2019-2020

Business Analytics with Soft Computing

CMSE11429

B149183

MSc Business Analytics

March 5, 2020

Contents

List of Figures

List of Tables

1 Executive Summary

This report draws attention to the Single Allocation p-Hub Location Problem (SApHLP) and the application of meta-heuristic algorithms. In this problem, the main goal is to select a number of nodes to be set as hubs. Using these hubs will allow the network to make use of the economies of scale principle for transportation logistics. However, using an analytical approach is at most times infeasible to solve due to the size of the solution space. In that case, heuristics are a favored approach. They are able to navigate the solution space efficiently making use of natural characteristics of the problem.

In the following sections we discuss the implementation of two algorithms used, both of which come under the meta-heuristics branch of computing. The first algorithm used is Tabu Search (TS) in which a neighborhood structure is defined and the solution space is traversed starting from an initial random solution. This approach delivered good results for a small number of nodes with almost zero optimality gap however it faced some issues for larger numbers of nodes. The second algorithm used is the Genetic Algorithm (GA). It is a population based approach based on the biological principle of natural selection. In this approach it was easier to converge to the optimal solution on a small number of nodes. In addition to that, the algorithm kept its performance for larger data sets. An interesting aspect to note about the GA approach is that it keeps an almost constant runtime. Increasing the number of nodes does not greatly affect time of convergence.

Since speed and computing efficiency are a main concern in this problem, the algorithms were implemented using Golang, a CPU efficient programming language developed by Google.

2 Introduction

The hub location problem is considered to be a classical problem that has been extensively studied. In a network of connected locations (referred to as nodes), the main purpose is to identify strategic nodes which would act as hubs. Hubs would be a point of contact between other nodes; moreover, they are also considered as processing centers. This approach makes use of the concept of economies of scale where the additional costs of setting up hubs could potentially be less than the cost incurred to operate direct routes between every node. In economies of scale, a discount factor is considered to calculate the overall cost of the network. This factor is usually selected in the range of 0 to 1 and will measure a trade off of having more or less optimized hub interconnections.

There are many variants to the p-hub allocation problem. These variants depend on whether the problem is a multiple or single assignment and whether it is capacitated or uncapacitated. Regarding assignment, in the former problem a node can be assigned to multiple hubs whereas in the latter it is assigned to strictly one hub in which all of its flow must go through. In the case of capacity, some problems might have a capacity limit on hub-hub flows. The application of this problem spans multiple industries where scaling is inevitable such as airline [1] and public transportation [2] services.

In this report, two meta-heuristic algorithms are explored and implemented in an attempt to find the best solution for the CAB dataset and the TR dataset. The size of the datasets varies from the 10 to 25 nodes for the CAB dataset and between 25 and 55 nodes for the TR dataset. We consider the implementation of TS and GA and compare their performance in terms of optimality and computing efficiency.

3 Literature Review

The hub location problem comes down to selecting a set of nodes to be considered as hubs then assigning nodes to respective hubs. In [3], after selecting the hubs, the authors consider various techniques of node to hub allocations ranging from distance only based on multiple criteria including a weighted sum of common traffic between a hub and node. As for selecting the hubs, they start with an initial random solution and then evaluate an improvement score for each non-hub node that could potentially replace a current hub. As described by them, their double exchange algorithm performed well even for larger data sets with an 11 second convergence time.

Other papers discuss more generalized and known algorithms for solving the hub allocation problem. In [4], the authors discuss their implementation of a variant of the TS algorithm for incomplete networks. They start of from a partial and incomplete solution and traverse the solution space in search of nodes that would provide a feasible solution. The algorithm is divided into a construction phase and an improvement phase. As described, the algorithm first seeks to perturb the selected hubs of the solution, then construct allocations and finally reduce hub connectivity. Maintaining a tabu list of moves avoids cycling through previously visited solutions. The paper showed interesting results as the authors compared their heuristic method to the analytical method solved in CPLEX. they concluded that their algorithm was able to achieve an efficient solution using less CPU time than the CPLEX method.

In [5], the authors tackle the problem for a less-than-truckload services company in Brazil. To select the hubs and assign the nodes, the authors adopt a GA approach. They include a varying cost instead of a constant discount factor that depends on the total amount of freight between hub terminals. Also, they modify the GA algorithm to incorporate a local search heuristic on each individual of the population. The results obtained were at optimality for most the datasets they tested however they did see a small optimality gap for a higher number of nodes. Also, their runtime could not be compared to current numbers as they were using a Pentium IV 1.7 GHz machine. Given today's computing power improvements, their runtime would be significantly lower.

4 Model Description

4.1 Overview

Both algorithms discussed in this report rely on an initial starting point on which they intend to maneuver in search of a better solution. In the discussed implementations, generating the initial solution involves randomly selecting a specified number of hubs from the nodes and allocating nodes to the hubs based on the shortest distance. In the TS approach, the solution space is explored through neighborhood structures of the current best solution found. However, in the GA approach, we start with a population of initial solutions.

4.2 Tabu Search

TS involves exploring the neighborhood space of a solution and evaluating whether a better solution than the current best could be achieved. Multiple neighborhood structures were considered in this implementation and the results varied widely depending on the chosen approach. TS also maintains a tabu list which prevents revisiting recently visited nodes. In this strategy, a form of short term term memory is utilized to make the algorithm more efficient in CPU time usage.

In general, the TS algorithm takes an initial solution which is considered the best one found so far. Then on each iteration it explores its neighborhood via the defined neighborhood structure. To explore this space, a number of neighbor solutions are generated starting from the best current solution. The pool of new candidate solutions is then observed and the best candidate is used as the starting point for the next iteration, given that it is not in the tabu list or the aspiration condition has been met. If the candidate provides a lower cost, it becomes our new best solution. On the other hand, if the best candidate is in the tabu list and we have not reached aspiration, the next best neighbor that meets the mentioned criteria will be selected for the next iteration.

4.2.1 Neighborhood Structures

- Structure A:

  In this structure, a neighboring solution is achieved by swapping a random node with its hub and then reallocating the nodes of the previous hub to the new selected hub.

- Structure B:

  In this structure, a more basic approach is used where two non-hub nodes are randomly selected and their hub allocations are switched.

- Structure C:

  In this structure, we select a random node and assign it to a random hub.

4.2.2 Stopping Criteria

In order to set the termination condition of the algorithm, it is considered that the algorithm converges if no better solution has been found in a certain number of iterations.

4.2.3 Tabu List Entries

The tabu list keeps track of the node swaps that resulted in a better solution. In this strategy, this node will be static for a number of iterations depending on the size of the tabu list. By giving resilience to a good move from future random changes, we are able to explore the possibility that this node is indeed in its optimal position and try to improve the overall solution by exploring other moves.

4.2.4 Parameters

The parameters of the TS algorithm include the number of iterations to run and the number of candidate neighbor solutions to generate at each iteration of the algorithm, starting from the best solution found so far. The number of iterations is set high enough to explore many solutions' neighborhoods, in this report the number of iterations was initially set to 100000 and the number of candidates to generate was set to 60. The high number of candidates generated each iteration will minimize the possibility of not exploring a possible neighbor due to the randomness of the solution space maneuvers. Later, the number of candidates was made to be a function of the number of nodes.

The tabu list size is chosen to be a function of the size of the dataset. In cases where the dataset is small, a large tabu list size would eventually block all possible moves while generating a neighboring solution, while a rather small list for a large number of nodes would reduce the benefits of making a good move static for a number of iterations. Therefore, it was set to be 20% of the size of the dataset.

4.3 Genetic Algorithm

The GA is a population based meta-heuristic technique which is based on the biological process of natural selection. In this algorithm, we start off with a population of random initial solutions and seek to simulate the process of reproducing generations that inherit the best fit solutions and disregard the less fit ones. In general, the algorithm goes through the following steps which are explained later on in this report:

- Generate an initial population
- Build the breeding pool
- Reproduce next generation

    Perform crossover

    Perform mutation

4.3.1 The Organism

In GA, the solution is treated as an organism with a DNA. In the p-hub allocation problem, the DNA is the set of selected hubs and the node-hub allocations. In order to measure fitness in this minimization problem, we consider the inverse of the normalized total cost of the solution.

4.3.2 Natural Selection

To simulate the process of natural selection, on each iteration a new breeding pool is generated from the current population. The breeding pool would favor the more fit organisms in the random selection process for crossovers. In detail, the breeding pool is constructed by creating a population where members are replicated with respect to their fitness level scaled by a factor 100.

$$replications = 100 * \frac{fitness}{maxFitness}$$

where *fitness* is that of the organism and *maxFitness* is the maximum fitness in the current population.

