonstrates that GRASP achieved a 6.61% improvement in route optimisation compared to NN.

Table 3 Comparison of NN and GRASP Results for Melaka

Metric	Nearest Neighbour	GRASP	Improvement (%)
Total Distance (km)	55.50	51.83	6.61
Computation Time (s)	0.0002	6.3784	—

The optimised GRASP route began and ended at St. Paul's Hill, covering major attractions such as Jonker Street, Klebang Beach, Christ Church, and the Melaka Straits Mosque. This route efficiently minimises total travel distance while maintaining logical connectivity between popular sites.

These findings show that GRASP yields superior solutions by exploring a more expansive search space; however, this comes at the cost of higher computational time but still in acceptable range. Conversely, the NN algorithm, while faster, tends to settle for suboptimal paths.

Table 4 Optimal solution and travel distance (km) for Melaka tourist attractions

Optimal Solution																				Total Distance (km)
13	12	6	2	7	14	3	17	18	19	5	11	10	4	16	8	9	15	1	13	51.83

From Table 4, location 13 is selected as the starting point, and the recommended tourist attraction place to visit next is location 12, followed by location 6 and so on. Table 5 shows the order of the of Melaka attractions for the optimal route.

Table 5 Order of Melaka attractions for optimal route

Order of Melaka attractions
1. St Paul's Hill
2. Red Square (Dutch Square)
3. Jonker Street
4. Baba Nyonya Heritage Museum
5. Klebang Beach
6. Submarine Museum
7. Cheng Hoon Teng Temple
8. The Shore Sky Tower
9. Villa Sentosa
10. Zoo Melaka

11. Hang Tuah Centre
12. Portuguese Settlement
13. Melaka Straits Mosque
14. Dataran Pahlawan Melaka Megamall
15. Taming Sari Tower
16. Maritime Museum
17. Melaka River Cruise
18. Sultanate Palace Museum
19. A Famosa
20. St Paul's Hill (Return to start)

Fig. 2 and Fig. 3 show the optimal route from GRASP for Melaka attractions on a map, which starts and ends at location 13. Fig. 4 illustrates the optimal route from GRASP for Melaka attractions, utilising graph theory.

