OTML Project (Codes)

Instructions to run the codes

There were three problems in this project, ZDT1, ZDT2 and Antenna.
The codes are implemented in MATLAB.

ZDT1

Filename: ZDT1.mlx

Steps:

1. Open the file ZDT1.mlx in MATLAB
2. Clicking on the “Run” button or “Run to end” button will generate a graph showing the Pareto front of the problem evaluated for 100 points.
3. Since the output of the graph is a convex figure and the points are equi-spaced, the implementation is successfully validated.
4. Altenatively use the file `ZDTCodeOutput1.pdf` for results.

ZDT2

Filename: ZDT2.mlx

Steps:

1. Open the file ZDT2.mlx in MATLAB
2. Clicking on the “Run” button or “Run to end” button will generate a graph showing the Pareto front of the problem evaluated for 100 points.
3. Since the output of the graph is a concave figure and the points are equi-spaced, the implementation is successfully validated.
4. Altenatively use the file `ZDTCodeOutput2.pdf` for results.

Antenna

Filename: Final.mlx

Steps:

1. In the `Antenna Properties` section:
    - adjust the value of `fv` to the desired value of the frequency of the antenna in GHz.
    - adjust the value of `er` to the desired value of the dielectric constant
2. In the `Set the bounds for the antenna` section:
    - Update the variables `lmin`, `lmax` to the lower bounds of the length of the patch in millimeters.
    - Update the variables `wmin`, `wmax` to the lower bounds of the width of the patch in millimeters.
    - Update the variables `hmin`, `hmax` to the lower bounds of the height of the patch in millimeters.
3. Now click the “Run” button to start solving the problem and get the solution.
4. At the bottom of the outputs, the optimal dimensions of the antenna including the error (f1) can be obtained.
5. A graph showing the results of both optimization functions can also be seen.
6. Altenatively use the file `AntennaCodeOutput1.pdf` for results.
7. To alter the solutions, the values of `z` and `w` can also be adjusted in the `Generator` section.

Troubleshooting

If the algorithm doesn’t print anything, it is unable to find a feasible `guess` solution to the problems inorder to satisfy the constraints. Updating the `z` and `w` values could solve this issue.

If the algorithm repeats iterations, it is rejecting solutions which are `converging to an infeasible point`. Updating the `z` and `w` values could solve

this issue.