4.3.3 Crossover

The crossover happens between two randomly selected organisms from the breeding pool. In the implementation for this report, the crossover is performed by splitting the hub array at a random location and the resulting child would take the respective portions from each parent. After that, the nodes are reallocated according to the nearest hub approach. However, in the case where a child's solution is infeasabile, for example when the same node is selected as a hub more than once, the child is replaced by a new random organism generated using the same strategy as for an initial solution.

4.3.4 Mutation

In order to avoid being stuck at a local optimum, the mutation rate allows the algorithm to jump between different regions of the solution in a non-deterministic fashion. During the mutation phase, a random number is generated for each node and compared to the defined rate. If the generated number is less, the organism's node-hub allocation is randomly changed for the current node.

4.3.5 Parameters

The GA algorithm takes in three parameters which are the population size, mutation rate and the number of generations to produce. In this report the parameters were initially set to a population size of 5000, a mutation rate of 5% and 100 generations of off-springs. However, since the process of natural selection depends on generating better off-springs, the parameters were also tested with a small population size but a higher number of generations.

5 Computational Results and Analysis

5.1 Tabu Search

The results of TS in appendix A show interesting observations across different configurations. Starting with a high iteration parameter (100,000), a high number of candidates (60) and a small tabu list size (5) we find that the average optimality gap of the best of 10 runs is at 4.40% for the CAB dataset and at 5.90% for the TR dataset. Also the optimality gap across all 10 runs averages to around 15% on both datasets. However, the runtime of this configuration was in the order of minutes for the 10 runs of each configuration (number of nodes, number of hubs and discount factor). Starting from 8 seconds per run on the small 10 node CAB dataset and up to 30 seconds for the 25 nodes version of this dataset, a 3.75 times increase in time respective to doubling the number of nodes. This shows that this algorithm is running in almost quadratic time $O(n^2)$.

In an attempt to improve the results of TS, the tabu list size and candidates generated at each iteration were set to be dynamic; a function of the number of nodes. The tabu list size was set to be one fifth (1/5) of the number of nodes while the number of candidates was set to 5 times the number of nodes. The results showed an improvement of the best solution found to an optimality gap of 3.75% for the CAB dataset and 4.0% for the TR dataset with no major change in the average gap across all runs.

Moreover, it was observed that the algorithm was converging rather fast, withing 12 iterations for the largest dataset of 55 nodes in the TR dataset. Therefore the number of iterations was reduced to 100. This allows the algorithm to run much faster at 58ms per run for the largest dataset.

Other neighborhood structures were also tested with TS; however, they were dropped due to lower performance with respect to the structure used.

TS is a meta-heuristic method that improves the local search. This fact makes TS dependent on the initial solution it starts from. We notice from the results that for a low number of nodes, the algorithm was performing well and very close to optimality. However, when the number of nodes increases, we see that the optimality gap increases. This is explained by the fact that the solution space is much larger and there is a higher probability for the algorithm to find a local optimum and get stuck in it.

In summary, TS shows a trade-off between fast convergence and the optimality gap achieved.

5.2 Genetic Algorithm

The GA results as shown in appendix B, show promising results in terms of runtime. Initially, it is observed that the runtime was almost constant and irrespective of the number of nodes in the dataset. Except for the 55 nodes TR dataset, the runtime of the algorithm was around 13 seconds per run. This is an important feature as the algorithm would be a good scalable solution. In addition to that, the best run and the average optimality gap of all 10 runs showed promising results with average best run optimality gaps of 0.86% and 1.96% for the CAB and TR datasets respectively and 2.0% average across 10 runs for both datasets.

The GA is population based and therefore it was reasonable to test its performance with different population sizes. Starting with a population size of 5000 and for 100 generations, the results showed good performance across all number of nodes. The optimality gap did indeed increase with the numbers of nodes, however it was not a significant increase.

On further runs of the algorithm, it was noticed that the optimal solution was being found within less than 100 generations. It was therefore intuitive to test other parameter configurations. The parameters were changed to a smaller population size of 100 and run for 1000 generations with an aspiration condition of 100 generations of no improvement. This is to rule out the fact that better solutions could be found with more generations as this is the principle of natural selection. The runtime with 5000 organisms of population size did not allow testing for 1000 generations. In this configuration, we notice a major improvement in performance. The average optimality gap of the best run was reduced to 0.90% for the CAB dataset and to 1.22% for the TR dataset. However, what is more notable is the increased efficiency in CPU time. A single run was reduced from an average of 11 seconds to an average of 200 milliseconds and a maximum of 790 milliseconds for the largest number of nodes (55-TR), a 98% reduction on average. Given the improvement, the algorithm was tested with the same configuration but with a higher terminating condition of 500 which did not show any additional benefit, therefore it was kept at 100. Furthermore, changing the population size from 100 to 200 then to 300 showed significant improvement. Under a population size of 200 the best run optimality gap decreased to the order of 0.50% and the average optimality gap of all runs decreased to the order of 3.0% for both the CAB and TR datasets. A population size of 300 further reduced the average optimality gap of all 10 runs to the order of 2.0% while keeping the runtime under 1 second for all number of nodes except for the 55 nodes TR dataset which was at 2 seconds per run. This increase in runtime is not an issue since it was less than double after we doubled the number of nodes. This shows a sub linear runtime of the algorithm with respect to the number nodes.

In summary, the GA performed very well with a small population size and a higher number of generations. The best configuration used for this report was found to be at a population size of 300 and for 200 generations. The algorithm ran in reasonable time and provided the best optimality gaps across runs, datasets and node configurations.

5.3 Comparison

TS and GA provide two different approaches that attempt to determine the optimums of a large solution space. The end goal of both is to find the best solution possible based on a defined metric, the total network cost in this case. The GA performed better against the TS approach on all metrics considered. Looking at the average optimality gap for the best solution found across 10 runs of each algorithm, it is clear that GA has almost an 8 times better percentage gap with 0.50% vs 4.0%. Moreover, the runtime complexity of TS is almost in quadratic time were runtime for 20 nodes increased by almost 3 times from that of 10 nodes. On the other hand, GA had a slight increase in runtime when the number of nodes was doubled and could be almost considered to run in constant time $O(1)$ with respect to the number of nodes. In GA the population size had a much higher effect on runtime than the number of nodes. This makes the number of nodes in the problem almost negligible as can be seen in the time difference per run between a population size of 5000 and population size of 100, both for 100 generations. This allows us to run the algorithm for a higher number of generations without facing an exponential runtime complexity. However, in the case of TS, runtime could rise exponentially with increasing the number of nodes.

In addition to that, TS was easily trapped in local optimum, whereas GA was able to overcome this issue. This is shown when comparing the average optimality gap across all 10 runs of both algorithms. TS was in the order of 15.0% while GA was in the order 3.0%.

Finally, it can be noted that GA outperforms the TS approach and is recommended to be used especially for higher number of spokes where TS is more likely to return a local optimum.

5.4 Operational Insights

The analysis of the results shows that the number of hubs selected has a consistent effect of lowering the total network cost as shown in figures 1, 2 and 3. This is due to making use of the economies of scale principle. The more hubs that are present, the more transportation logistics can be aggregated to share the cost across multiple items. The discount factor measures the quality of the hubs. Under the studied values of the discount factor we notice that the total network cost increases proportionally with it. However, what is more interesting is that the best solution found is different for every discount factor of the same data set under the same number of hubs. This observation points to the importance of determining the best approximation of the discount factor before deciding on the location of the hubs and node allocations.

