H-ACO: A Heterogeneous Ant Colony Optimisation approach with Application to the Travelling Salesman Problem

A.T.I Fayeez[1], E.Keedwell[1], M.Collett[2]

[1]College of Engineering, Mathematics and Physical Sciences, Harrison Building, University of Exeter
[2]Animal Behaviour Laboratory, College of Life and Environmental Sciences, Washington Singer Building, University of Exeter
{ab835, E.C.Keedwell, M.Collett}@exeter.ac.uk

Abstract. Ant Colony Optimization (ACO) is a field of study that mimics the behaviour of ants to solve computationally hard problems. The majority of research in ACO focuses on homogeneous artificial ants although animal behaviour research suggests that heterogeneity of behaviour improves the overall efficiency of ant colonies. Therefore, this paper introduces and analyses the effects of heterogeneity of behavioural traits in ACO to solve hard optimisation problems. The developed approach implements different behaviour by introducing unique biases towards the pheromone trail and local heuristic (the next hop distance) for each ant. The well-known Ant System (AS) and Max-Min Ant System (MMAS) are used as the base algorithms to implement heterogeneity and experiments show that this method improves the performance when tested using several Travelling Salesman Problem (TSP) instances particularly for larger instances. The diversity preservation introduced by this algorithm helps balance exploration-exploitation, increases robustness with respect to parameter settings and reduces the number of algorithm parameters that need to be set.

Keywords. Heterogeneity, Heterogeneous, ACO, TSP

1 Introduction

Natural systems provide inspiration for tackling complex tasks by being able to self-organize without the need of a central controller. These behaviours are due to evolution, development and learning thus providing a platform for nature-inspired algorithms to achieve good solutions to complex problems. Another example of such inspiration is the swarm behaviour in which natural organisms behave when they are in groups. As an example, ant collectives are capable of achieving complex tasks such as nest construction and food foraging that would not be possible for individual ants. A colony of ants is capable of finding the shortest path from nest to food in a sophisticated way. Inspiration can be taken from these behaviours and used to tackle optimization problems in the real world where their behaviours have been implemented in ant colony optimization (ACO) research [1]. The main contribution of ants in this research is the foraging behaviour where ants lay pheromone on the ground to mark their path from the nest to food source. This is to guide the ant back to the nest and also guide its colony members towards a food source during the recruitment process. ACO implements similar concepts when optimising combinatorial optimization problems such as the Travelling Salesmen Problem (TSP) [2]. TSP is one of the most widely studied by researchers working on combinatorial optimization problems, is an NP-hard problem and is an interpretation of a salesman requiring to visit *n* cities via the shortest complete tour.

A significant issue with ACO, as in most other metaheuristic approaches is to find a proper balance between exploitation and exploration. Exploitation is a process of concentration of the algorithm in the areas of the search space where good quality solutions have been previously been found while exploration of the search space denotes action by the search agent in moving towards unexplored areas. Several studies reviewed in [1] show that a proper balance between exploitation and exploration is required in order for a metaheuristic algorithm to achieve good to optimal results. In this paper, we investigate the influence of each ant having different behavioural characteristics or ‘traits’ in contrast to standard ACO where all ants have the same behavioural traits. In the proposed heterogeneous approach, each ant has individual pheromone (α) and heuristics coefficients (β) where both α and β are parameters that control the relative importance of the pheromone trail and local heuristics used in transition probability [3]. It is known that too much emphasis on pheromone trail or local heuristics may hinder the performance of the algorithm through over exploration or exploitation. Hence the proposed method can overcome the exploration-exploitation problem thus improving the performance of ACO. The heterogeneous approach implemented in this study stems from the actual behaviour of social insects which are heterogeneous in nature, displaying different traits and in some circumstances behavioural roles within a colony [4] [5]. The paper is structured as follows. In Section 2, ACO is discussed briefly while Section 3 discuss the previous work on heterogeneous approach in ACO. Section 4 describes the methodology of this study and Section 5 explains the experimental setup. Section 6 present the results of the study and the paper is concluded in Section 7 with discussion and conclusion.

