Assignment 8: Global convexity (fitness-distance/similarity correlations) tests

Mateusz Tabaszewski 151945 Bartłomiej Pukacki 151942

Source code:
https://github.com/MatTheTab/Evolutionary-Computation/blob/main/Assignment_8_Global_Convexity/Assignment_8_Global_Convexity.ipynb

1. Description of the Problem

This report uses global convexity (fitness-distance/similarity correlations) tests to analyze the problem, previously solved in the earlier reports. Unlike the previous experiments, however, the ones described here focus on the analysis of the problem itself rather than the use of a new algorithm to improve the score or computation time. The problem is as described previously, a redefined TSP. To solve the task 50% of all available nodes need to be selected with the objective function to be minimized being the sum of the cycle created from choosing those nodes plus the sum of weights of the nodes involved in creating the cycle. The following figures show the two available instances of the problem TSPA and TSPB.

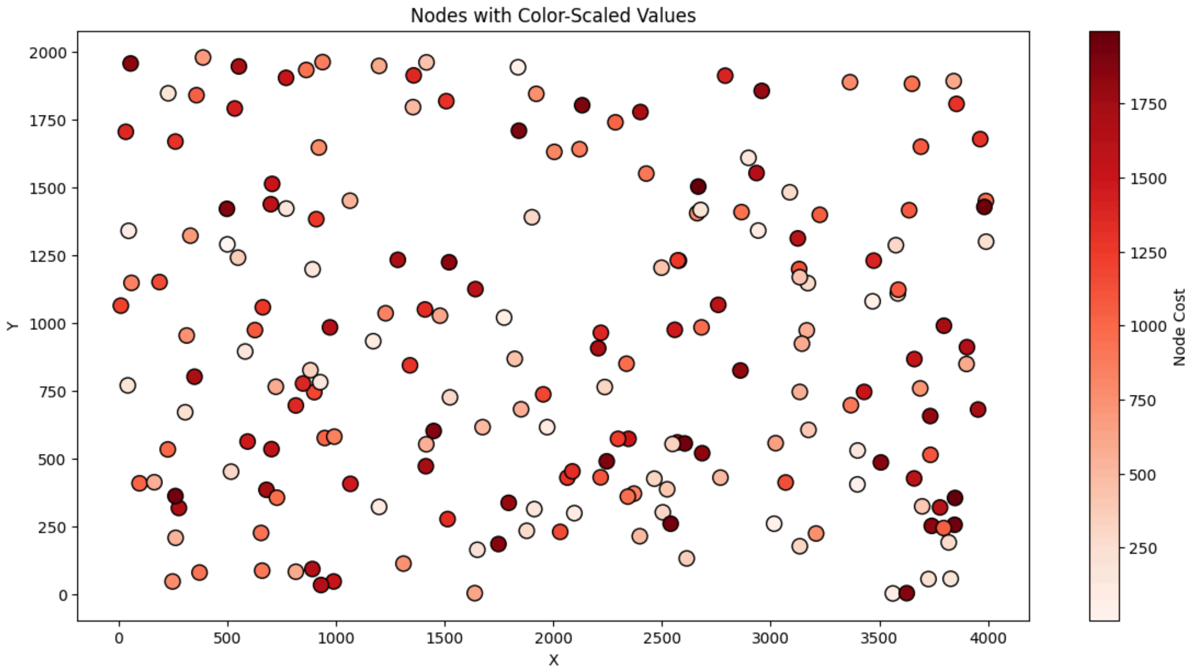

Fig 1. Visualization of the TSPA problem instance, each node's x and y locations on the plot correspond to their given x and y locations and the color intensity signifies the weight/cost of each node. The total length of the cycle and the sum of node weights should be minimized.

