# INF273 – Assignment 3

Lukas Schramm 20th February 2022

## 1 Functions

Currently, I am maintaining the following helper functions in my Utils file.[1]

- load_problem: Given a path to a file, it reads the content of the file into a dictionary of information.
- feasibility_check: It takes a solution (list) and a problem dictionary and checks if the solution is feasible. If it is not feasible, it outputs the reason why. It does not check validity (since the aim
- cost_function: It takes a solution (list) and a problem dictionary and calculates the cost of the function. As feasibility_check it does not check if the original solution was valid.
- splitting_a_list_at_zeros: Helper function which splits a solution into vehicles and if needed a dummy vehicle.
- initial_solution: Generates an initial default solution to start with. This is always a solution where the dummy vehicle handles all calls.
- random_solution: Generates a random solution. The generator itself is quite bad in my view because I overtuned it a bit. It automatically gives one vehicle exactly one call and the rest goes to the dummy vehicle. That way I got solutions for file 3 and 4 but the solutions for all files are quite bad.[2]
- blind_random_search: Takes a problem and a number of iterations to find the best out of n random feasible solutions if any is found.
- blind_search_latex_generator: This function runs the blind_random_search and writes the data into LaTeXtables since I am obviously too lazy to do it myself.
- latex_add_line: Adds a new result line into an results table of this file.
- latex_replace_line: Change the optimal solution and its seed in that file.

  If there are any questions or nice recommendations to get a better structure, just send me a message.

[1] The green ones are changes or additions from the last assignment

[2] But since we do not need that random solution generator any longer I keep it like that.

Moreover, there is a new file for Heuristics where I collect all of the important algorithms and their helper functions.

- alter_solution_1insert: A function which takes a current solution and outputs a next solution by using the 1-insert-operation. The output is not necessary feasible, but of course valid
- alter_solution_2exchange: A function which takes a current solution and outputs a next solution by using the 2-exchange-operation. The output is not necessary feasible, but of course valid
- alter_solution_3exchange: A function which takes a current solution and outputs a next solution by using the 3-exchange-operation. The output is not necessary feasible, but of course valid
- local_search: This function takes a problem, an initial solution, a number of iterations (10.000) and the allowed neighbouring function and performs a local search
- simulated_annealing: This function takes a problem, an initial solution, a number of iterations (10.000) and the allowed neighbouring function and performs a simulated annealing
- local_search_sim_annealing_latex: This function takes as input the allowed neighbouring function(s), the heuristics method, the problem and the number of iterations and performs the heuristics on randomly chosen seeds. It then calculates the average time and objective and runs the LaTeXfunctions to change the tables of this PDF

## 2 Result tables

Table 1: Call_7_Vehicle_3

| Method | Average objective | Best objective | Improvement (%) | Running time |
|---:|---:|---:|---:|---:|
| Random search | 2289893.35 | 2120884 | 34.59% | 0.62s |
| Local Search-1-insert | 1002862.60 | 903963 | 72.12% | 1.45s |
| Local Search-2-exchange | 993772.90 | 905099 | 72.09% | 1.06s |
| Local Search-3-exchange | 983794.20 | 905099 | 72.09% | 1.18s |
| Simulated Annealing-1-insert | 1008944.70 | 934580 | 71.18% | 1.08s |
| Simulated Annealing-2-exchange | 990368.00 | 903963 | 72.12% | 1.16s |
| Simulated Annealing-3-exchange | 991212.50 | 905099 | 72.09% | 1.19s |

Listing 1: Optimal solution call_7_vehicle_3

```
sol = [4, 4, 0, 0, 2, 1, 2, 5, 5, 3, 3, 1, 6, 6, 7, 7, 0]
seeds = [140411156, 578424865, 16569813, 217022394, 14341897, 70767204,
    463379884, 424556384, 624532274, 234736142]
```

Table 2: Call_18_Vehicle_5

| Method | Average objective | Best objective | Improvement (%) | Running time |
|---:|---:|---:|---:|---:|
| Random search | 7195792.08 | 6215552 | 29.80% | 0.80s |
| Local Search-1-insert | 2247932.40 | 1948946 | 78.25% | 1.83s |
| Local Search-2-exchange | 2272580.10 | 2086534 | 76.71% | 2.24s |
| Local Search-3-exchange | 2350191.10 | 2017793 | 77.48% | 2.11s |
| Simulated Annealing-1-insert | 2177814.40 | 1999101 | 77.69% | 1.93s |
| Simulated Annealing-2-exchange | 2213670.00 | 2047042 | 77.15% | 1.83s |
| Simulated Annealing-3-exchange | 2234001.00 | 2060878 | 77.00% | 1.80s |