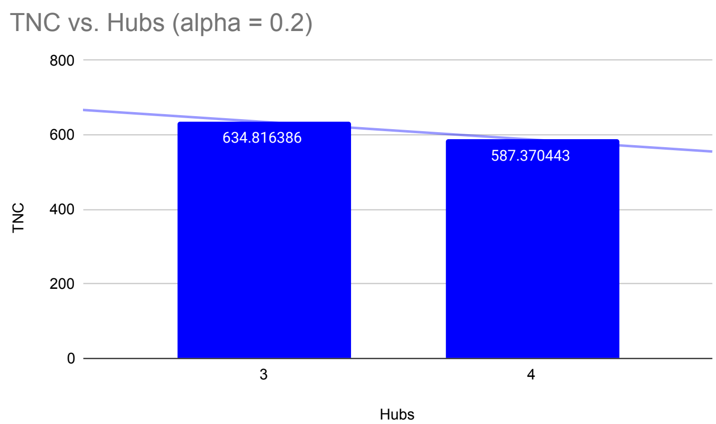

Figure 1: TNC vs. Hubs (alpha = 0.2) - 55 Nodes TR Dataset

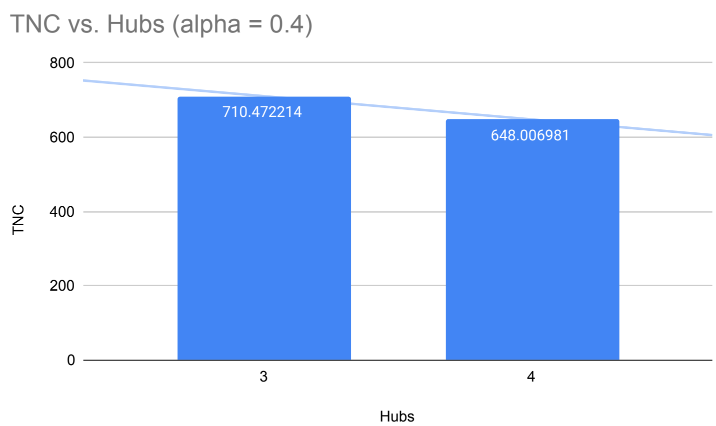

Figure 2: TNC vs. Hubs (alpha = 0.4) - 55 Nodes TR Dataset

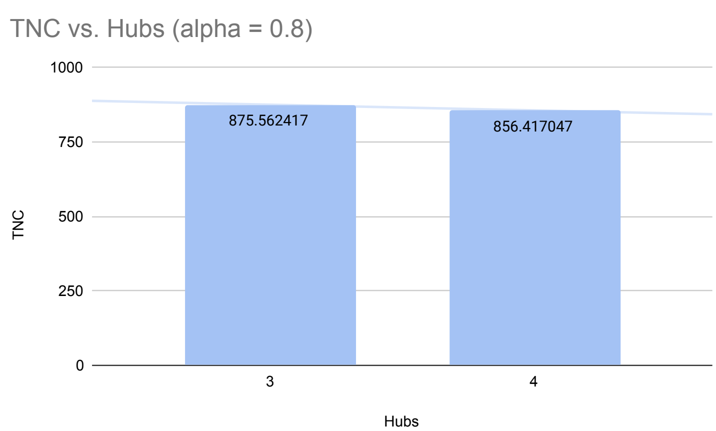

Figure 3: TNC vs. Hubs (alpha = 0.8) - 55 Nodes TR Dataset

6 Assumptions, Limitations and Future Work

This paper studied the implementation of TS and GA for finding the best solution for the CAB and TR dataset. The solutions of the algorithms were compared to the given optimal solutions of the CAB and TR datasets. An assumption was taken on the number of hubs required which was fixed to 3 and 4. The algorithms were run on single cores, however, the use of Golang as a programming language was intentional of developing implementations that make use of multiple cores on the machine. This was not implemented due to time limitations. However, the approach for improving the GA runtime would be to delegate the generation of the breeding pool to a sub process running independently on parts of the population and then combining the result. A similar divide and conquer approach would also be useful when selecting the best organism from each generated population.

As for TS, the generation of neighbor candidates can also be distributed across multiple processes which can also accommodate the integration of multiple neighborhood structures at once. Currently the generation of neighbors is done randomly, however this could be improved to generate all neighbors of a solution explicitly. In the case of GA, an analysis of the mutation rate parameter could also be considered to test its effect on the results.

In addition to implementation improvements, a sensitivity analysis would be beneficial on the number nodes respective to each level of discount factor.

References

[1] P. Jaillet, G. Song, and G. Yu, "Airline network design and hub location problems," *Location Science*, vol. 4, no. 3, pp. 195 – 212, 1996. Hub Location.

[2] S. Nickel, A. Schöbel, and T. Sonneborn, *Hub Location Problems in Urban Traffic Networks*, pp. 95–107. Boston, MA: Springer US, 2001.

[3] J. Klincewicz, "Heuristics for the p-hub location problem," *European Journal of Operational Research*, vol. 53, no. 1, pp. 25 – 37, 1991.

[4] H. Calk, S. A. Alumur, B. Y. Kara, and O. E. Karasan, "A tabu-search based heuristic for the hub covering problem over incomplete hub networks," *Computers & Operations Research*, vol. 36, no. 12, pp. 3088 – 3096, 2009. New developments on hub location.

[5] C. B. Cunha and M. R. Silva, "A genetic algorithm for the problem of configuring a hub-and-spoke network for a ltl trucking company in brazil," *European Journal of Operational Research*, vol. 179, no. 3, pp. 747 – 758, 2007.

Appendices

A Tabu Search Results

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run	Total Time	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[6 5 3]	491.934331	583.156188	1.464821038s	14.81374772s	491.93	0.00%	18.54%
		0.4	[6 2 3]	590.487572	620.113674	1.492874098s	14.945236175s	567.91	3.98%	9.19%
		0.8	[3 6 5]	719.869732	787.54857	1.496282026s	15.09700492s	716.98	0.40%	9.84%
	4	0.2	[6 2 5 3]	403.195915	471.823204	2.153886541s	17.982692755s	395.13	2.04%	19.41%
		0.4	[3 5 2 6]	494.955132	534.877781	1.546430889s	15.553967827s	493.79	0.24%	8.32%
		0.8	[3 5 0 6]	668.530405	718.513767	1.543722042s	15.334478779s	661.41	1.08%	8.63%
15 (CAB)	3	0.2	[3 11 12]	823.447845	957.276642	3.321190191s	27.840879354s	799.97	2.93%	19.66%
		0.4	[11 7 3]	987.367033	1048.238309	2.518827991s	25.16849998s	905.1	9.09%	15.81%
		0.8	[0 3 10]	1121.830943	1174.692239	2.694955359s	26.274714843s	1099.51	2.03%	6.84%
	4	0.2	[11 3 0 6]	679.303822	828.801316	2.609406083s	26.829996475s	639.77	6.18%	29.55%
		0.4	[3 11 5 6]	823.696862	904.054701	2.58939191s	26.968257104s	779.71	5.64%	15.95%
		0.8	[3 6 11 0]	1060.50087	1143.04363	2.537903559s	25.770521856s	1026.52	3.31%	11.35%
20 (CAB)	3	0.2	[16 3 6]	814.3378	871.441948	4.156887594s	41.250644731s	724.54	12.39%	20.28%
		0.4	[4 6 16]	900.882491	979.841076	3.931960404s	39.485899454s	847.77	6.26%	15.58%
		0.8	[3 19 11]	1113.422305	1167.886355	4.280156274s	41.292299721s	1091.05	2.05%	7.04%
	4	0.2	[6 16 3 11]	634.245109	705.256159	4.643569933s	46.465619878s	577.62	9.80%	22.10%
		0.4	[11 4 10 16]	815.285571	882.15131	4.631950228s	48.374253253s	727.1	12.13%	21.32%
		0.8	[16 13 8 10]	1034.408966	1071.370035	4.597993121s	49.907630003s	1008.49	2.57%	6.24%
25 (CAB)	3	0.2	[17 20 11]	790.462894	973.701163	6.845905336s	1m8.26509493s	767.35	3.01%	26.89%
		0.4	[17 11 3]	914.532207	1063.922451	6.950189149s	1m9.065813281s	901.7	1.42%	17.99%
		0.8	[6 11 19]	1213.970214	1258.046788	6.787076485s	1m8.580695377s	1158.83	4.76%	8.56%
	4	0.2	[3 16 11 0]	681.813476	832.608319	6.605478189s	1m8.62246824s	629.63	8.29%	32.24%
		0.4	[16 3 0 11]	790.653477	996.815135	7.102924839s	1m15.437034118s	787.51	0.40%	26.58%
		0.8	[16 19 11 20]	1149.028839	1204.887603	6.952239472s	1m23.109994584s	1087.66	5.64%	10.78%
Average									4.40%	16.20%

Table 1: Tabu Search CAB Results (Iterations: 100000, Tabu List Size: 5, Max Candidates: 60)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Optimal TNC	Gap %	Avg Gap %
25 (TR)	3	0.2	[14 8 2]	518.101606	569.152165	8.058580877s	1m20.295130212s	511.15	1.36%	11.35%
		0.4	[0 2 14]	633.11086	701.367986	7.549893353s	1m15.678053325s	597	6.05%	17.48%
		0.8	[17 2 14]	797.912402	831.202382	7.685185942s	1m16.98329331s	752.79	5.99%	10.42%
	4	0.2	[8 11 2 17]	410.524909	491.776798	7.843819696s	1m17.675215872s	403.51	1.74%	21.87%
		0.4	[2 0 14 22]	574.925842	614.856171	7.721734583s	1m21.471522719s	515.08	11.62%	19.37%
		0.8	[3 2 0 14]	728.071049	798.266272	8.596112888s	1m17.738948191s	705.47	3.20%	13.15%
55 (TR)	3	0.2	[44 29 14]	634.816386	703.207909	29.438036679s	5m3.309211051s	592.64	7.12%	18.66%
		0.4	[3 29 33]	710.472214	782.883434	29.262129511s	5m5.438599611s	684.27	3.83%	14.41%
		0.8	[3 33 29]	875.562417	920.643865	30.522072745s	5m27.195211349s	853.35	2.60%	7.89%
	4	0.2	[35 25 3 33]	587.370443	646.495938	31.026726038s	5m21.63586596s	501.74	17.07%	28.85%
		0.4	[29 3 14 54]	648.006981	726.519741	31.386824106s	5m24.882947182s	618.13	4.83%	17.54%
		0.8	[18 3 32 29]	856.417047	899.744565	30.726614217s	5m21.622299375s	812.37	5.42%	10.76%
Average									5.90%	15.98%