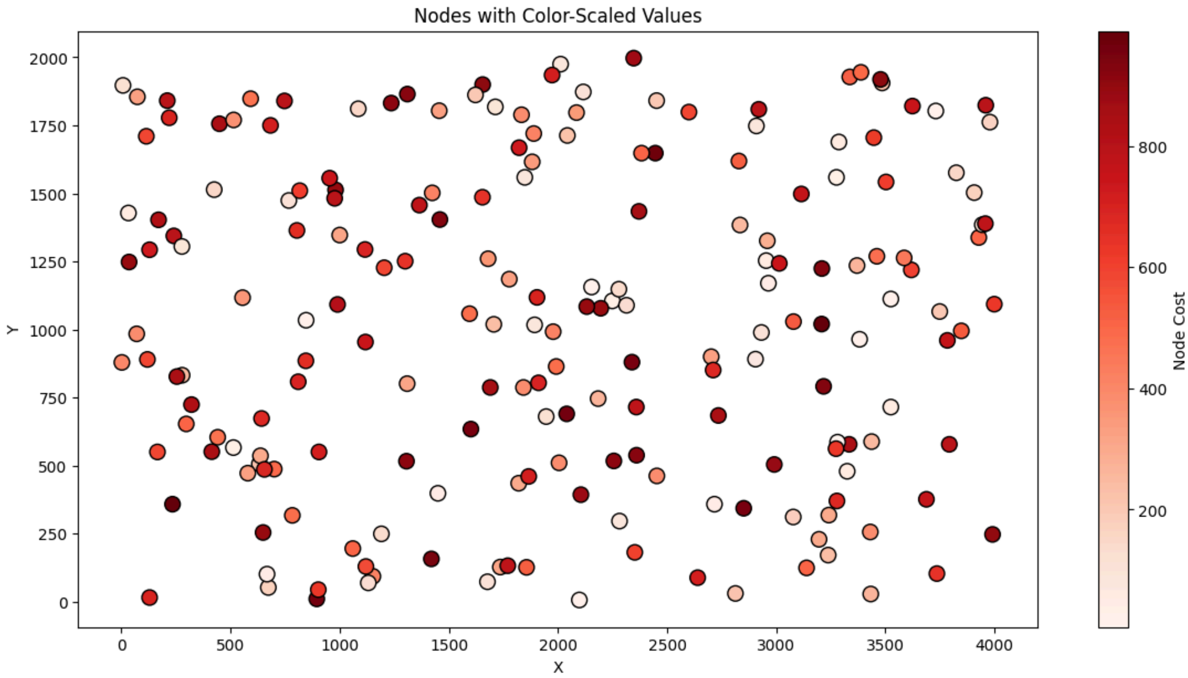

Fig 2. Visualization of the TSPB problem instance, each node's x and y locations on the plot correspond to their given x and y locations and the color intensity signifies the weight/cost of each node. The total length of the cycle and the sum of node weights should be minimized.

2. Experiments

Through the conducted experiments we sought to discover the relations between different similarity measures and the obtained scores on both instances (TSPA, TSPB). First, a very good solution was created by taking the best solution from five runs of **Large Neighborhood Search**, and then 1000 experiments were run with the use of the **Greedy Local Search Algorithm**. Afterward, the similarity scores with regard to nodes and edges, from now on referred to as **node similarity** and **edge similarity** were calculated between all of the obtained solutions. The edge similarity defines edges as going both ways, so 1 to 2 is the same as 2 to 1. Finally, the mean similarity between the target and all nodes was calculated, as well as the similarity between a particular example and the best of the obtained solutions (most likely the **Large Neighborhood Search**'s solution). However, when calculating the similarity between the current solution and the best solution, the best solution was skipped to not calculate the similarity between the solution and itself so as to not produce an outlier with 100% similarity to itself. The whole aforementioned process was repeated for both types of similarities for both instances, resulting in 8 total charts. For each chart, a correlation between similarity and the obtained score was calculated.

```
FUNCTION Run_Experiments(distance_matrix, weights):
    INPUT:
        distance_matrix - matrix of distances between nodes
        weights - an array of weights associated with each node

    best_solution, best_score ← BEST OF(RUN Large_Neighborhood(distance_matrix,
                                                               weights) 5 times
    solutions, scores ← RUN greedy_local_search_edges(distance_matrix, weights)
                                                                    1000 Times
    node_similarities ← get_node_similarities(best_solution + solutions)
    edge_similarities ← get_edge_similarities(best_solution + solutions)
    scores += best_score

    mean_similiarities_node ← MEAN OF node_similarities FOR solution in
                                                    node_similarities
    mean_similiarities_edge ← MEAN OF edge_similarities FOR solution in
                                                    edge_similarities
    best_similiarities_node ← GET similarity between best_solution in node_similarities
AND EVERY OTHER solution in node_similarities
    best_similiarities_edge ← GET similarity between best_solution in edge_similarities
AND EVERY OTHER solution in edge_similarities

    mean_corr_node ← get_correlation(mean_similiarities_node, scores)
    mean_corr_edge ← get_correlation(mean_similiarities_node, scores)
    best_corr_node ← get_correlation(mean_similiarities_node, scores)
    best_corr_edge ← get_correlation(mean_similiarities_node, scores)

    plot_results(mean_similiarities_node, scores, mean_corr_node)
    plot_results(best_similiarities_node, scores, best_corr_node)
    plot_results(mean_similiarities_edge, scores, mean_corr_edge)
    plot_results(best_similiarities_edge, scores, best_corr_edge)

    RETURN best_solution, best_score, num_iterations
```

```
FUNCTION get_node_similarities(solutions):
    INPUT:
        solutions - solutions to be considered

    similarities ← EMPTY
    FOR solution_1 in solutions:
         FOR solution_2 in solutions:
            set_1 ← set(nodes in solution_1)
            set_2 ← set(nodes in solution_2)
            intersection_set ← INTERSECTION OF set_1 AND set_2
            similarity ← (LENGTH OF intersection_set)/(LENGTH OF set_1)
            similarities.add(similarity)
    RETURN similarities
```

```
FUNCTION get_edge_similarities(solutions):
    INPUT:
        solutions - solutions to be considered

    similarities ← EMPTY
    FOR solution_1 in solutions:
         FOR solution_2 in solutions:
             set_1 ← set(edges in solution_1)
             set_2 ← set(edges in solution_2)
             intersection_set ← INTERSECTION OF set_1 AND set_2
             similarity ← (LENGTH OF intersection_set)/(LENGTH OF set_1)
             similarities.add(similarity)
    RETURN similarities
```

3. Results

3.1. Best Solutions

The **best scores achieved** are visualized below, to them other results will be compared.