2 Ant Colony Optimization

ACO is an optimisation algorithm that takes inspiration from the foraging behaviours of real ants. Some of the most popular conventional ACO are Ant System (AS) [3] and Max Min Ant System (MMAS) [6] that use metaheuristics approach inspired by ant colonies behaviour to find good solutions for an optimization problem. AS was the first ACO algorithm to be developed and acts as proof of ACO concept while MMAS is one of the best performing ACO algorithms in the literature. Both algorithms work through the deposition of pheromone by virtual ants who traverse the set of cities creating a tour, where the level of pheromone deposited on that tour is a function of tour optimality and the pheromone on all paths is evaporated uniformly. Subsequent ants probabilistically choose paths with a preference for those paths with greater pheromone with the goal of converging towards a near-optimal solution. The algorithms differ in that ant system allows all ants to contribute to the deposition of the pheromone, whereas the max-min ant system allows only the best performing ant within a population to contribute and has a lower-bound on pheromone levels. Both AS and MMAS have been applied to numerous TSP instances, a combinatorial optimization problem that has attracted extensive research [7]. This paper implements the heterogeneous approach on these two ACO variants. Due to limited space, AS and MMAS will not be discussed in detail here and can be referred to [3] and [6] respectively.

3 Heterogeneous ACO

Heterogeneity in swarm intelligence was firstly described in Particle Swarm Optimization (PSO) by Engelbrecht in [8] who proposed that introduction of heterogeneity in a search algorithm can improve the performance. This concept can also be adopted in ACO where artificial ants with different traits of behaviour can help to improve the performance of the ACO algorithm. This mimics the actual behaviour of real ants in a colony in terms of diversity and division of labour [9]. Heterogeneity in ACO can be grouped into *individual* and *colony* level. Artificial ants with different 'behaviours' among them is said to be heterogeneous at the individual level while colonies of ants that differ in behaviour between the colonies is said to be the latter. Heterogeneous individual ants in ACO were first introduced by [10] where the authors used modified ACO with heterogeneity for path planning in mobile robots in order to find obstacle-free path in a certain environment. The author deployed ants with different sight, speed and function behaviours and found that the performance of Heterogeneous ACO (HACO) is better in terms of path planning when compared to conventional ACO. Chira et al. discussed the different sensitivity of the artificial ants to the pheromone trail level in [11]. Ants with higher pheromone sensitivity strongly follow the pheromone trail while ants with lower pheromone sensitivity are more inclined towards random search. In the meantime, Hara et al. [12] proposed the use of classic and exploratory ants where each ant constructs a partial solution which is then combined to produce one single solution. Yoshikawa et al [13] introduces a cranky ant approach to tackle the exploration-exploitation problem which appears to prevent the algorithm from being stuck in local optima. The cranky ants will explore paths with low pheromone level which is the opposite of the behaviour of standard artificial ant. Meanwhile, Zhang et al. [14] proposed colony level heterogeneity where ant colonies have different pheromone updating rules in order to balance exploration and exploitation in the search process. The authors proposed two colonies where each exhibits behaviour of Elitist Ant System (EAS) and Ant Colony System (ACS) characteristics respectively. They discussed that the algorithm overcomes stagnation and the early suboptimal path convergence problem. Melo et al [15] proposed a multi-caste ant colony in Ant Colony System (ACS) where ants with different preference towards *q0,* parameter that controls the degree of exploration or exploitation in ACS. Many more approaches implement heterogeneity at the colony level, but as this paper study and implementation at individual level, thus colony level heterogeneity will not be discussed in detail here. Each of these algorithms approach the principle of heterogeneity from a different standpoint, either using different ant roles or through the implementation of problem specific heterogeneity. The approach taken in this paper is one of biological plausibility for ants with similar roles, but differing behavioural traits, which would normally be expressed through genetic differences, but here are drawn from a distribution.

4 Methodology

The main motivation of this research work is to study the ant colonies as heterogeneous, multi-behaviours agents that can further improve the performance of the algorithm. The hypothesis is that with heterogeneity, a mixture of ants that are more inclined towards exploration of the search space with other ants that exploit the best path found creates a balance in the search process. This is due to the behaviours of the ants of which are randomly initialized either to be more inclined towards exploration or exploitation. The algorithm proposed a simple heterogeneous method in this study by pre-assigning a random behavioural trait for each of the ants in the population size during initialization that will not change during the iterations, as would be the case with genetic variation in real ants. Each behaviour has a pair of continuous traits that can be related to pheromone trail intensity and visibility or the local heuristic information. The heterogeneous approach in both AS and MMAS platform and comparison were carried out and compared with the original versions of each algorithm.