Table 2: Tabu Search TR Results (Iterations: 100000, Tabu List Size: 5, Max Candidates: 60)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run	Total Time	Iteration Found	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[3 6 2]	495.825585	552.048652	299.934529ms	2.898472633s	2	491.93	0.79%	12.22%
		0.4	[8 2 6]	592.61946	651.200324	285.824467ms	2.846715211s	2	567.91	4.35%	14.67%
		0.8	[3 8 6]	721.300979	822.095007	280.934465ms	2.847755314s	3	716.98	0.60%	14.66%
	4	0.2	[2 3 6 5]	395.130366	472.382976	248.813528ms	2.693458666s	0	395.13	0.00%	19.55%
		0.4	[7 3 5 6]	493.793763	536.456815	243.13592ms	2.464948731s	0	493.79	0.00%	8.64%
		0.8	[5 3 6 4]	673.487017	766.536009	245.573363ms	2.457009619s	0	661.41	1.83%	15.89%
15 (CAB)	3	0.2	[12 3 11]	812.523628	911.939044	627.377611ms	6.878483592s	1	799.97	1.57%	14.00%
		0.4	[11 3 6]	962.739039	1046.508734	754.816735ms	7.093875863s	3	905.1	6.37%	15.62%
		0.8	[6 0 3]	1133.597341	1194.616863	711.992639ms	7.568294939s	3	1099.51	3.10%	8.65%
	4	0.2	[6 5 3 11]	702.408721	831.677898	610.800936ms	6.584794378s	1	639.77	9.79%	30.00%
		0.4	[11 3 4 6]	829.078366	919.90233	617.684984ms	6.384596446s	2	779.71	6.33%	17.98%
		0.8	[7 6 3 4]	1050.594289	1115.384946	657.603844ms	6.781733975s	1	1026.52	2.35%	8.66%
20 (CAB)	3	0.2	[11 3 17]	813.103146	891.010588	1.293877017s	13.422820721s	2	724.54	12.22%	22.98%
		0.4	[11 17 3]	913.661709	989.783004	1.290956344s	12.94070172s	1	847.77	7.77%	16.75%
		0.8	[16 7 4]	1096.700765	1131.470721	1.2943522s	12.90772488s	3	1091.05	0.52%	3.70%
	4	0.2	[3 16 11 15]	604.123644	710.624244	1.253863575s	12.580921054s	2	577.62	4.59%	23.03%
		0.4	[16 13 3 11]	784.920865	845.041791	1.253864186s	12.649198188s	1	727.1	7.95%	16.22%
		0.8	[19 11 3 0]	1041.898434	1081.536266	1.253565071s	12.612546284s	4	1008.49	3.31%	7.24%
25 (CAB)	3	0.2	[16 11 4]	795.458629	969.51049	2.348982226s	23.511046744s	3	767.35	3.66%	26.35%
		0.4	[1 20 11]	916.832589	1044.715747	2.359686463s	23.534572975s	2	901.7	1.68%	15.86%
		0.8	[11 20 24]	1166.310709	1235.815973	2.349747948s	23.562091802s	3	1158.83	0.65%	6.64%
	4	0.2	[15 16 3 11]	654.962713	789.946272	2.952992323s	24.966383723s	3	629.63	4.02%	25.46%
		0.4	[3 13 17 11]	820.242298	956.149555	2.503174843s	23.791714642s	2	787.51	4.16%	21.41%
		0.8	[3 17 7 0]	1113.601616	1189.689043	2.224905055s	22.320704153s	2	1087.66	2.39%	9.38%
Average										3.75%	15.65%

Table 3: Tabu Search CAB Results (Iterations: 100000, Tabu List Size: Number of Nodes / 5, Max Candidates: Number of Nodes x 5, Aspiration: 40000)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Iteration	Optimal TNC	GAP %	Avg Gap %
25 (TR)	3	0.2	[14 2 8]	511.148025	562.741757	2.389972828s	24.219420201s	2	511.15	0.00%	10.09%
		0.4	[0 2 14]	630.967879	670.174583	2.472163426s	24.016737236s	5	597	5.69%	12.26%
		0.8	[0 14 2]	759.339411	832.77987	2.476174565s	25.329102811s	5	752.79	0.87%	10.63%
	4	0.2	[2 17 8 11]	419.162493	499.567427	2.338576626s	23.344978864s	4	403.51	3.88%	23.81%
		0.4	[0 17 2 14]	543.924846	608.426572	2.35629048s	25.421979134s	6	515.08	5.60%	18.12%
		0.8	[22 14 2 0]	747.05766	789.215184	2.349857878s	23.928577697s	5	705.47	5.90%	11.87%
55 (TR)	3	0.2	[29 33 3]	635.058973	689.856246	23.014211243s	3m54.298190763s	6	592.64	7.16%	16.40%
		0.4	[33 17 25]	724.66182	814.162041	21.439111297s	4m5.058123129s	1	684.27	5.90%	18.98%
		0.8	[0 29 3]	868.558557	910.199635	23.114475271s	3m47.073875811s	13	853.35	1.78%	6.66%
	4	0.2	[25 3 32 18]	513.97897	589.124494	22.090251052s	3m55.212688589s	1	501.74	2.44%	17.42%
		0.4	[44 11 25 14]	662.28727	719.587796	21.277701627s	3m33.484093531s	4	618.13	7.14%	16.41%
		0.8	[32 3 29 0]	826.036662	878.859091	21.275576594s	3m32.644329366s	11	812.37	1.68%	8.18%
										4.00%	14.24%

Table 4: Tabu Search TR Results (Iterations: 100000, Tabu List Size: Number of Nodes / 5, Max Candidates: Number of Nodes x 5, Aspiration: 40000)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run	Total Time	Iteration Found	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[2 6 3]	495.825585	591.085831	748.003s	7.699019ms	2	491.93	0.79%	20.16%
		0.4	[5 3 6]	575.302328	616.076783	991.756s	8.888797ms	1	567.91	1.30%	8.48%
		0.8	[3 6 5]	736.825317	796.011085	633.808s	7.112303ms	3	716.98	2.77%	11.02%
	4	0.2	[5 2 3 6]	395.130366	446.267366	685.921s	7.391405ms	0	395.13	0.00%	12.94%
		0.4	[3 5 2 6]	494.955132	537.437785	582.671s	7.070606ms	0	493.79	0.24%	8.84%
		0.8	[8 1 6 3]	698.621883	757.290186	605.357s	6.78761ms	2	661.41	5.63%	14.50%
15 (CAB)	3	0.2	[3 6 11]	835.977241	920.576474	1.868665ms	18.354368ms	2	799.97	4.50%	15.08%
		0.4	[3 6 11]	905.096021	1021.889909	1.666204ms	17.765851ms	1	905.1	0.00%	12.90%
		0.8	[11 12 3]	1137.484711	1192.911467	2.292117ms	18.680062ms	2	1099.51	3.45%	8.49%
	4	0.2	[6 13 3 11]	649.419512	772.109814	1.565634ms	15.583343ms	0	639.77	1.51%	20.69%
		0.4	[3 2 12 11]	860.069019	912.626814	1.906594ms	17.560046ms	4	779.71	10.31%	17.05%
		0.8	[6 0 11 3]	1051.426463	1109.696569	1.516902ms	15.482408ms	0	1026.52	2.43%	8.10%
20 (CAB)	3	0.2	[11 3 16]	772.098669	876.909309	3.573635ms	37.562905ms	0	724.54	6.56%	21.03%
		0.4	[16 4 11]	876.097732	970.616892	3.482012ms	35.889904ms	1	847.77	3.34%	14.49%
		0.8	[4 16 7]	1094.60598	1157.364258	4.51562ms	35.485541ms	3	1091.05	0.33%	6.08%
	4	0.2	[6 13 16 3]	655.142605	756.055386	3.84233ms	35.767889ms	2	577.62	13.42%	30.89%
		0.4	[16 3 0 11]	752.495432	881.299023	3.158318ms	35.762914ms	2	727.1	3.49%	21.21%
		0.8	[6 3 16 19]	1028.583983	1085.606253	3.505974ms	34.984519ms	5	1008.49	1.99%	7.65%
25 (CAB)	3	0.2	[16 11 3]	776.259924	924.490991	6.249869ms	63.127796ms	3	767.35	1.16%	20.48%
		0.4	[11 16 4]	912.114031	1048.402365	6.563684ms	65.085292ms	5	901.7	1.15%	16.27%
		0.8	[19 3 11]	1166.778683	1268.971508	6.544853ms	70.216617ms	2	1158.83	0.69%	9.50%
	4	0.2	[11 16 13 3]	639.704622	817.282682	6.539493ms	65.754229ms	3	629.63	1.60%	29.80%
		0.4	[17 12 11 3]	845.456059	958.143104	6.068187ms	62.703338ms	1	787.51	7.36%	21.67%
		0.8	[7 1 3 6]	1121.614335	1189.104548	7.38771ms	63.368375ms	5	1087.66	3.12%	9.33%
Average										3.21%	15.28%