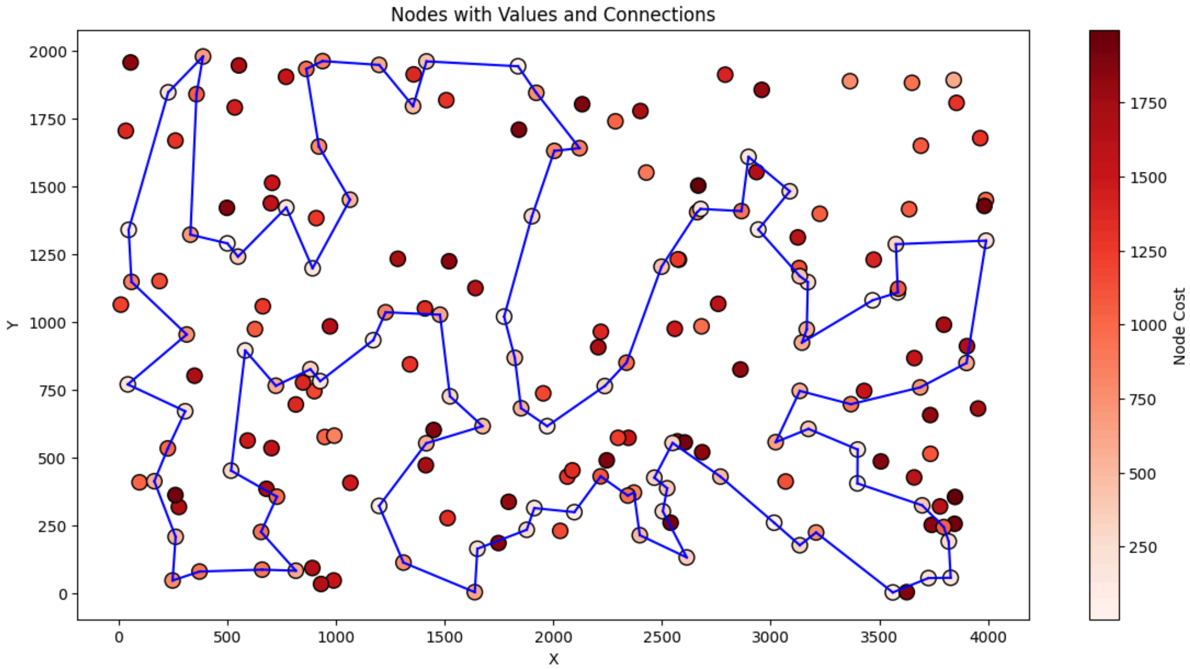

Fig 3. Visualization of the best solution found by the **Large Neighborhood Search** on the TSPA problem instance, with the score of 69207.

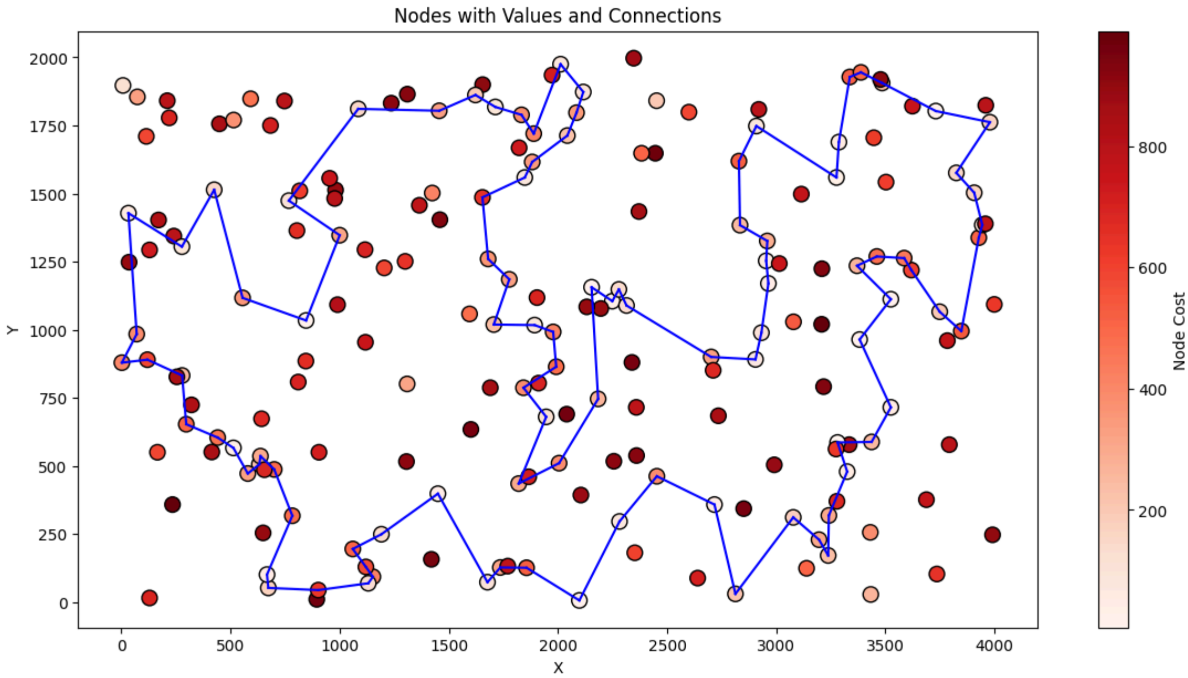

Fig 4. Visualization of the best solution found by the **Large Neighborhood Search** on the TSPB problem instance with the score of 43873.