Algorithm 1: Heterogeneous ACO for TSP

```
1. Begin;
2. Input: Distance Matrix of TSP;
3. Initialize: Max Iteration, m, n, Q, τ, ρ;
4. Initialize ants:
     For i = 1: m
      a=0; b=2;
      Alpha (i) = rand (1) * (b-a) + a;

      c=3; d=5;
      Beta (i) = rand (1) * (d-c) + c;
     End
5. Start Iteration:
      For it=1: Max Iteration
        For k=1: m
        Position each ant on starting node;
           While TourSize < n+1
              Tour Construction;
           End
        End
       End
6. Update solution;
7. Update Pheromone;
8. Pheromone Evaporation;
9. Check if termination criteria = true;
10. End
```

Algorithm 1 depicts the pseudocode of our proposed algorithm and the major difference between this and the base algorithm is that ants will have α and β values that are initialized randomly between a set of pre-determined values rather than identical parameters throughout the run. The range for α and β values were based on experiments by Dorigo et al in [3] and additional extensive experiments have been conducted to determine the best range for α and β (discussed briefly in section 6.1).

4.1 Travelling Salesman Problem [2]

Travelling Salesman Problem (TSP) is a widely studied combinatorial optimization problem in computer science. The main objective of solving a TSP is to achieve the shortest tour visiting n cities and returning to the starting city when there are no more cities left to be visited. TSP is visualized in a graphical format where nodes act as the cities and edges as the link or path between the cities. The edges will have weighting determining the cost of following that edge. TSP has been a popular case study in ACO and other various optimization algorithms.

5 Experimental Setup

The experiments were conducted on an Intel Core i7 CPU-based computer running Windows 7 equipped with 4GB RAM. The base algorithms used are the Ant System (AS) and Max Min Ant System (MMAS) approach developed using the Matlab version R2015a. Each algorithm is tested using several TSP instances taken from TSPLIB [2]. Firstly, the developed AS and MMAS was compared with that of [3] and [6] to show a level of confidence that the developed algorithm is similar to the original version. All the parameters were set according to the authors' recommendations where for AS the parameters were set as follows: α=1, β=5, ρ=0.5 and m = n where m is the number of ants and n is the number of cities related to the TSP. Meanwhile the parameters for MMAS were set as follows: α = 1, β = 2, m = n, ρ = 0.98 and P_{best} is 0.05. The function evaluations for all the experiments were set as *k.n.10000* where k=1 for symmetrical TSPs used, n=number of cities of the TSP instance and 10 000 is the maximum number of iterations. Table 1 shows the comparison between the developed algorithms against its original versions where the results for the developed algorithms are the average of 15 trials. As can be seen, the best cost of the developed algorithm and that of the original developers' are very similar demonstrating that the base algorithm formulations are working appropriately.

Table 1: Developed AS, MMAS vs Original AS, MMAS. Results show the average of the best cost. (*Note: Average of 15 trials*)

TSP	Optimum (Integer Length)	Optimum (Real Length)	Developed A.S	AS	Developed MMAS	MMAS [6]
Oliver 30	420[2]	423.741 [3]	423.7406	423.74 [3]	N.A	N.A
Eil51	426 [3]	428.87 [2]	437.56	437.3 [6]	427.5	427.1
kroA 100	21285 [3]	21285.44 [2]	22451.9819	22471.4 [6]	21299.6	21291.6
D198	15780 [3]	15808.65 [2]	16692.24	16702.1 [6]	15960.2	15956.8

6 Heterogeneous ACO Results

6.1 Exploring the ranges of Alpha and Beta

An extensive experiment based on AS was conducted to find the best range of α and β for our heterogeneous approach where lower and upper bounds of α and β were based on the recommendation of [3]. Both α and β values are varied to create a heterogeneous approach as both α & β plays an important role in exploration and exploitation of the search space. Hence, varying both parameters will introduce more variance in the agents. In addition, Stützle et al [16] suggests that both α and β are good candidates for parameter adaptation in ACO. As can be seen, the recommended range for α is between 0.25 and 1.5 while β has a range of 1 to 5. Therefore, extensive experiments were conducted where the ants were set to have a uniform distribution of α between 0 and 1 and 0 to 2 while a uniformly distributed β was varied between 0 and