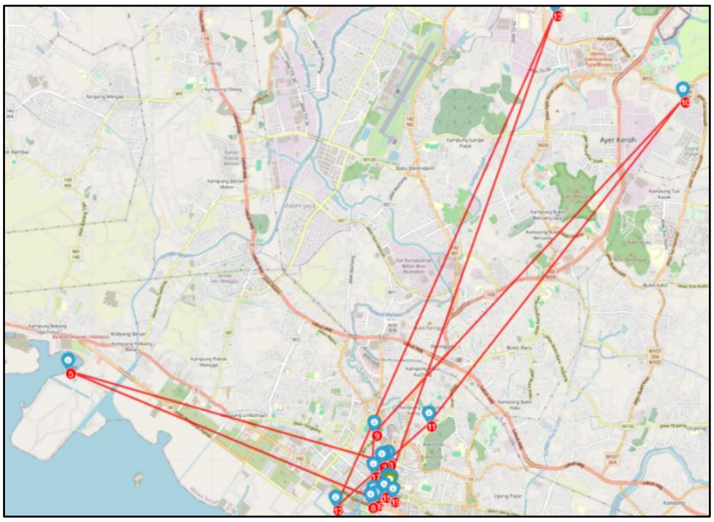

Fig. 2 Optimal route from the GRASP algorithm for Melaka attractions in the map

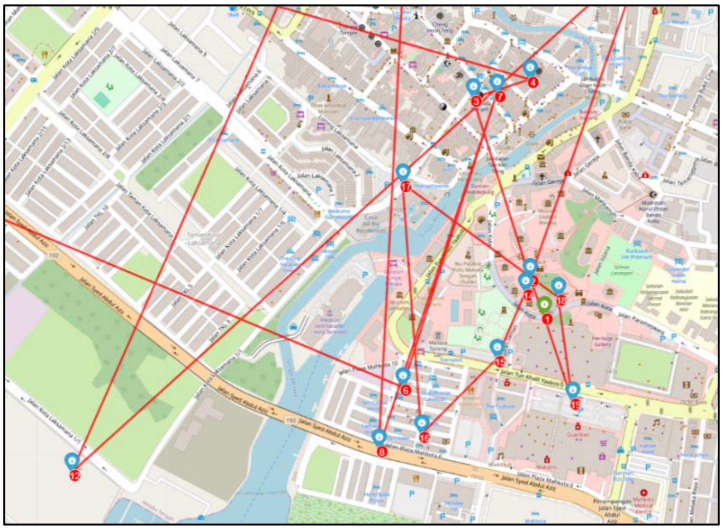

Fig. 3 Zoom-in view of the optimal route from the GRASP algorithm for Melaka attractions in the map

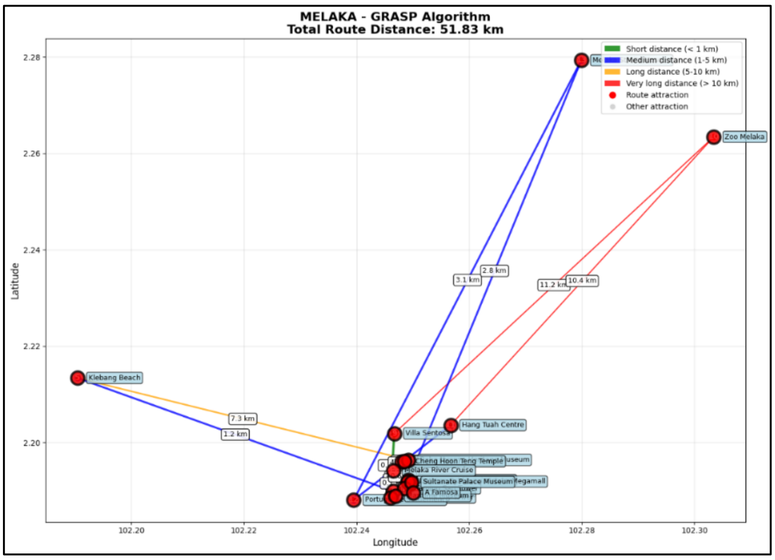

Fig. 4 Optimal route from the GRASP algorithm for Melaka attractions in the graph

The performance results for Penang are shown in Table 6. The total distance of the Nearest Neighbour algorithm is calculated by using Equation (4), while the improvement percentage is obtained by using Equation (5). The Nearest Neighbour algorithm generated a route of 65.92 km, while GRASP produced a shorter route of 61.75 km, marking an improvement of 6.33%. Similar to Melaka, NN performed faster but less accurately, while GRASP yielded better optimisation with longer computation time (2.87 seconds).

Table 6 Comparison of NN and GRASP Results for Penang

Metric	Nearest Neighbour	GRASP	Improvement (%)
Total Distance (km)	65.92	61.75	6.33
Computation Time (s)	0.0002	2.8650	—

The GRASP-optimised route covered key Penang attractions including Entopia Butterfly Farm, Penang Hill, Cheong Fatt Tze Mansion, George Town, and Kek Lok Si Temple. The route sequence maintained logical continuity, minimising backtracking and maximising travel efficiency.

Table 7 Optimal solution and travel distance (km) for Penang tourist attractions

Optimal Solution																			Total Distance (km)
11	15	18	17	6	5	14	8	10	7	13	16	3	4	12	2	9	1	11	61.75

From Table 7, location 11 is selected as the starting point, and the recommended tourist attraction place to visit next is location 15, followed by location 18 and so on. Table 8 shows the order of the Penang attractions for the optimal route.