Table 5: Tabu Search CAB Results (Iterations: 100, Tabu List Size: Number of Nodes / 5, Max Candidates: Number of Nodes x 5)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Iteration	Optimal TNC	GAP %	Avg Gap %
25 (TR)	3	0.2	[18 14 17]	536.815977	603.655945	6.034194ms	64.484275ms	0	511.15	5.02%	18.10%
		0.4	[14 2 0]	596.99509	684.079187	6.618658ms	64.208839ms	5	597	0.00%	14.59%
		0.8	[18 14 2]	774.752309	820.096357	6.244703ms	63.540901ms	3	752.79	2.92%	8.94%
	4	0.2	[2 18 11 17]	441.63628	486.877246	6.478897ms	64.381728ms	3	403.51	9.45%	20.66%
		0.4	[11 2 0 16]	533.25756	597.265489	6.865988ms	65.913439ms	2	515.08	3.53%	15.96%
		0.8	[2 8 0 14]	739.993668	798.732363	7.249421ms	66.032387ms	5	705.47	4.89%	13.22%
55 (TR)	3	0.2	[44 14 29]	625.967367	698.469855	66.417453ms	605.970262ms	1	592.64	5.62%	17.86%
		0.4	[3 33 29]	714.766602	784.406814	57.413548ms	594.193807ms	3	684.27	4.46%	14.63%
		0.8	[33 29 3]	886.704397	924.963148	59.404103ms	598.172604ms	0	853.35	3.91%	8.39%
	4	0.2	[3 18 31 25]	528.885368	637.94871	57.979712ms	590.598013ms	5	501.74	5.41%	27.15%
		0.4	[29 33 19 3]	671.40429	706.519065	59.664876ms	604.701431ms	2	618.13	8.62%	14.30%
		0.8	[0 3 31 25]	833.366608	899.190522	58.929283ms	595.77221ms	12	812.37	2.58%	10.69%
										4.70%	15.37%

Table 6: Tabu Search TR Results (Iterations: 100, Tabu List Size: Number of Nodes / 5, Max Candidates: Number of Nodes x 5)

B Genetic Algorithm Results

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Time (s)	Generation Found	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[6 3 5]	491.934331	491.934331	12.358773196s	1m59.484891228s	32	491.93	0.00%	0.00%
		0.4	[3 6 5]	567.912798	567.912798	12.722659686s	2m8.205338259s	33	567.91	0.00%	0.00%
		0.8	[8 6 3]	717.397641	719.622523	12.716637989s	2m6.327561719s	29	716.98	0.06%	0.37%
	4	0.2	[6 5 3 2]	395.130366	395.130366	13.697294766s	2m6.165471469s	52	395.13	0.00%	0.00%
		0.4	[5 3 7 6]	493.793763	494.258311	12.957123396s	2m9.09000009s	54	493.79	0.00%	0.09%
		0.8	[6 8 3 7]	661.828412	664.575592	12.986986454s	2m10.948198546s	36	661.41	0.06%	0.48%
15 (CAB)	3	0.2	[3 6 11]	801.745093	801.745093	13.265337277s	2m10.748847108s	53	799.97	0.22%	0.22%
		0.4	[11 6 3]	908.309424	926.649732	12.25715599s	2m7.088468131s	59	905.1	0.35%	2.38%
		0.8	[12 7 3]	1116.449971	1131.019653	12.122288783s	2m4.92198805s	46	1099.51	1.54%	2.87%
	4	0.2	[3 6 11 13]	649.419512	653.350879	12.192099278s	2m7.713539088s	61	639.77	1.51%	2.12%
		0.4	[3 11 6 0]	786.697598	799.20236	12.503320963s	2m15.625593908s	62	779.71	0.90%	2.50%
		0.8	[7 6 3 0]	1038.360435	1059.032596	13.399089267s	2m10.147774077s	64	1026.52	1.15%	3.17%
20 (CAB)	3	0.2	[16 3 11]	724.537984	729.934119	13.632334277s	2m14.10882911s	59	724.54	0.00%	0.74%
		0.4	[16 3 11]	865.040342	880.599766	12.946254777s	2m15.799480085s	65	847.77	2.04%	3.87%
		0.8	[10 16 3]	1096.089685	1101.324386	13.547305443s	2m19.404467922s	66	1091.05	0.46%	0.94%
	4	0.2	[11 3 16 15]	582.557248	591.651292	13.917445378s	2m24.748448202s	80	577.62	0.85%	2.43%
		0.4	[3 11 16 0]	742.773402	763.29874	13.72068766s	2m15.165916359s	72	727.1	2.16%	4.98%
		0.8	[16 0 6 3]	1022.59767	1037.716362	13.203457275s	2m12.5622781s	47	1008.49	1.40%	2.90%
25 (CAB)	3	0.2	[11 3 16]	770.551998	770.927704	15.031201216s	2m22.275882507s	74	767.35	0.42%	0.47%
		0.4	[17 11 3]	911.249076	912.549854	13.875973492s	2m25.791815293s	63	901.7	1.06%	1.20%
		0.8	[11 24 20]	1170.468093	1193.674875	14.401022573s	2m27.727180311s	49	1158.83	1.00%	3.01%
	4	0.2	[16 11 3 15]	636.7198	648.435857	13.732749265s	2m21.174402147s	80	629.63	1.13%	2.99%
		0.4	[11 3 0 17]	802.992007	829.309805	13.018820945s	2m22.285122928s	61	787.51	1.97%	5.31%
		0.8	[11 3 10 24]	1113.937496	1131.198363	14.112957506s	2m23.439405524s	61	1087.66	2.42%	4.00%
Average										0.86%	1.96%

Table 7: Genetic Algorithm CAB Results (Population: 5000, generations: 100)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Generation Found	Optimal TNC	GAP %	Avg Gap %
25 (TR)	3	0.2	[14 2 8]	511.148025	512.448113	14.235893917s	2m20.726190011s	63	511.15	0.00%	0.25%
		0.4	[14 2 0]	596.99509	596.99509	16.873461693s	2m30.542042166s	58	597	0.00%	0.00%
		0.8	[2 14 0]	752.788164	752.830488	15.400242153s	2m33.302573117s	48	752.79	0.00%	0.01%
	4	0.2	[17 11 8 2]	403.507465	413.88913	14.907816908s	2m21.922360713s	75	403.51	0.00%	2.57%
		0.4	[11 17 2 0]	515.08186	524.896425	12.717628547s	2m7.29791507s	64	515.08	0.00%	1.91%
		0.8	[14 2 17 0]	707.892172	719.182709	13.425338245s	2m17.624411028s	64	705.47	0.34%	1.94%
55 (TR)	3	0.2	[29 18 3]	592.822144	597.901463	21.07940106s	3m36.314903294s	77	592.64	0.03%	0.89%
		0.4	[29 33 3]	686.683094	692.162688	23.313951333s	3m42.848036533s	77	684.27	0.35%	1.15%
		0.8	[29 3 0]	857.641813	865.581334	23.789444651s	3m57.964988753s	75	853.35	0.50%	1.43%
	4	0.2	[29 18 32 3]	517.089303	539.81765	23.081919432s	3m48.988267565s	66	501.74	3.06%	7.59%
		0.4	[25 54 3 31]	625.746253	648.35782	23.653632875s	3m51.541515257s	73	618.13	1.23%	4.89%
		0.8	[3 29 0 31]	825.350722	840.831885	22.7620848s	3m47.309791092s	63	812.37	1.60%	3.50%
										0.59%	2.18%