5, narrowed down to 4 to 5. The other parameters were set according to [3]: 10 000 iterations, m = n, ρ = 0.5, Q = 100, initial pheromone trail = m/L_{nn} where L_{nn} is the tour length of the tsp instance using nearest neighbour heuristic. 3 tsp instances were used to test the algorithm namely oliver30.tsp (integer length optimum = 420, real length optimum = 423.7406), eil51.tsp (integer length optimum = 426, real length optimum = 428. 8716) and eil101.tsp (integer length optimum = 629). Table 2 and Table 3 summarizes the outcome of our extensive experiment. The results are best tour length found in 15 trials.

Table 2: Results from experimentation where α is uniformly distributed between 0 to 1 and the β distribution varies. Algorithm tested on oliver30.tsp, eil51.tsp and eil101.tsp. Results represent average best cost out of 15 trials while values in bold represents the best average.

α	β	oliver30	eil51	eil101
0 -1	0 -5	427.0934	445.301	699.1238
0 -1	1 -5	425.3379	441.6734	685.7444
0 -1	2 -5	426.0892	439.5271	678.2238
0 -1	3 -5	**423.7406**	**436.2947**	661.9443
0- 1	4 - 5	**423.7406**	436.3278	**659.4744**

Table 3: Results from experimentation whereα is uniformly distributed between 0 and 2 and the β distribution varies. Algorithm tested on oliver30.tsp, eil51.tsp and eil101.tsp for 10 000 iterations with values representing average best cost out of 15 trials while values in bold represents the best average.

α	β	oliver30	eil51	eil101
0- 2	0 -5	427.2749	437.1203	688.2972
0 -2	1 -5	424.6639	442.3749	672.3319
0 -2	2 -5	423.9117	438.0173	665.7093
0 -2	3 -5	**423.7406**	**436.0904**	**645.5318**
0- 2	4- 5	**423.7406**	436.6167	651.2821

Both Table 2 and Table 3 above show that the best range is α: 0 to 2 and β: 3 to 5. The experiment did not include α values greater than 2 because it is proven can lead to stagnation behaviour [3]. Therefore, the following experiments related to heterogeneous AS hereafter will use this parameter range.

6.2 Comparison with Base Algorithms

Next, the Heterogeneous Ant System (HAS) was compared against AS developed by [3] based on several symmetrical tsp instances. The AS (and later MMAS [6]) systems have been subjected to extensive experiments to determine the optimal alpha and beta settings for these problems. The resulting comparisons are therefore made between the heterogeneous system and well-tuned examples of the base ACO algorithms. HAS has the same parameter settings as AS (mentioned in the previous section) expect that α is a uniform distribution between 0 and 2 while β is varied from 3 and 5. The function evaluations for all experiments remain the same as previous section. Table 4 summarizes the comparison of AS and HAS on eil51.tsp for 25 trials. It can be seen that HAS improves on the best cost found by AS where the average is 436 compared to that of AS which is 437.56. Both AS and HAS was not able to find the optimum but it is shown that HAS performs better than AS in terms of 1% deviation and 2% deviation of the optimum. A value is said to be 1% deviation of optimum when it is within the range of 1% to the optimum. In eil51.tsp case, 1% deviation is 1/100 x 426 (optimum value from TSPLIB [2]) =4.26+426 =430.26.

426 < X< 430.26 = 1% deviation of optimum

430.26 < X < 434.52 = 2% deviation of optimum

Table 4: Best, average and worst cost comparison between AS & HAS for eil51.tsp, 10 000 iterations over 25 trials.

Method	Best	Average	Worst	# Optimum Found	1% dev of opt	2% dev of opt
AS	433	437.56	**441**	0	0	1
HAS	**428**	**436.00**	442	0	**1**	**5**

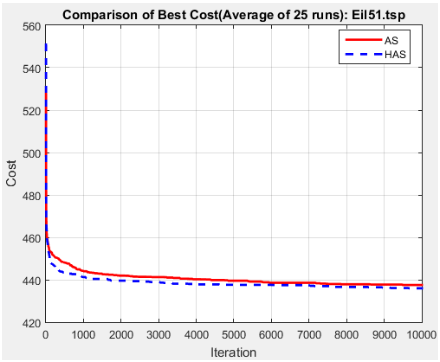


Figure 2: Comparison of average best cost for 25 independent trials of eil51.tsp with 10 000 iterations for each trial.