Table 8 Order of Penang attractions for optimal route

Order of Melaka attractions
1. Entopia by Penang Butterfly Farm
2. Tropical Spice Garden
3. Penang Floating Mosque
4. Dhammikarna Burmese Temple
5. Cheong Fatt Tze - The Blue Mansion
6. Pinang Peranakan Mansion
7. Wonderfood Museum Penang

8. Chew Jetty
9. Street Art Penang
10. Armenian Street
11. THE TOP Penang
12. George Town
13. Kek Lok Si Temple
14. Penang Hill
15. Penang Botanic Gardens
16. Batu Feringghi Beach
17. Penang National Park
18. ESCAPE Penang
19. Entopia by Penang Butterfly Farm (Return to start)

Fig. 5 and Fig. 6 show the optimal route from GRASP for Penang attractions on a map, which starts and ends at location 11. Fig. 7 shows the optimal route from GRASP for Penang attractions using graph theory.

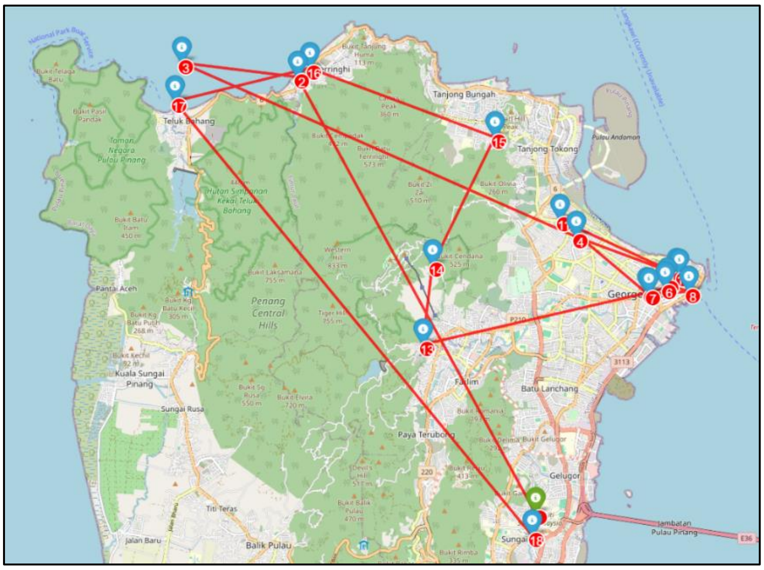

Fig. 5 Optimal route from the GRASP algorithm for Penang attractions in the map

Fig. 6 Zoom-in view of the optimal route from the GRASP algorithm for Penang attractions in the map

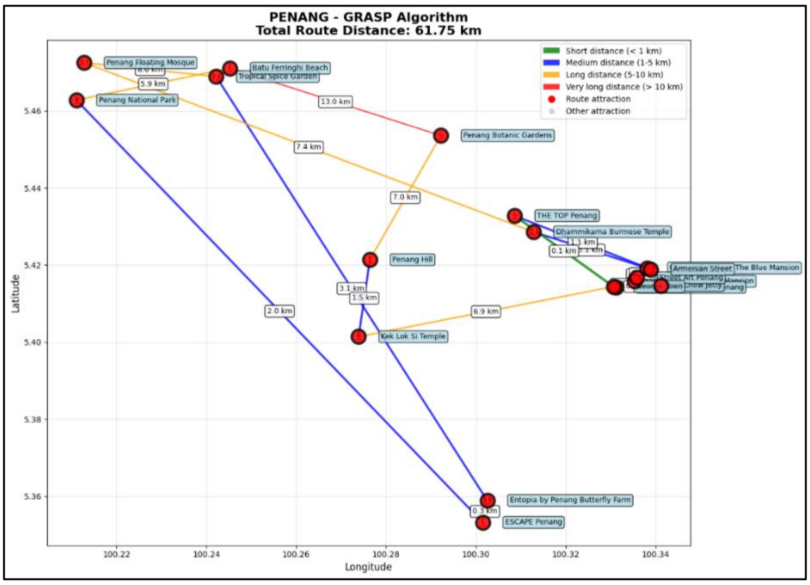

Fig. 7 Optimal route from the GRASP algorithm for Penang attractions in the graph