Listing 2: Optimal solution call_18_vehicle_5

```
sol = [15, 15, 4, 4, 16, 16, 11, 11, 0, 8, 7, 7, 9, 8, 9, 13, 13, 0, 17,
     2, 17, 2, 6, 6, 5, 5, 18, 18, 0, 3, 3, 10, 1, 1, 12, 10, 12, 0, 14,
    14, 0]
seeds = [28683758, 559688925, 878136211, 802340437, 310606466,
    740634992, 695238957, 392760566, 479279907, 96858603]
```

Table 3: Call_35_Vehicle_7

| Method | Average objective | Best objective | Improvement (%) | Running time |
|---:|---:|---:|---:|---:|
| Random search | 15924073.22 | 14436028 | 20.19% | 1.09s |
| Local Search-1-insert | 5819340.10 | 5228356 | 71.57% | 3.47s |
| Local Search-2-exchange | 5454843.20 | 4969831 | 72.97% | 3.07s |
| Local Search-3-exchange | 5640922.60 | 5204477 | 71.70% | 3.09s |
| Simulated Annealing-1-insert | 5990902.20 | 5202297 | 71.71% | 3.53s |
| Simulated Annealing-2-exchange | 5640291.20 | 5172819 | 71.87% | 3.40s |
| Simulated Annealing-3-exchange | 6230478.40 | 5429998 | 70.47% | 3.33s |

Listing 3: Optimal solution call_35_vehicle_7

```
sol = [7, 12, 12, 30, 7, 30, 33, 33, 0, 15, 15, 23, 23, 17, 29, 29, 17,
    0, 8, 8, 24, 24, 21, 21, 20, 20, 31, 31, 0, 28, 34, 28, 34, 4, 4, 6,
    6, 27, 27, 13, 13, 26, 26, 0, 14, 9, 18, 9, 14, 19, 19, 5, 18, 5, 2,
    2, 0, 22, 16, 16, 3, 3, 11, 11, 25, 22, 32, 32, 25, 0, 1, 1, 35, 35,
    10, 10, 0]
seeds = [602440671, 201885175, 752505642, 260785913, 336519552,
    134763448, 226276419, 18479076, 326119647, 458207768]
```

Table 4: Call_80_Vehicle_20

| Method | Average objective | Best objective | Improvement (%) | Running time |
|---:|---:|---:|---:|---:|
| Random search | 39584864.24 | 37697832 | 18.28% | 2.44s |
| Local Search-1-insert | 12571064.80 | 11665495 | 75.06% | 7.66s |
| Local Search-2-exchange | 13030850.90 | 12197881 | 73.92% | 7.15s |
| Local Search-3-exchange | 13565248.30 | 12393698 | 73.50% | 8.14s |
| Simulated Annealing-1-insert | 12586025.80 | 11678340 | 75.03% | 8.12s |
| Simulated Annealing-2-exchange | 12821646.40 | 11652331 | 75.09% | 6.96s |
| Simulated Annealing-3-exchange | 13451335.20 | 12609386 | 73.04% | 8.65s |

Listing 4: Optimal solution call_80_vehicle_20

```
sol = [11, 11, 73, 35, 35, 73, 80, 80, 0, 53, 32, 32, 53, 69, 69, 27,
    27, 49, 49, 5, 5, 0, 71, 71, 9, 9, 56, 56, 20, 20, 0, 4, 4, 25, 25,
    23, 23, 40, 40, 63, 63, 0, 74, 67, 74, 67, 16, 16, 0, 61, 54, 61, 54,
     0, 28, 28, 64, 64, 0, 22, 22, 34, 46, 46, 34, 0, 68, 37, 21, 68, 21,
     37, 0, 70, 15, 15, 70, 18, 18, 48, 17, 48, 17, 0, 50, 41, 59, 41,
    50, 59, 0, 7, 7, 55, 36, 55, 26, 26, 36, 0, 58, 29, 58, 29, 78, 78,
    33, 33, 0, 31, 44, 44, 31, 0, 8, 1, 8, 19, 19, 1, 0, 62, 62, 57, 57,
    10, 12, 72, 10, 12, 72, 24, 24, 6, 6, 0, 30, 66, 66, 30, 43, 43, 65,
    65, 75, 75, 0, 13, 51, 13,
       2, 2, 47, 51, 47, 42, 3, 42, 3, 0, 39, 39, 76, 76, 52, 52, 0, 79,
           14, 60, 60, 14, 38, 79, 38, 45, 77, 45, 77, 0]
seeds = [410507890, 306056627, 570963682, 962688322, 20076689, 10957088,
     341999556, 118995085, 721070358, 818664824]
```