The Further results are the showcase the visualizations of the mean and best similarities for both instances.

3.2. Similarity - TSPA

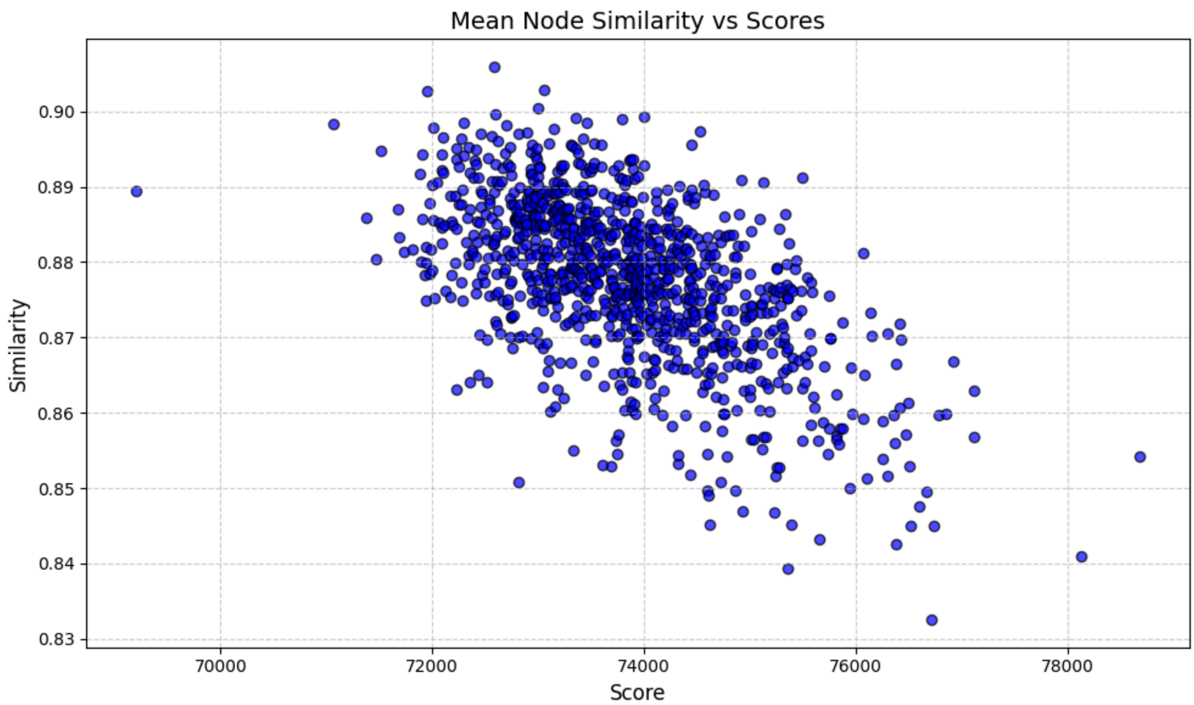

Fig 5. The mean node similarity compared against each solution’s score for the TSPA instance.