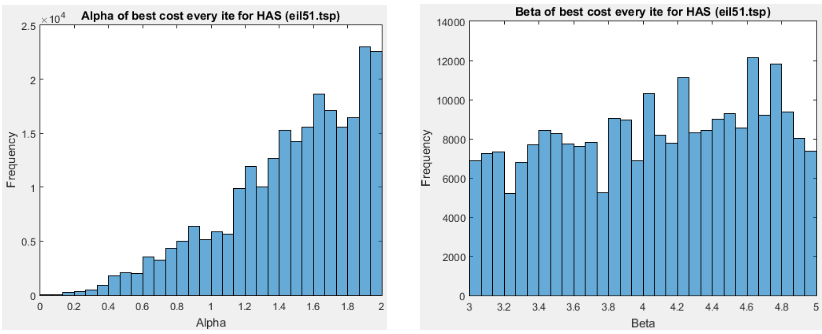


Figure 3 (a) Figure 3 (b)

Figure 3: Frequency of Alpha & Beta values that managed to find the best cost in every iteration for HAS (eil51.tsp) (*Note: 25 trials x 10 000 iterations each trial = 250 000 iterations*)

Figure 2 shows that the Heterogeneous Ant System (HAS) has a better performance in terms of average best cost compared to AS over the duration of the optimisation. Figure 3 show the frequency of alpha and beta values of ants that found the best cost in every iteration. It can be seen the alpha values that mostly contribute are between 1.9 and 2, with a strong skew towards these values whereas the beta distribution is much more uniform with a small skew towards beta values of 4.6 and 4.75. This shows that heterogeneous approach introduces diversity in the algorithm and suggests the mechanism behind the improved performance over the algorithm with a single 'behavioural trait'.

Table 5: Best, average and worst cost comparison between AS & HAS for kroA100.tsp (optimum: 21282). Results in bold are the best in the table.

Method	Best	Average	Worst	# Optimum Found	5% dev of opt	6% dev of opt
AS	22384	22469.4	22666	0	0	5
HAS	**22215**	**22347.6**	**22487**	0	**22**	**25**

Table 5 shows the comparison between AS and HAS for 100-city tsp, kroA100.tsp. HAS managed to improve on the fitness solution compared to AS where average best cost for HAS is 22347.6 and that of AS is 22469.4. Although both AS and HAS did not manage to find the optimum for 100-city TSP problem, HAS managed to find a best cost that is within 5% of the optimum 22 times compared to none by AS. In addition, HAS found a best cost of 22215 compared to 22384 of AS out of 25 trials.

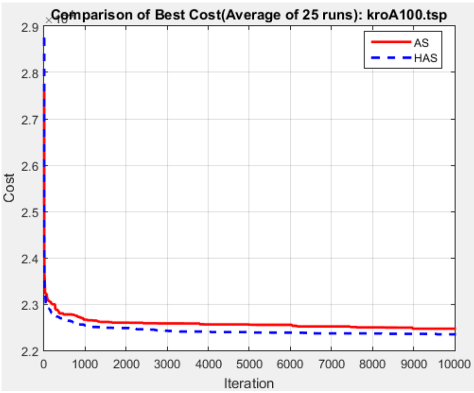


Figure 4: Average best cost comparison for AS & HAS (kroA100.tsp) for 10 000 iterations over 25 trials.

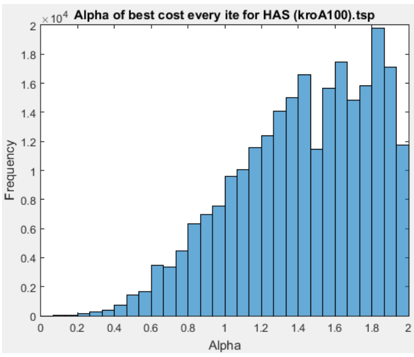


Figure 5 (a)

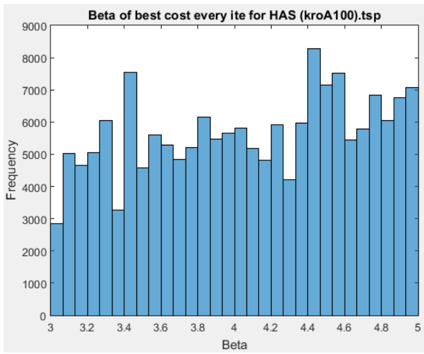


Figure 5 (b)