Table 5: Call_130_Vehicle_40

| Method | Average objective | Best objective | Improvement (%) | Running time |
|---:|---:|---:|---:|---:|
| Random search | 76627567.00 | 76627567 | 0.00% | 4.52s |
| Local Search-1-insert | 22359397.90 | 19853349 | 74.09% | 15.42s |
| Local Search-2-exchange | 22483781.20 | 21396402 | 72.08% | 14.74s |
| Local Search-3-exchange | 22897432.70 | 22261446 | 70.95% | 14.28s |
| Simulated Annealing-1-insert | 22572406.80 | 19897353 | 74.03% | 13.85s |
| Simulated Annealing-2-exchange | 21844483.10 | 20984143 | 72.62% | 12.09s |
| Simulated Annealing-3-exchange | 22770710.20 | 21553305 | 71.87% | 13.09s |

Listing 5: Optimal solution call_130_vehicle_40

```
sol = [21, 91, 21, 84, 84, 91, 0, 44, 27, 27, 44, 0, 120, 120, 49, 49,
    32, 32, 112, 112, 0, 37, 106, 37, 106, 0, 114, 68, 7, 7, 114, 68, 83,
     83, 0, 23, 98, 23, 61, 61, 41, 41, 98, 0, 46, 96, 96, 55, 55, 46,
    12, 12, 0, 33, 33, 4, 4, 0, 1, 101, 40, 40, 1, 101, 0, 15, 15, 129,
    129, 0, 110, 110, 113, 69, 69, 113, 0, 88, 88, 22, 22, 0, 50, 50, 25,
     25, 92, 10, 92, 122, 35, 122, 35, 10, 0, 115, 79, 79, 115, 5, 5, 0,
    48, 17, 48, 62, 62, 17, 0, 66, 66, 107, 107, 20, 20, 0, 103, 103, 74,
     74, 118, 118, 0, 123, 53, 34, 123, 51, 53, 34, 51, 0, 39, 39, 108,
    108, 94, 94, 0, 67, 30, 67, 30, 26, 26, 0, 72, 105, 72, 105,
        16, 16, 0, 124, 73, 109, 73, 109, 124, 0, 6, 6, 100, 100, 0, 11,
            70, 11, 87, 70, 87, 0, 80, 3, 3, 80, 71, 71, 52, 52, 0, 119,
            119, 104, 104, 0, 2, 2, 19, 19, 54, 125, 54, 125, 0, 36, 121,
            42, 121, 130, 36, 42, 130, 0, 45, 60, 45, 60, 85, 85, 0, 24,
            24, 56, 56, 57, 57, 0, 111, 63, 63, 111, 78, 78, 0, 28, 28, 93,
             93, 18, 9, 18, 9, 0, 86, 86, 126, 127, 127, 126, 47, 47, 64,
            64, 0, 59, 97, 97, 82, 82, 81, 59, 81, 0, 77, 29, 90, 29, 77,
            90, 76, 76, 0, 13, 13, 116, 116, 43, 43, 0, 8, 8, 117, 117,
            128, 128, 0, 102, 95, 58, 95, 102, 58, 0, 14, 99, 14, 65, 65,
            99, 0, 75, 89, 89, 75, 38, 38, 31, 31, 0]
seeds = [492108247, 811493354, 647421449, 117681803, 946564870,
    405499103, 758788389, 645775508, 926172454, 203655223]
```

Table 6: Call_300_Vehicle_90

| Method | Average objective | Best objective | Improvement (%) | Running time |
|---:|---:|---:|---:|---:|
| Random search | 170784643.00 | 170784643 | 0.00% | 10.66s |
| Local Search-1-insert | 54685510.70 | 53203600 | 68.85% | 30.49s |
| Local Search-2-exchange | 54681910.50 | 52940080 | 69.00% | 27.82s |
| Local Search-3-exchange | 56914588.70 | 55732587 | 67.37% | 25.46s |
| Simulated Annealing-1-insert | 54686777.30 | 53279416 | 68.80% | 30.30s |
| Simulated Annealing-2-exchange | 54294805.00 | 51353808 | 69.93% | 27.62s |
| Simulated Annealing-3-exchange | 57467044.50 | 56086733 | 67.16% | 26.64s |