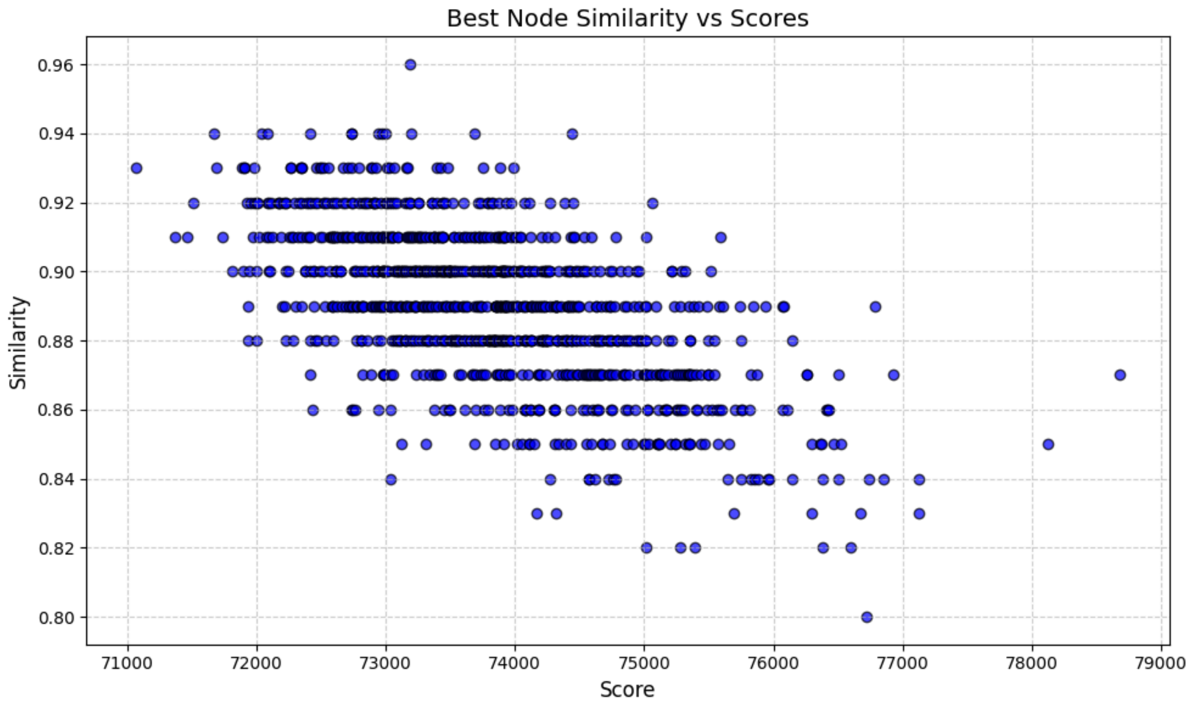

Fig 6. The node similarity compared against each solution's score for every solution and the best solution for the TSPA instance.

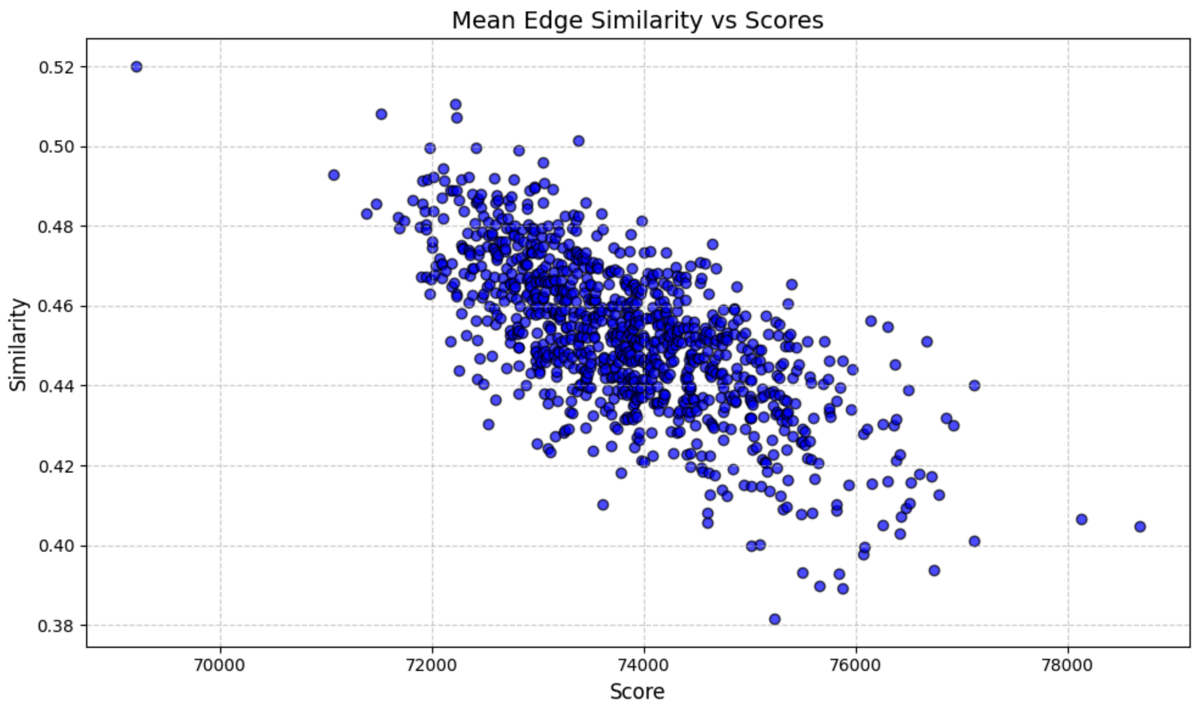

Fig 7. The mean edge similarity compared against each solution's score for the TSPA instance.