Figure 5: Frequency of Alpha & Beta values that managed to find the best fitness solution in every iteration for HAS (kroA100.tsp) (*Note: 25 trials x 10 000 iterations each trial = 250 000 iterations*)

Figure 4 shows the improved performance of HAS over AS in terms of average best cost while Figure 5 (a) and 5 (b) show the frequency of alpha and beta values of ants that managed to find best cost in all the 10 000 iterations for 25 trials. The distributions are similar to those of the previous experiments with alpha values peaking at 1.85 while beta has a peak at 4.45.

Table 6: Best, average and worst cost comparison between AS & HAS for d198.tsp (Optimum: 15780). Results in bold are the best in the table.

Method	Best	Average	Worst	# Optimum Found	3% dev of opt	4% dev of opt
AS	16356	16572.48	16724	0	0	3
HAS	**16186**	**16359.04**	**16700**	0	**6**	**19**

Table 6 summarizes the outcome of 25 trials of d198.tsp using both AS and HAS. AS found a best cost of 16356 throughout the 25 trials while HAS found a best cost of 16186. In additionHAS has a lower average compared to AS. Although the optimum is not found by any of the algorithms, HAS managed to find fitness solutions that are 3%

within the optimum range 6 times and 19 times within 4% of the optimum compared to 0 and 3 times respectively by AS.

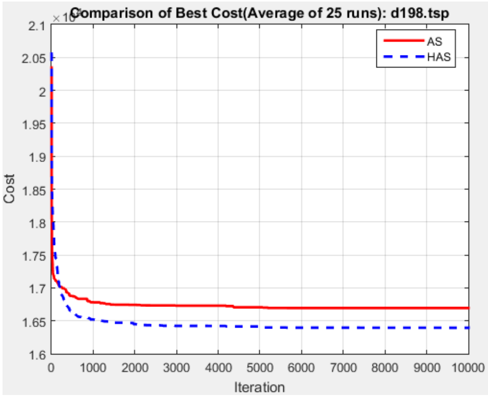


Figure 6: Average of best cost comparison between AS & HAS (d198.tsp) over 25 trials, 10 000 iterations in each trial.

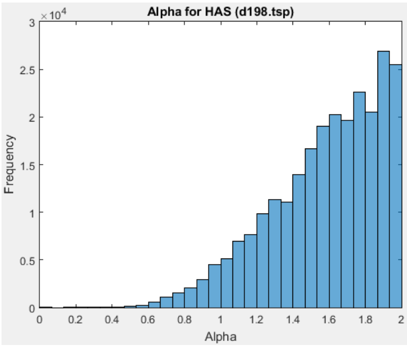


Figure 7 (a)

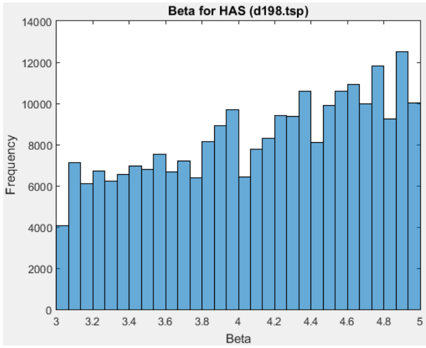


Figure 7 (b)

Figure 7: Frequency of Alpha & Beta values that found the best fitness solution in every iteration for HAS (d198.tsp) (*Note: 25 trials x 10 000 iterations each trial = 250 000 iterations*)

Figure 6 shows the major improvement in terms of average best cost performance of HAS over AS while it can be seen clearly in Figure 7 that even though both α=1 and β=5 as per suggested in [3] are covered in the pre-determined range for heterogeneous approach, both α and β values that managed to find best cost in every iteration increases rapidly from 0.5 to 2 and a steady increase from 3 to 5 respectively with α values having a peak at 1.9 while β values have a peak of 4.9.

