TU Wien
Institute for Logic and Computation
Algorithms and Complexity Group

Heuristic Optimization Techniques, WS 2019

Programming Exercise: Assignment 1

v1, 2019-10-23

Work on the problem within your programming exercise group. You can and are encouraged to split tasks evenly but everybody has to understand and be able to explain all the concepts involved, your overall implementation and the submitted report. Hand in your complete report **and** submit the solutions for the instances **and** upload your source code via TUWEL by **Sunday, 1st December 2019, 23:55**. For further questions please send an e-mail to: `heuopt@ac.tuwien.ac.at`

The first programming assignment is to develop single-solution-based metaheuristics for the Traveling Salesperson Problem with Target Distances for Multiple Drivers (TSPTDMD) in your preferred programming language. The problem description is given in the *general information* document. The subtasks for this exercise are:

1. Give a formula to calculate a reasonable big-M constant $M \in \mathbb{N}$ for a given graph G with edge weights $d_{i,j}$. Prove that when using M as weights for all non-existing edges, the objective of any "infeasible" solution is greater than the objective of every feasible solution. *Hint: What is the worst-case for the tour and corresponding assignment? How can you upper bound the objective for this worst case?*
2. Develop a deterministic construction heuristic.
3. Develop a randomized construction heuristic.
4. Develop or make use of a framework for basic local search which is able to deal with
   - different neighborhood structures
   - different step functions (first-improvement, best-improvement, random)
5. Develop at least three different neighborhood structures.
6. Implement a GRASP using your randomized construction heuristic and one of your neighborhood structures with one step function.
7. Develop or make use of a VND framework which uses your neighborhood structures.
8. Implement one of the following metaheuristics:
   - General Variable Neighbourhood Search (GVNS) which uses your VND
   - Simulated Annealing (SA)
   - Tabu Search (TS)
9. Run experiments and compare all your algorithms on the instances provided in TUWEL:
   (a) deterministic and randomized construction heuristic and GRASP
   (b) Use the solution of the deterministic construction heuristic to test the other implementations:
      i. Local search for each of your neighborhood structures using each of the three step functions (at least 9 different algorithms)
      ii. VND
      iii. GVNS, SA, or TS
10. **Use delta-evaluation whenever possible**. Test your approaches once with incremental evaluation and once without incremental evaluation and compare the performance and running times.

11. Write a report containing the description of your algorithms and the experiment results (see the *general information* document for more details)

For the development and the report consider the following points:

- How is your solution represented?
- Ad 3: How do you generate different solutions? Which parts of your algorithm can be reasonably randomized and how can you control the degree of randomization?
- Ad 2 & 3: Does randomization improve the generated solutions?
- What parameters do you use and which values do you choose for them?
- Consider the composite structure of a solution, consisting of the tour and the drivers' assignment, when developing different neighborhood structures.
- Can subsequent – possibly non-improving – moves in your neighborhood structures reach every solution in the search space?
- Local search: How many iterations does it take to reach local optima? What does this say about your neighborhood structures?
- How does delta-evaluation work for your neighborhood structures?
- VND: Does the order of your neighborhoods affect the solution quality?