Both case studies demonstrate the contrasting nature of NN and GRASP algorithms. NN's main advantage lies in its computational simplicity and speed. However, its greedy nature limits the ability to escape local minima which often results in suboptimal tours. On the other hand, GRASP integrates randomness and iterative improvement, allowing it to explore multiple feasible solutions and achieve superior route efficiency.

GRASP achieved approximately 6.00–7.00% distance reduction compared to NN in both Melaka and Penang. Although the time cost was higher, the trade-off is justifiable in tourism route planning, where accuracy and total distance minimization are more critical than computational time. The findings confirm the effectiveness of GRASP in producing optimised travel paths, while NN remains a valuable benchmark for quick approximations.

These results provide a foundation for future research in tourism route optimisation, especially for integrating additional factors such as travel time, weather conditions, and tourist preferences into dynamic optimisation models.

4. Conclusion

This research successfully achieved the three objectives outlined in the proposal by constructing tourism route optimisation models for Melaka and Penang using graph-based representation. In objective 1, tourism attractions were modelled as nodes while distances acted as weighted edges, forming a complete graph used for route computation. Objective 2 implemented two heuristic optimisation techniques: Nearest Neighbour (NN) and Greedy Randomised Adaptive Search Procedure (GRASP), to generate travel sequences and compute total route distances. Objective 3 compared both algorithms based on total travel distance and computation time.

The results demonstrated that GRASP consistently generated shorter route distances than NN, with improvements of 6.61% for Melaka and 6.33% for Penang, which proves its ability to escape local minima and explore more optimal solutions. However, NN achieved significantly lower computation time which makes it suitable for fast decision-making or real-time routing applications. GRASP required more processing time due to iterative refinement but produced more efficient tours, especially for larger datasets. Overall, the research confirms that algorithm selection should consider dataset size, optimisation priority, and time constraints. The combination of graph modelling and heuristic optimisation provides a practical and effective approach for tourist route planning and could be implemented in recommendation or navigation systems.

Based on the results obtained, several recommendations are proposed for future improvement of this research. Firstly, the model can be enhanced by integrating real-time factors, such as traffic conditions, peak tourist hours, and weather information, as these dynamic elements can significantly impact route efficiency and travel time. Secondly, the current optimisation treats every attraction equally; therefore, future work should consider user-based preferences, such as cultural interests, food spots, family-friendly places or cost-based route selection to create more personalised travel planning. The system can also be extended into multi-day itinerary planning, especially for tourists who require scheduled rest stops or hotel selection. In addition, developing a mobile or web-based interface would make the model more practical for real-world use, allowing routes to be visualised and navigated directly. Finally, researchers are encouraged to explore hybrid or alternative metaheuristics, such as Ant Colony Optimisation, Particle Swarm Optimisation, Genetic Algorithm or Simulated Annealing, for potential improvements in optimisation performance. These enhancements would not only broaden the applicability of the routing system but also strengthen its accuracy, flexibility and value for tourism planners and travellers.

Acknowledgement

Appreciation is also extended to Universiti Tun Hussein Onn Malaysia (UTHM) for providing the facilities and resources necessary for the completion of this research. The authors would like to thank the Faculty of Applied Science and Technology, University Tun Hussein Onn Malaysia, for its support.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Yung Qing Xiu;* ***data collection:*** *Yung Qing Xiu;* ***analysis and interpretation of results:*** *Yung Qing Xiu, Wan Nor Munirah Ariffin;* ***draft manuscript preparation:*** *Yung Qing Xiu, Wan Nor Munirah Ariffin. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Department of Statistics Malaysia. (2024). *Tourism Satellite Account 2023.* https://www.dosm.gov.my/portal-main/release-content/tourism-satellite-account-2023