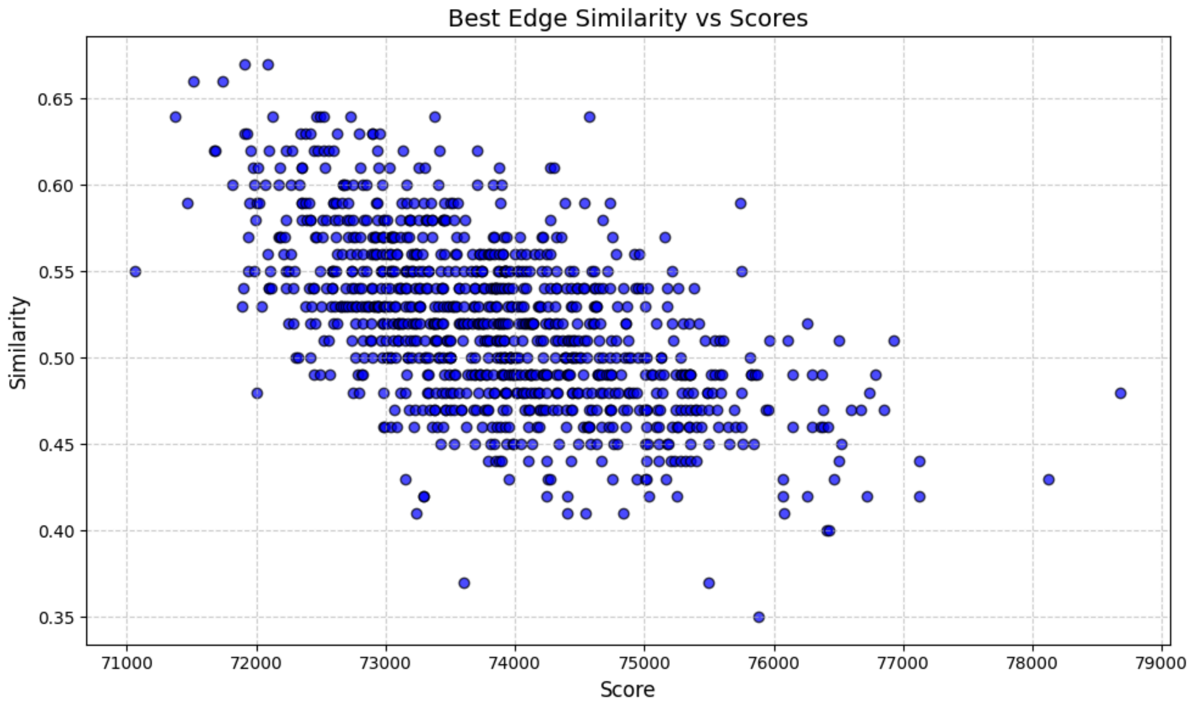

Fig 8. The edge similarity compared against each solution's score for every solution and the best solution for the TSPA instance.

3.3. Similarity - TSPB

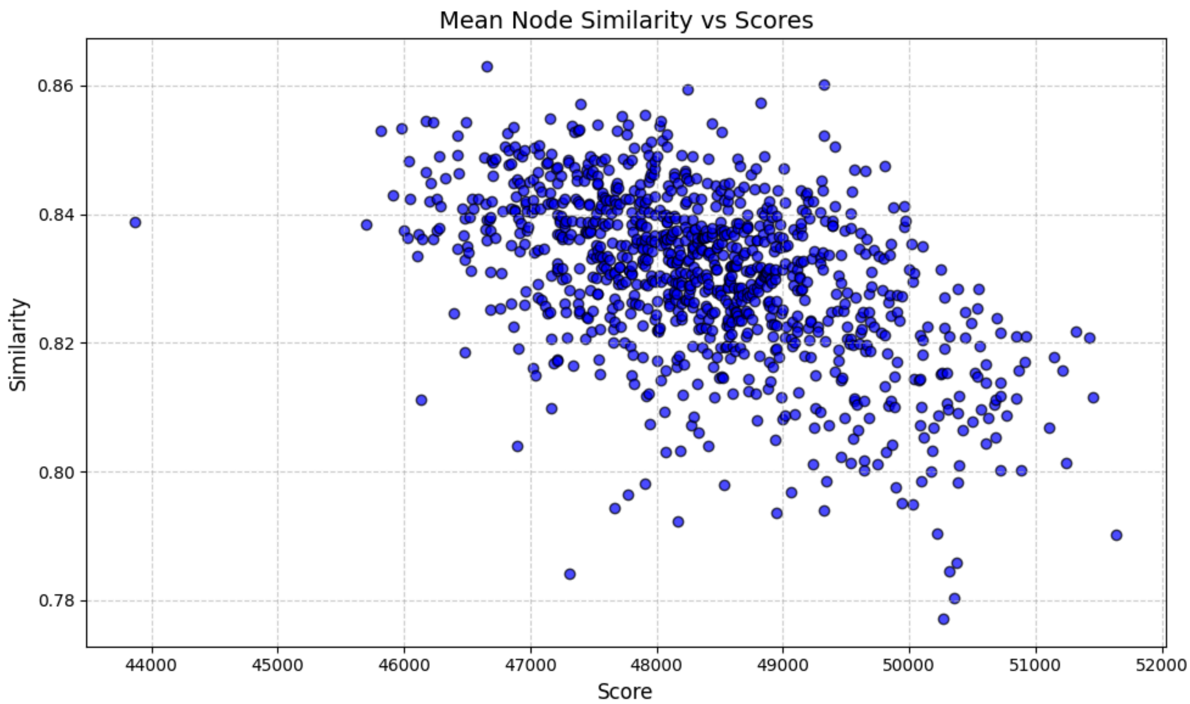

Fig 9. The mean node similarity compared against each solution's score for the TSPB instance.