Table 8: Genetic Algorithm TR Results (Population: 5000, generations: 100)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Time (s)	Avg Generations	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[5 3 6]	491.934331	499.001883	131.069711ms	700.355479ms	24	491.93	0.00%	1.44%
		0.4	[6 3 5]	567.912798	578.459302	60.644589ms	611.692411ms	24	567.91	0.00%	1.86%
		0.8	[6 8 3]	717.397641	718.164306	92.414298ms	857.71526ms	72	716.98	0.06%	0.17%
	4	0.2	[5 2 3 6]	395.130366	399.607602	82.94212ms	663.355585ms	29	395.13	0.00%	1.13%
		0.4	[7 5 3 6]	493.793763	498.94363	59.324645ms	715.030636ms	48	493.79	0.00%	1.04%
		0.8	[8 7 6 3]	661.828412	665.868666	61.269628ms	628.961158ms	33	661.41	0.06%	0.67%
15 (CAB)	3	0.2	[3 6 11]	799.971107	814.891913	95.473038ms	832.081679ms	43	799.97	0.00%	1.87%
		0.4	[3 6 11]	908.309424	949.338165	90.715219ms	860.374529ms	46	905.1	0.35%	4.89%
		0.8	[12 3 7]	1110.522031	1138.467076	119.413668ms	826.807971ms	44	1099.51	1.00%	3.54%
	4	0.2	[11 13 3 6]	639.775329	684.326209	108.471036ms	894.447213ms	48	639.77	0.00%	6.96%
		0.4	[13 6 11 3]	781.958103	813.519848	131.362909ms	974.009919ms	64	779.71	0.29%	4.34%
		0.8	[7 6 3 0]	1038.360435	1067.783762	67.964216ms	816.08416ms	40	1026.52	1.15%	4.02%
20 (CAB)	3	0.2	[16 3 11]	724.537984	766.386442	134.044817ms	1.172943614s	60	724.54	0.00%	5.78%
		0.4	[16 11 3]	865.040342	915.296464	111.507341ms	1.05949602s	44	847.77	2.04%	7.97%
		0.8	[0 3 16]	1097.251762	1103.987537	116.667727ms	1.126421909s	55	1091.05	0.57%	1.19%
	4	0.2	[16 3 11 0]	590.986146	619.320641	124.84775ms	1.096609911s	46	577.62	2.31%	7.22%
		0.4	[12 16 11 3]	743.239176	776.594366	186.442427ms	1.366081785s	78	727.1	2.22%	6.81%
		0.8	[5 16 3 0]	1036.432013	1054.530039	74.530771ms	918.801243ms	24	1008.49	2.77%	4.57%
25 (CAB)	3	0.2	[3 11 16]	767.349393	804.655596	124.401258ms	1.227616013s	35	767.35	0.00%	4.86%
		0.4	[20 11 1]	913.084733	957.113112	141.677247ms	1.455910405s	59	901.7	1.26%	6.15%
		0.8	[24 3 18]	1188.922587	1214.711988	135.959191ms	1.221721091s	32	1158.83	2.60%	4.82%
	4	0.2	[11 16 3 0]	635.824451	664.173625	221.657884ms	1.399257458s	52	629.63	0.98%	5.49%
		0.4	[23 11 16 3]	803.586111	851.064373	156.641195ms	1.494719162s	60	787.51	2.04%	8.07%
		0.8	[3 17 11 0]	1107.726022	1139.604439	159.383415ms	1.419514296s	52	1087.66	1.84%	4.78%
Average										0.90%	4.15%

Table 9: Genetic Algorithm CAB Results (Population: 100, generations: 1000, aspiration: 100)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Generation Found	Optimal TNC	GAP %	Avg Gap %
25 (TR)	3	0.2	[14 8 2]	511.148025	535.037795	142.06831ms	1.274287529s	38	511.15	0.00%	4.67%
		0.4	[2 14 0]	596.99509	613.09351	134.533046ms	1.210375767s	27	597	0.00%	2.70%
		0.8	[2 14 0]	752.788164	784.19646	168.149233ms	1.220711492s	31	752.79	0.00%	4.17%
	4	0.2	[17 2 0 11]	411.108911	440.288154	119.942496ms	1.318557995s	35	403.51	1.88%	9.11%
		0.4	[17 11 0 2]	515.08186	539.768759	126.277676ms	1.457406618s	53	515.08	0.00%	4.79%
		0.8	[17 2 0 14]	707.892172	744.395152	104.921489ms	1.274757575s	37	705.47	0.34%	5.52%
55 (TR)	3	0.2	[29 3 54]	602.495102	625.249199	395.45848ms	4.134340634s	49	592.64	1.66%	5.50%
		0.4	[54 11 3]	704.596	733.134959	560.932121ms	5.320461204s	93	684.27	2.97%	7.14%
		0.8	[29 3 33]	867.086885	904.961115	693.320224ms	5.110983312s	84	853.35	1.61%	6.05%
	4	0.2	[3 25 33 31]	524.568182	559.736285	480.104051ms	4.109759372s	44	501.74	4.55%	11.56%
		0.4	[33 25 3 26]	622.441638	657.237585	790.645189ms	5.581193653s	88	618.13	0.70%	6.33%
		0.8	[25 0 31 3]	820.23128	849.799267	359.592579ms	4.610160269s	48	812.37	0.97%	4.61%
										1.22%	6.01%

Table 10: Genetic Algorithm TR Results (Population: 100, generations: 1000, aspiration: 100)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Time (s)	Avg Generations	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[5 6 3]	491.934331	495.992812	73.790874ms	635.649434ms	23	491.93	0.00%	0.83%
		0.4	[6 5 3]	567.912798	570.823702	61.096837ms	570.133754ms	17	567.91	0.00%	0.51%
		0.8	[6 8 3]	717.397641	722.668167	90.643404ms	833.409549ms	72	716.98	0.06%	0.79%
	4	0.2	[3 5 2 6]	395.130366	408.241936	74.59085ms	624.940054ms	26	395.13	0.00%	3.32%
		0.4	[7 5 6 3]	493.793763	498.576707	70.176671ms	669.130051ms	35	493.79	0.00%	0.97%
		0.8	[3 8 6 7]	661.415348	676.454222	92.542551ms	738.949318ms	53	661.41	0.00%	2.27%
15 (CAB)	3	0.2	[11 6 3]	799.971107	829.244309	90.337164ms	888.31317ms	52	799.97	0.00%	3.66%
		0.4	[6 11 3]	908.309424	960.620099	100.371272ms	809.534062ms	36	905.1	0.35%	6.13%
		0.8	[12 3 7]	1116.449971	1132.587703	72.265091ms	879.474638ms	54	1099.51	1.54%	3.01%
	4	0.2	[3 13 11 6]	649.419512	681.409606	84.564641ms	835.453758ms	39	639.77	1.51%	6.51%
		0.4	[3 6 11 0]	786.697598	819.611421	83.967992ms	838.508985ms	39	779.71	0.90%	5.12%
		0.8	[7 3 6 0]	1038.360435	1066.475492	59.780023ms	842.793359ms	42	1026.52	1.15%	3.89%
20 (CAB)	3	0.2	[3 16 11]	731.283153	774.115225	102.308286ms	995.803108ms	34	724.54	0.93%	6.84%
		0.4	[17 3 11]	876.135376	909.006139	115.27376ms	971.524641ms	33	847.77	3.35%	7.22%
		0.8	[3 16 10]	1096.089685	1109.600329	91.017574ms	1.277086046s	79	1091.05	0.46%	1.70%
	4	0.2	[3 11 16 15]	582.557248	615.739168	99.899736ms	1.130773251s	50	577.62	0.85%	6.60%
		0.4	[0 16 11 3]	727.098946	788.132898	162.208464ms	1.309974377s	75	727.1	0.00%	8.39%
		0.8	[6 3 17 0]	1029.824925	1050.766062	117.135524ms	1.245398125s	67	1008.49	2.12%	4.19%
25 (CAB)	3	0.2	[16 11 3]	767.349393	804.193098	207.292296ms	1.39287454s	49	767.35	0.00%	4.80%
		0.4	[3 11 17]	911.249076	945.286951	155.368284ms	1.596168543s	56	901.7	1.06%	4.83%
		0.8	[24 7 20]	1194.019666	1213.814397	202.937346ms	1.343040287s	44	1158.83	3.04%	4.74%
	4	0.2	[11 3 16 23]	632.380138	686.244588	112.74794ms	1.421868205s	51	629.63	0.44%	8.99%
		0.4	[20 16 0 11]	827.405862	859.289741	114.983205ms	1.35261124s	46	787.51	5.07%	9.11%
		0.8	[4 1 6 11]	1121.563681	1142.052527	166.257638ms	1.51793744s	60	1087.66	3.12%	5.00%
Average										1.08%	4.56%

