TERM PROJECT REPORT

1.Gurobi Solution

Gurobi, an optimization tool, was employed in our project to determine the optimal solution for the Vehicle Routing Problem (VRP). Notably, a specific time limit was not set for the optimization process. Upon initiating the Gurobi solver, we allowed it to run for 25.16 minutes in anticipation of obtaining an optimal solution. Regrettably, the solver did not converge to an optimal solution within the allocated time frame, We ended the execution manually.

Subsequently, we obtained a solution from Gurobi, which, according to the solver's output, carries an optimality gap of 24.8%. The Gurobi answer status, denoted as 11, signifies that optimization was terminated by the user. The decision to interrupt the code execution was driven by the extended runtime of 25.16 minutes without achieving optimality.

INTERRUPTED	11	Optimization was terminated by the user.

- The tour that we found in gurobi with 24.8% optimality gap is:

  [[0, 12, 21, 8, 18, 3, 10, 20, 2, 15, 29, 28, 11, 0]), ([0, 23, 26, 1, 24, 14, 9, 0]),([0, 27, 22, 16, 13, 30, 6, 25, 7, 5, 17, 4, 19, 0]]

- The total tour length we found with Gurobi is 2453.5 (objective value)
- The total demand in each cluster is [120, 60, 120], no cluster exceeds the capacity of the vehicle(120) means that this solution is feasible.

Gurobi is capable of identifying optimal solutions; however, when dealing with large-scale problems, such as the one under consideration, the computational time required for Gurobi to achieve optimality may extend to days, weeks, or even decades. Consequently, to address the impractical timeframes associated with Gurobi's exhaustive optimization process for such extensive problems, heuristic algorithms are employed. These heuristics provide sufficiently good feasible solutions within significantly shorter timeframes compared to Gurobi's exhaustive search.

While Gurobi is acknowledged for its ability to yield optimal solutions, the substantial size of the problem in question hinders the practical attainment of an optimal solution within a reasonable timeframe. In light of this challenge, a comparative analysis with heuristic solutions is pursued. To facilitate this comparison, we will compare the answer that we get from heuristic (with computational time under a minute) and gurobis solution(with computational time of 25.16 minute). This metric is computed using the following formula:

((heuristic solution tour length – gurobi solution tour length)/(gurobi solution tour length))*100

This formula enables to find the gap of between the tour lengths of heuristic solutions and the tour length generated by Gurobi, even though it is not optimal. We will mention this formula as the "Gurobi solution gap" in the report

2. Heuristic Solutions

a) Apply saving algorithm for VRP
   - The tour that we found in part a is [[0, 16, 13, 30, 6, 25, 7, 12, 21, 8, 18, 3, 0], [0, 22, 17, 5, 11, 10, 20, 15, 2, 29, 28, 26, 1, 0], [0, 27, 14, 4, 19, 24, 23, 9, 0]].
   - The total tour length we found in part a is 2542.2
   - The gurobi solution gap is 3.615%
   - Total demand in each cluster is [110, 120, 70], there is no cluster which exceed the capacity of the vehicle(120) means that this solution is feasible.

b) *Apply 2-exchange algorithms exhaustively (try all possible improvements) to improve the solution you obtained from 1.*
   - The tour that we found in part b is [[0, 16, 13, 30, 6, 25, 7, 21, 8, 12, 18, 3, 0], [0, 22, 17, 5, 11, 10, 20, 2, 15, 29, 28, 26, 1, 0], [0, 27, 14, 4, 19, 24, 23, 9, 0]] node 2 and node 15 swapped.
   - The tour length we found with that improvement is 2522.9
   - The tour length that we found in part a is slightly improved with the improvement that we applied in part b(2542.2 – 2522.9 = 19.3)
   - The gurobi solution gap decreased to 2.828%
   - Total demand in each cluster is [110, 120, 70], there is no cluster which exceed the capacity of the vehicle(120) means that this solution is feasible.

c) *Apply 2-opt algorithms greedily and stop when the first improvement is recognised to improve the solution you obtained from part b.*
   - The tour that we the improvement in part c is [[0, 16, 13, 30, 6, 25, 7, 21, 8, 12, 18, 3, 0], [0, 1, 26, 28, 29, 15, 2, 20, 10, 11, 5, 17, 22, 0], [0, 27, 14, 4, 19, 24, 23, 9, 0]]
   - The tour length we found with that improvement is 2520.7
   - The tour length according to solution obtained from part c negligibly improved (2522.9 - 2520.7 = 2.2)
   - The gurobi solution gap decreased to 2.738%
   - Total demand in each cluster is [110, 120, 70], there is no cluster which exceed the capacity of the vehicle(120) means that this solution is feasible.

Conclusion

In conclusion, while Gurobi excels in determining optimal solutions, it may face challenges in real-world scenarios characterized by large-scale problems and time constraints. The decision to employ an optimization tool necessitates a thoughtful consideration of the trade-off between obtaining the absolute best result and ensuring computational efficiency. In our specific problem, Gurobi failed to produce an optimal solution within a time frame of 25 minutes. After 25 minutes of execution, no optimal solution was obtained, leading us to interrupt the code. Consequently, we utilized the solution with a 24.8% optimality gap as a practical alternative for further analysis and comparison with solutions derived from heuristic algorithms. Heuristic algorithms, though not guaranteeing optimal solutions, prove effective in providing practical solutions even in situations with limited computational resources. The optimality gap serves as a critical metric for understanding the compromise between optimality and efficiency in this trade-off. In our case, we opted for the Gurobi solution as a representative alternative to the optimal solution, aiming to delineate differences between Gurobi and heuristic solutions. The heuristic implementations gave a solution under a minute with a 2.738% difference from the Gurobi's solution, that Gurobi found in 25 minutes with a 24.8% optimality gap.