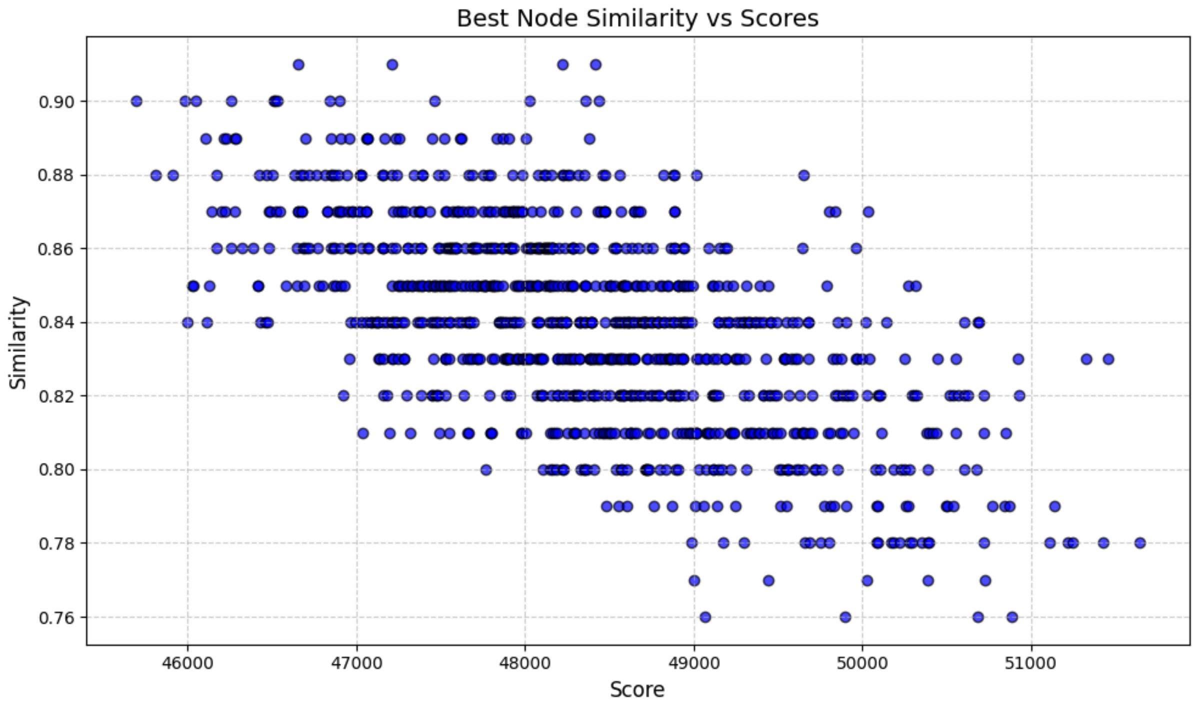

Fig 10. The node similarity is compared against each solution's score for every solution and the best solution for the TSPB instance.

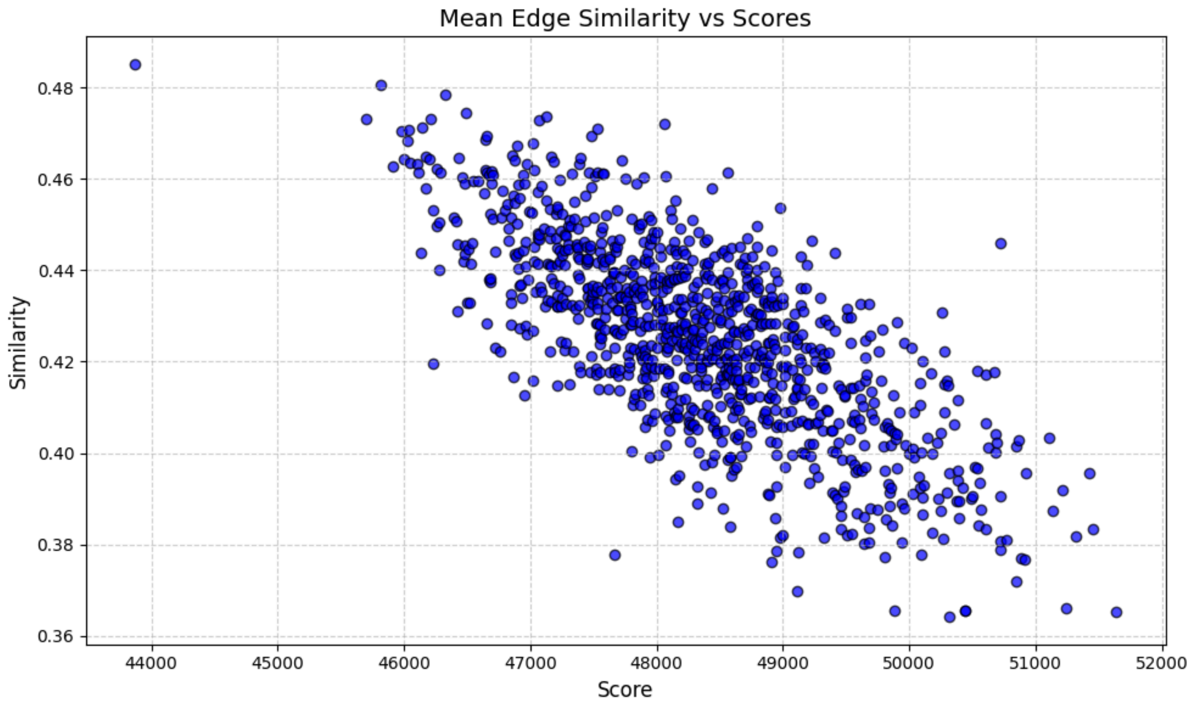

Fig 11. The mean edge similarity compared against each solution's score for the TSPB instance.