Table 11: Genetic Algorithm CAB Results (Population: 100, generations: 1000, aspiration: 500)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Generation Found	Optimal TNC	GAP %	Avg Gap %
25 (TR)	3	0.2	[8 2 14]	511.148025	525.911964	146.023909ms	1.31315214s	40	511.15	0.00%	2.89%
		0.4	[2 0 14]	596.99509	624.476913	126.801063ms	1.349021966s	43	597	0.00%	4.60%
		0.8	[2 0 14]	752.788164	791.101974	125.493736ms	1.408696636s	50	752.79	0.00%	5.09%
	4	0.2	[8 11 17 2]	403.507465	433.928248	152.730354ms	1.337569676s	39	403.51	0.00%	7.54%
		0.4	[14 2 12 0]	523.931576	546.103506	124.710254ms	1.373731256s	48	515.08	1.72%	6.02%
		0.8	[0 14 2 17]	707.892172	727.007691	113.706487ms	1.524687486s	48	705.47	0.34%	3.05%
55 (TR)	3	0.2	[3 33 29]	596.249992	639.60348	531.678195ms	4.284262146s	48	592.64	0.61%	7.92%
		0.4	[29 3 54]	688.697655	727.518349	615.6766ms	5.075426318s	74	684.27	0.65%	6.32%
		0.8	[54 29 3]	868.622609	898.314315	327.028354ms	4.373196713s	52	853.35	1.79%	5.27%
	4	0.2	[11 33 25 3]	527.430804	557.635573	451.325359ms	4.999814097s	74	501.74	5.12%	11.14%
		0.4	[3 18 32 29]	627.354203	661.718944	533.139579ms	4.712412251s	71	618.13	1.49%	7.05%
		0.8	[29 44 3 0]	840.512752	856.791254	900.478696ms	5.902704092s	104	812.37	3.46%	5.47%
										1.27%	6.03%

Table 12: Genetic Algorithm TR Results (Population: 100, generations: 1000, aspiration: 500)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Time (s)	Generations	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[3 6 5]	491.934331	491.934331	201.879323ms	1.631739826s	27	491.93	0.00%	0.00%
		0.4	[6 5 3]	567.912798	570.823702	142.471042ms	1.479907893s	24	567.91	0.00%	0.51%
		0.8	[3 6 8]	717.397641	719.38366	181.831491ms	1.684462243s	43	716.98	0.06%	0.34%
	4	0.2	[5 2 3 6]	395.130366	396.082171	164.799052ms	1.735144275s	41	395.13	0.00%	0.24%
		0.4	[5 7 6 3]	493.793763	496.593925	124.790525ms	1.748826516s	44	493.79	0.00%	0.57%
		0.8	[3 7 6 8]	661.415348	663.964163	153.317816ms	2.118859291s	79	661.41	0.00%	0.39%
15 (CAB)	3	0.2	[6 11 3]	801.745093	804.905442	207.536039ms	1.943321996s	39	799.97	0.22%	0.62%
		0.4	[3 11 6]	908.309424	947.422633	414.137792ms	2.576532839s	86	905.1	0.35%	4.68%
		0.8	[7 3 12]	1116.449971	1129.038876	142.551219ms	2.049297029s	51	1099.51	1.54%	2.69%
	4	0.2	[13 11 3 6]	649.419512	661.189555	216.01534ms	2.304523293s	61	639.77	1.51%	3.35%
		0.4	[6 0 3 11]	782.702722	792.642674	216.888552ms	2.466750495s	77	779.71	0.38%	1.66%
		0.8	[7 3 0 6]	1038.360435	1061.574763	155.653803ms	2.20058684s	62	1026.52	1.15%	3.41%
20 (CAB)	3	0.2	[3 16 11]	724.537984	744.051703	290.148574ms	2.457421051s	45	724.54	0.00%	2.69%
		0.4	[16 3 11]	865.040342	888.239762	269.638854ms	2.611009143s	45	847.77	2.04%	4.77%
		0.8	[16 8 10]	1096.303216	1106.039544	285.64291ms	2.781449427s	72	1091.05	0.48%	1.37%
	4	0.2	[11 3 16 0]	585.040454	603.994079	414.294099ms	3.499714864s	84	577.62	1.28%	4.57%
		0.4	[11 3 16 0]	727.098946	760.946263	394.700936ms	3.48264553s	96	727.1	0.00%	4.65%
		0.8	[10 16 3 0]	1022.411689	1036.052414	204.17312ms	2.9835206s	77	1008.49	1.38%	2.73%
25 (CAB)	3	0.2	[3 11 16]	767.349393	776.755489	408.269158ms	3.411209626s	49	767.35	0.00%	1.23%
		0.4	[11 16 3]	903.493086	912.760519	462.175602ms	4.115620947s	85	901.7	0.20%	1.23%
		0.8	[11 24 20]	1170.468093	1193.353921	278.790328ms	3.194216704s	53	1158.83	1.00%	2.98%
	4	0.2	[11 3 16 23]	632.380138	657.357605	351.855789ms	3.843405659s	72	629.63	0.44%	4.40%
		0.4	[11 3 16 0]	795.997586	827.700546	421.652797ms	3.311196619s	45	787.51	1.08%	5.10%
		0.8	[11 16 3 0]	1090.896182	1125.084621	443.102891ms	3.765609705s	69	1087.66	0.30%	3.44%
Average										0.56%	2.40%

Table 13: Genetic Algorithm CAB Results (Population: 200, generations: 1000, aspiration: 100)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Generation Found	Optimal TNC	GAP %	Avg Gap %
25 (TR)	3	0.2	[2 14 8]	511.148025	518.008539	324.844582ms	3.414482986s	51	511.15	0.00%	1.34%
		0.4	[14 0 2]	596.99509	604.073507	393.450436ms	3.355948165s	44	597	0.00%	1.18%
		0.8	[2 14 0]	752.788164	758.56784	350.186512ms	3.354190559s	64	752.79	0.00%	0.77%
	4	0.2	[17 8 11 2]	403.507465	422.967617	325.702542ms	3.620728107s	63	403.51	0.00%	4.82%
		0.4	[14 0 2 17]	520.830183	539.392245	371.975725ms	3.442338327s	55	515.08	1.12%	4.72%
		0.8	[2 17 0 14]	705.755094	718.759231	429.970957ms	3.688845348s	73	705.47	0.04%	1.88%
55 (TR)	3	0.2	[18 29 3]	595.583563	608.280328	931.595522ms	11.459920411s	82	592.64	0.50%	2.64%
		0.4	[54 3 29]	688.627425	711.325151	1.581513667s	11.687509209s	82	684.27	0.64%	3.95%
		0.8	[3 29 0]	857.641813	877.517539	943.990833ms	10.156420149s	64	853.35	0.50%	2.83%
	4	0.2	[32 25 33 3]	506.353943	541.850134	1.299176434s	12.520123676s	92	501.74	0.92%	7.99%
		0.4	[3 54 26 25]	623.530675	652.712897	931.282617ms	12.900451982s	113	618.13	0.87%	5.59%
		0.8	[31 29 3 0]	815.164564	846.084918	1.135645776s	13.118218306s	120	812.37	0.34%	4.15%
										0.41%	3.49%

Table 14: Genetic Algorithm TR Results (Population: 200, generations: 1000, aspiration: 100)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Time (s)	Avg Generations	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[5 3 6]	491.934331	494.41989	151.368048ms	1.507638506s	23	491.93	0.00%	0.51%
		0.4	[3 6 5]	567.912798	569.36825	171.112693ms	1.568642492s	31	567.91	0.00%	0.26%
		0.8	[3 6 8]	716.982795	719.172716	232.921491ms	1.964595588s	67	716.98	0.00%	0.31%
	4	0.2	[5 3 6 2]	395.130366	399.799317	158.569108ms	1.745707743s	38	395.13	0.00%	1.18%
		0.4	[6 3 5 7]	493.793763	497.594194	210.053734ms	1.779456754s	51	493.79	0.00%	0.77%
		0.8	[7 3 6 5]	661.86863	666.992871	131.750135ms	1.708952193s	41	661.41	0.07%	0.84%
15 (CAB)	3	0.2	[6 3 11]	799.971107	806.054751	247.791084ms	2.387528413s	73	799.97	0.00%	0.76%
		0.4	[3 12 11]	916.010171	939.536768	175.927174ms	1.91014678s	35	905.1	1.21%	3.80%
		0.8	[3 7 12]	1110.522031	1128.989317	271.467309ms	1.932261039s	42	1099.51	1.00%	2.68%
	4	0.2	[6 13 3 11]	649.419512	662.356737	216.871784ms	2.17400158s	50	639.77	1.51%	3.53%
		0.4	[11 6 3 0]	782.702722	794.187064	287.130717ms	2.316072694s	67	779.71	0.38%	1.86%
		0.8	[3 7 6 0]	1038.360435	1065.104446	208.546654ms	2.086975096s	59	1026.52	1.15%	3.76%
20 (CAB)	3	0.2	[11 3 16]	724.537984	763.404328	305.110193ms	2.654047204s	57	724.54	0.00%	5.36%
		0.4	[16 11 3]	865.040342	885.993621	294.955177ms	2.446840963s	46	847.77	2.04%	4.51%
		0.8	[3 16 10]	1096.089685	1099.875487	255.318529ms	2.426882064s	50	1091.05	0.46%	0.81%
	4	0.2	[11 3 16 0]	585.040454	605.552033	345.64845ms	3.004045408s	77	577.62	1.28%	4.84%
		0.4	[11 16 3 0]	727.098946	755.280693	331.865304ms	3.053908818s	86	727.1	0.00%	3.88%
		0.8	[0 16 10 3]	1022.411689	1036.1352	253.291178ms	3.071016388s	88	1008.49	1.38%	2.74%
25 (CAB)	3	0.2	[3 11 16]	767.349393	786.809868	417.371357ms	3.320148724s	68	767.35	0.00%	2.54%
		0.4	[17 11 3]	901.698844	914.323901	411.296557ms	3.184163706s	58	901.7	0.00%	1.40%
		0.8	[24 11 20]	1167.774404	1191.098018	396.478738ms	3.306437025s	70	1158.83	0.77%	2.78%
	4	0.2	[3 11 23 16]	632.380138	658.219368	324.311708ms	3.365128077s	58	629.63	0.44%	4.54%
		0.4	[16 11 3 0]	787.515028	820.550169	476.339675ms	4.119757915s	88	787.51	0.00%	4.20%
		0.8	[20 3 24 11]	1105.472698	1131.961337	445.857831ms	3.698019516s	87	1087.66	1.64%	4.07%
Average										0.56%	2.58%