Listing 6: Optimal solution call_300_vehicle_90

```
sol = [32, 32, 167, 167, 0, 255, 171, 297, 171, 297, 255, 0, 278, 278,
    120, 120, 0, 63, 154, 154, 63, 0, 248, 248, 292, 292, 198, 198, 0,
    48, 79, 79, 48, 273, 273, 104, 104, 0, 155, 155, 15, 15, 275, 275, 0,
     73, 182, 182, 73, 0, 37, 158, 37, 158, 262, 262, 0, 265, 270, 270,
    265, 0, 126, 126, 140, 102, 140, 40, 102, 40, 0, 195, 36, 195, 36, 0,
     92, 92, 106, 17, 17, 106, 0, 213, 219, 213, 219, 113, 113, 0, 272,
    272, 300, 300, 0, 294, 294, 224, 224, 281, 281, 60, 60, 0, 41, 70,
    41, 62, 62, 70, 0, 28, 28, 172, 172, 0, 122, 131, 131, 122, 0, 98,
    271, 271, 23, 23, 98, 0, 136, 240, 136, 55, 194, 194, 55, 240, 0,
    130, 130, 21, 227, 227, 21, 0, 35, 65, 65, 35, 5, 5,
        0, 7, 7, 59, 59, 258, 258, 226, 226, 0, 118, 50, 118, 291, 50,
        291, 0, 25, 25, 10, 121, 121, 10, 110, 110, 0, 264, 212, 264,
        288, 212, 288, 0, 14, 14, 127, 127, 11, 11, 0, 280, 280, 69,
        69, 0, 236, 236, 165, 165, 139, 139, 129, 129, 0, 180, 179,
        179, 180, 235, 235, 0, 168, 168, 211, 211, 0, 61, 84, 84, 205,
        61, 205, 0, 163, 228, 187, 163, 187, 228, 0, 210, 72, 210, 72,
        0, 157, 208, 157, 208, 243, 243, 132, 132, 56, 56, 0, 282, 282,
         221, 221, 20, 20, 0, 209, 209, 222, 123, 123, 222, 141, 250,
        250, 141, 0, 241, 83, 241, 83, 138, 138, 279, 279, 0, 185, 68,
        245, 185, 245, 68, 147, 147, 0, 218, 218, 119, 119, 30, 30, 0,
        285, 285, 24, 24, 200, 200, 193, 193, 0, 215, 217, 215, 217, 0,
         26, 26, 149, 149, 100, 100, 0, 166, 166, 170, 170, 0, 191,
        151, 191, 266, 151, 266, 177, 177, 0, 77, 77, 47, 47, 259, 259,
         0, 244, 244, 237, 237, 229, 229, 0, 144, 114, 144, 125, 114,
        125, 0, 159, 159, 150, 150, 27, 27, 0, 107, 107, 19, 19, 145,
        145, 0, 293, 293, 142, 142, 296, 296, 0, 286, 124, 82, 82, 286,
         124, 0, 184, 181, 184, 99, 99, 115, 115, 181, 0, 232, 81, 81,
        97, 232, 97, 0, 263, 220, 12, 12, 268, 220, 263, 268, 0, 31,
        31, 117, 117, 0, 298, 298, 22, 203, 203, 22, 267, 267, 156,
        156, 0, 277, 152, 201, 152, 201, 277, 0, 101, 295, 109, 53,
        109, 53, 295, 101, 0, 249, 249, 33, 33, 0, 276, 276, 153, 199,
        199, 153, 0, 44, 45, 44, 45, 6, 6, 85, 85, 0, 39, 39, 52, 52,
        13, 13, 0, 29, 18, 29, 94, 18, 94, 0, 133, 133, 87, 225, 87,
        216, 225, 216, 0, 4, 4, 137, 137, 256, 256, 0, 134, 134, 238,
        238, 251, 251, 192, 192, 161, 161, 0, 8, 8, 112, 112, 80, 80,
        0, 190, 190, 128, 91, 91, 128, 0, 290, 58, 290, 231, 58, 231,
        0, 253, 96, 96, 253, 178, 178, 0, 2, 284, 2, 64, 284, 64, 0,
        186, 260, 207, 207, 260, 197, 197, 186, 0, 108, 108, 90, 160,
        90, 160, 0, 188, 188, 16, 74, 74, 16, 0, 3, 3, 202, 254, 202,
        254, 0, 88, 34, 34, 176, 176, 88, 0, 239, 239, 49, 49, 0, 246,
        9, 257, 9, 173, 246, 257, 175, 175, 173, 0, 86, 86, 111, 46,
        51, 46, 51, 111, 0, 89, 1, 196, 196, 1, 234, 234, 89, 93, 93,
        0, 143, 42, 143, 189, 148, 189, 42, 148, 0, 71, 261, 261, 71,
        0, 116, 116, 76, 76, 0, 230, 230, 183, 183, 287, 287, 0, 269,
        214, 269, 38, 214, 38, 0, 233, 233, 164, 204, 164, 204, 0, 206,
         67, 67, 206, 103, 103, 75, 283, 283, 75, 0, 289, 95, 95, 289,
        252, 252, 146, 146, 0, 43, 43, 54, 54, 57, 57, 66, 66, 78, 78,
        105, 105, 135, 135, 162, 162, 169, 169, 174, 174, 223, 223,
        242, 242, 247, 247, 274, 274, 299, 299]
seeds = [3523968, 263920058, 903828883, 329163709, 138373585, 118410188,
   168608149, 449800115, 71276957, 929187971]
```