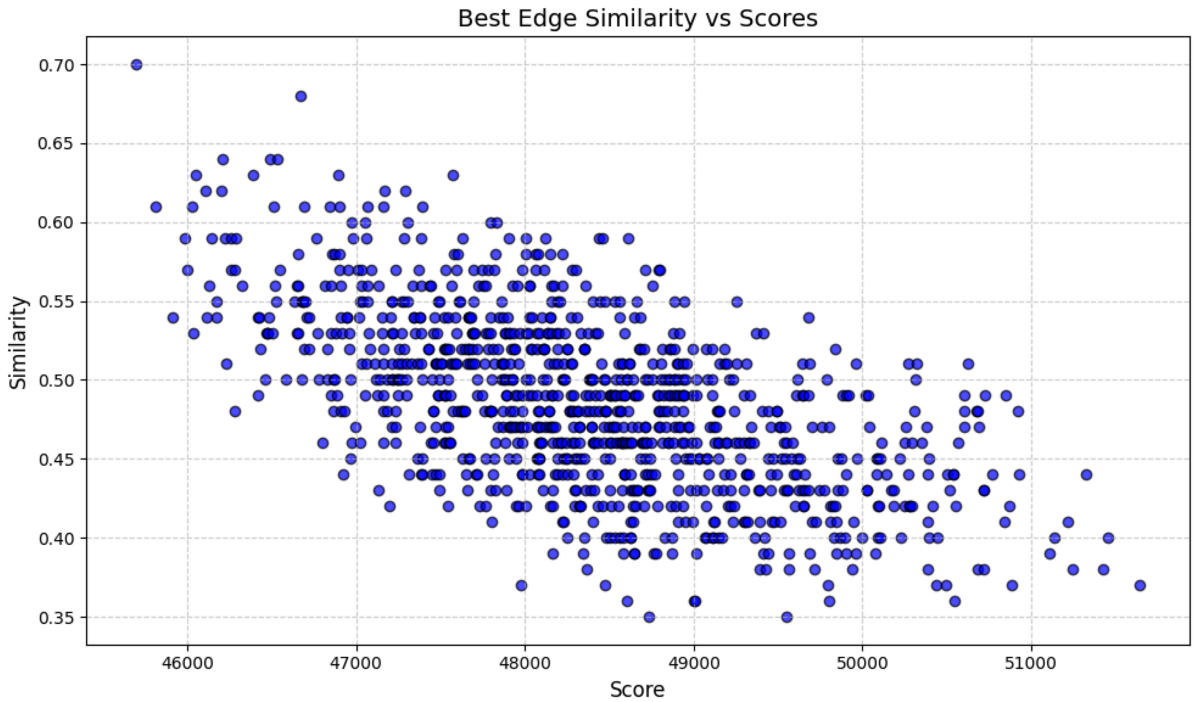

Fig 12. The edge similarity compared against each solution's score for every solution and the best solution for the TSPB instance.

Problem instance	Mean Node Correlation	Best Node Correlation	Mean Edge Correlation	Best Edge Correlation
TSPA	-0.585	-0.624	-0.694	-0.602
TSPB	-0.511	-0.620	-0.729	-0.602

Table 1. Correlation Values Comparison for each instance depending on the measured similarity metric.

4. Conclusions

The conducted experiments revealed that for both instances the lower the score (the better the solution), the higher the similarity is to the best discovered solution. Although a lot of variability can be seen, especially with the solutions around the same score value but with different similarities, despite that, the trend is clearly visible. This means that the solutions that do not differ too much from the best solution are more likely to have a better score, which makes sense intuitively. In a sense, one can say that small, local changes are unlikely to have a significant impact on the overall quality of the solution when it comes to the presented redefined TSP challenge. Furthermore, the best node and edge similarities values are more discreet than the mean similarities, which makes sense since comparing against a single solution introduces the smallest perceivable level of granularity - a single node/edge, this is not the case when it comes to calculating the mean between many solutions. Furthermore, the mean node and edge similarities are also higher for lower (better) scores for both instances. This is likely because all of the solutions manage to find the same or similar locally optimal changes, like some very good edges/nodes that should often be included in a good solution. If that is the case, then the best solutions would likely include many such locally optimal changes which would, in turn, increase the similarity between it and other solutions. This might be a good sign for the potential future use of evolutionary algorithms since those could leverage the whole population's collective "knowledge" to create better solutions. Furthermore, it can be seen that the edge similarities are lower than node similarities, this makes sense since for edge similarities both nodes and their neighbors matter. In other words, two solutions may have 1.0 node similarity and less than 1.0 edge similarity, due to having the exact same nodes but completely different edges because of the completely different order of those nodes with respect to each other, in essence, making edge similarity almost always less than the node similarity. In conclusion, there seems to be a negative correlation between the obtained similarity scores and the objective function scores. This observation is validated by the results of the correlation calculations visible in the presented table (Table 1).