[2] UNESCO (2008). *Melaka and George Town, Historic Cities of the Straits of Malacca.* https://whc.unesco.org/en/decisions/1486/

[3] Abidin, Z. Z., Latiff, H. A., & Hazizy, N. A. (2021). Tourists' perceptions of attraction factors in Melaka. *Politeknik & Kolej Komuniti Journal of Social Sciences and Humanities*, *6*(1), pp. 31-44. https://myjms.mohe.gov.my/index.php/PMJSSH/article/view/16308

[4] Amir, S. *et al.* (2015). Sustaining local community economy through tourism: Melaka UNESCO World Heritage City. *Procedia Environmental Sciences*, *28*, pp. 443–452. https://doi.org/10.1016/j.proenv.2015.07.054

[5] Tee, A. M. K. (2025). Environmental Analysis on Cruise Tourism: Case Study of Penang. *Journal of Responsible Tourism Management*, *5*(1), pp. 76–95. https://doi.org/10.47263/jrtm.05-01-06

[6] Elkaftangui, M. (2024). Addressing connectivity issues between the historical and natural touristic heritage sites of Egypt. In: Chica-Olmo, J. *et al.* (Eds.) *Sustainable Tourism, Culture and Heritage Promotion*. *CST 2022*. Advances in Science, Technology & Innovation, pp. 175–180. https://doi.org/10.1007/978-3-031-49536-6_18

[7] Jovanovic, A., Gavric, S., & Stevanovic, A. (2024). Evaluating Google Maps' Eco-Routes: A Metaheuristic-Driven Microsimulation Approach. *Geographies, 4(4)*, pp. 732–752. https://doi.org/10.3390/geographies4040040

[8] Souffriau, W. *et al*. (2008). A personalized tourist trip design algorithm for mobile tourist guides. *Applied Artificial Intelligence*, *22(10)*, pp. 964–985. https://doi.org/10.1080/08839510802379626

[9] Laporte, G. (2009). A concise guide to the Traveling Salesman Problem. *Journal of the Operational Research Society*, *61*(1), pp. 35–40. https://doi.org/10.1057/jors.2009.76

[10] Shinkarenko, V., Nezdoyminov, S., Galasyuk, S., & Shynkarenko, L. (2020). Optimization of the tourist route by solving the problem of a salesman. *Journal of Geology, Geography and Geoecology*, *29*(3), pp. 572-579. https://cyberleninka.ru/article/n/optimization-of-the-tourist-route-by-solving-the-problem-of-a-salesman

[11] Saad, S., Jaafar, W. N. W., & Jamil, S. J. (2013). Solving standard traveling salesman problem and multiple traveling salesman problem by using branch-and-bound. *AIP Conference Proceedings*, pp. 1406–1411. https://doi.org/10.1063/1.4801294

[12] Feo, T. A., & Resende, M. G. C. (1995). Greedy randomized adaptive search procedures. *Journal of Global Optimization*, *6*(2), pp. 109–133. https://doi.org/10.1007/BF01096763

[13] Resende, M. G. C., & Ribeiro, C. C. (2010). Greedy randomized adaptive search procedures: Advances, Hybridizations, and applications. In Gendreau, M. & Potvin, J.-Y. (Eds.), *Handbook of Metaheuristics*, pp. 281–317. Springer. https://doi.org/10.1007/978-1-4419-1665-5_10

[14] Wilson, R. J. (1996). *Introduction to graph Theory* (4th ed.). Addison Wesley Longman. https://webhomes.maths.ed.ac.uk/~v1ranick/papers/wilsongraph.pdf

[15] Matai, R., Singh, S., & Mittal M. L. (2010). Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches. In: Davendra D. (Eds.) *Traveling Salesman Problem, Theory and Applications*, pp. 1–24. In InTech eBooks. https://doi.org/10.5772/12909