Table 15: Genetic Algorithm CAB Results (Population: 200, generations: 200)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Generation Found	Optimal TNC	GAP %	Avg Gap %
25 (TR)	3	0.2	[14 8 2]	511.148025	519.591209	306.470568ms	3.166196235s	46	511.15	0.00%	1.65%
		0.4	[2 0 14]	596.99509	601.208398	418.814696ms	3.369760661s	51	597	0.00%	0.70%
		0.8	[2 14 0]	752.788164	767.943321	258.828382ms	3.081998707s	48	752.79	0.00%	2.01%
	4	0.2	[11 8 17 2]	403.507465	414.933188	403.506909ms	3.750891049s	64	403.51	0.00%	2.83%
		0.4	[0 17 2 11]	515.08186	528.687761	371.892205ms	3.507190627s	56	515.08	0.00%	2.64%
		0.8	[2 17 14 0]	707.892172	728.472615	373.494469ms	3.648909801s	74	705.47	0.34%	3.26%
55 (TR)	3	0.2	[3 18 29]	592.821514	615.424165	1.138957235s	10.842985548s	81	592.64	0.03%	3.84%
		0.4	[18 3 29]	688.363023	716.75697	1.066339179s	10.618595031s	82	684.27	0.60%	4.75%
		0.8	[3 29 0]	857.435578	878.626147	1.095080492s	11.18268524s	90	853.35	0.48%	2.96%
	4	0.2	[33 25 31 3]	524.568182	542.31778	1.078730274s	10.883561501s	86	501.74	4.55%	8.09%
		0.4	[3 31 18 25]	626.049051	649.830093	1.179206561s	11.292419759s	101	618.13	1.28%	5.13%
		0.8	[29 3 31 0]	815.164564	837.653304	1.224020507s	11.409606934s	106	812.37	0.34%	3.11%
										0.64%	3.42%

Table 16: Genetic Algorithm TR Results (Population: 200, generations: 200)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Time (s)	Avg Generations	Optimal TNC	Gap %	Avg Gap %
10 (CAB)	3	0.2	[5 3 6]	491.934331	492.712582	326.992684ms	2.927938049s	33	491.93	0.00%	0.16%
		0.4	[3 5 6]	567.912798	569.36825	314.789101ms	2.783887442s	31	567.91	0.00%	0.26%
		0.8	[8 3 6]	717.397641	718.642033	398.50974ms	3.534943389s	79	716.98	0.06%	0.23%
	4	0.2	[3 5 2 6]	395.130366	395.130366	373.384534ms	3.399409923s	51	395.13	0.00%	0.00%
		0.4	[6 7 5 3]	493.793763	496.544653	278.341803ms	3.460642546s	46	493.79	0.00%	0.56%
		0.8	[5 7 3 6]	661.86863	664.963041	495.168218ms	3.306111682s	58	661.41	0.07%	0.54%
15 (CAB)	3	0.2	[3 11 6]	799.971107	805.278665	534.226512ms	4.101503237s	69	799.97	0.00%	0.66%
		0.4	[11 6 3]	908.309424	927.236375	460.456855ms	4.167370473s	84	905.1	0.35%	2.45%
		0.8	[7 3 12]	1110.522031	1127.488902	485.489656ms	3.896857763s	68	1099.51	1.00%	2.54%
	4	0.2	[6 3 11 13]	639.775329	662.549681	376.339882ms	3.775899397s	51	639.77	0.00%	3.56%
		0.4	[3 0 11 6]	782.702722	793.544425	472.970707ms	3.751120349s	63	779.71	0.38%	1.77%
		0.8	[3 7 6 0]	1038.360435	1055.419544	446.450113ms	3.933028726s	59	1026.52	1.15%	2.82%
20 (CAB)	3	0.2	[3 16 11]	724.537984	739.998682	753.586648ms	5.558973853s	70	724.54	0.00%	2.13%
		0.4	[3 11 16]	865.040342	871.780159	380.509183ms	4.915072892s	49	847.77	2.04%	2.83%
		0.8	[10 3 16]	1096.089685	1102.145787	357.614077ms	4.502661521s	52	1091.05	0.46%	1.02%
	4	0.2	[16 15 3 11]	577.621421	599.633619	711.530921ms	5.854993757s	77	577.62	0.00%	3.81%
		0.4	[16 11 0 3]	727.098946	747.215965	598.654971ms	6.037426881s	106	727.1	0.00%	2.77%
		0.8	[3 16 0 10]	1022.411689	1030.584922	602.061116ms	4.813091694s	67	1008.49	1.38%	2.19%
25 (CAB)	3	0.2	[11 3 16]	767.349393	774.144874	582.78076ms	6.095137547s	82	767.35	0.00%	0.89%
		0.4	[1 3 11]	910.26105	921.342238	515.117706ms	5.713304473s	69	901.7	0.95%	2.18%
		0.8	[3 19 11]	1169.271226	1190.198002	636.650371ms	4.991219251s	63	1158.83	0.90%	2.71%
	4	0.2	[23 3 11 16]	629.633862	648.01099	675.363108ms	6.17463545s	80	629.63	0.00%	2.92%
		0.4	[11 0 3 16]	795.997586	825.208899	705.57125ms	6.482444104s	100	787.51	1.08%	4.79%
		0.8	[1 20 11 3]	1109.818774	1121.007894	718.073534ms	6.364468849s	96	1087.66	2.04%	3.07%
Average										0.49%	1.95%

Table 17: Genetic Algorithm CAB Results (Population: 300, generations: 200)

n	p	alpha	Hub Locations	TNC	Avg TNC	Time per Run (s)	Total Time (s)	Generation Found	Optimal TNC	GAP %	Avg Gap %
25 (TR)	3	0.2	[8 2 14]	511.148025	513.188563	470.258878ms	5.911993761s	67	511.15	0.00%	0.40%
		0.4	[2 14 0]	596.99509	601.011463	517.82927ms	5.583947553s	46	597	0.00%	0.67%
		0.8	[2 0 14]	752.788164	752.788164	640.276274ms	6.531025546s	74	752.79	0.00%	0.00%
	4	0.2	[2 8 11 17]	403.507465	413.11648	517.301703ms	6.550267045s	83	403.51	0.00%	2.38%
		0.4	[11 2 0 17]	515.08186	533.317108	569.010998ms	6.027481313s	67	515.08	0.00%	3.54%
		0.8	[14 17 2 0]	707.892172	715.878699	453.306302ms	5.895794147s	61	705.47	0.34%	1.48%
55 (TR)	3	0.2	[18 29 3]	592.822144	606.512181	1.836277605s	17.659492552s	91	592.64	0.03%	2.34%
		0.4	[3 29 33]	684.681932	692.667724	1.767039395s	15.959858352s	80	684.27	0.06%	1.23%
		0.8	[3 29 0]	856.008716	873.481437	1.885503393s	17.737355999s	117	853.35	0.31%	2.36%
	4	0.2	[25 26 18 3]	503.650361	528.34163	1.901046037s	17.836593404s	109	501.74	0.38%	5.30%
		0.4	[3 25 33 31]	627.534671	645.020912	2.051763545s	18.399397752s	109	618.13	1.52%	4.35%
		0.8	[31 3 0 29]	815.969848	831.035243	1.990310501s	18.890493298s	131	812.37	0.44%	2.30%
										0.26%	2.20%

Table 18: Genetic Algorithm TR Results (Population: 300, generations: 200)