The encouraging results of the heterogeneous approach on Ant System leads to the approach to be implemented on to Max Min Ant System (MMAS) known as Heterogeneous MMAS (HMMAS). All the parameters were set according to [6](discussed in experimental setup) except that of α which was set to 0 to 2 while β is varied between 1 and 3. The same sets of TSP instances were used to compare HMMAS against MMAS. Table 7 summarizes the comparison for eil51.tsp which has an optimum of 426. Although overall average of HMMAS is slightly higher compared to that of MMAS, HMMAS performed much better in relation to the number of times optimum found where both MMAS and HMMAS managed to find the optimum 4 times and 10 times respectively out of 25 trials. Figure 11 shows the comparison of the average best cost of both MMAS and HMMAS for the 51-city TSP problem.

Table 7: Best, average and worst cost comparison between MMAS & HMMAS for eil51.tsp (optimum: 426). Results in bold represents the best value in the table.

Method	Best	Average	Worst	# Optimum Found	1% dev of opt	2% dev of opt
MMAS	**426**	**427.4**	**430**	4	**25**	**25**
HMMAS	**426**	427.6	431	**10**	23	**25**

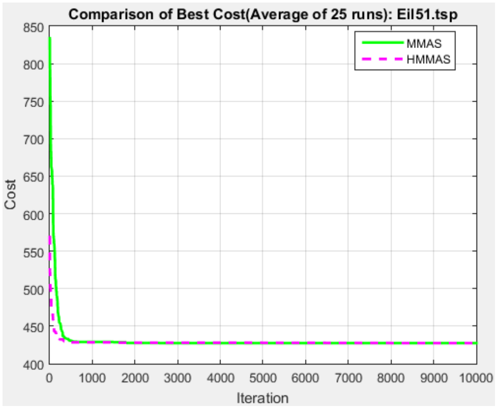


Figure 8: Comparison of average best cost for MMAS & HMMAS (eil51.tsp) over 25 trials, each trial = 10 000 iterations.

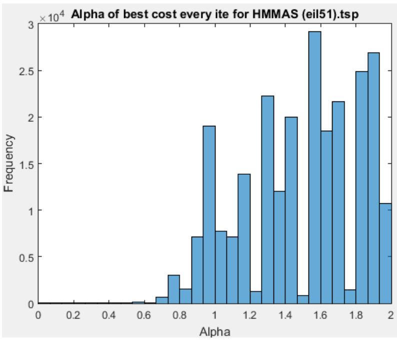


Figure 9 (a)

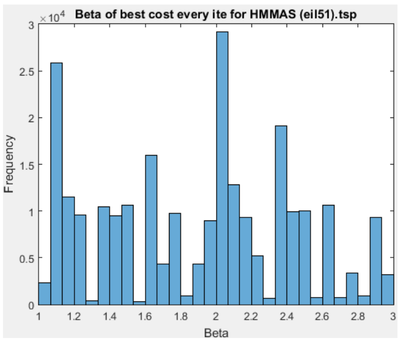


Figure 9 (b)

Figure 9: Frequency of Alpha & Beta values that managed to find the best fitness solution in every iteration for HMMAS (eil51.tsp) (*Note: 25 trials x 10 000 iterations each trial = 250 000 iterations*)

Figure 8 shows that both MMAS and HMMAS have a similar average best cost. Figure 9 illustrates that both the alpha and beta values of ants that managed to find the best cost in every iteration for HMMAS with alpha has a peak value of 1.55 while beta has a peak of 2.05. The overall distributions are somewhat similar to those from HAS. The diversity in the algorithm helps too as it shows that various alpha and beta values contribute towards finding the best cost.

Table 8: Best, average and worst cost comparison between MMAS & HMMAS for kroA100.tsp (optimum: 21828). Results in bold represent the best value in the table.

Method	Best	Average	Worst	# Optimum Found	1% dev of opt	2% dev of opt
MMAS	**21282**	**21299.6**	21390	4	**25**	**25**
HMMAS	**21282**	21316.6	**21379**	**11**	21	22

Table 8 shows the outcome of experiment on kroA100.tsp where both MMAS and HMMAS managed to find the optimum of 21282 while MMAS has an average of

21294.4 and HMMAS has an average of 21316.6. This can be due to several trials producing fitness solutions out of the 1% and 2% range of optimum thus causing the HMMAS to have a higher average. Although MMAS have a lower average best and lower worst cost, HMMAS still outperforms MMAS by finding the optimum 11 times compared to 4 times for MMAS. Figure 10 shows the comparison of the average best cost between MMAS and HMMAS.

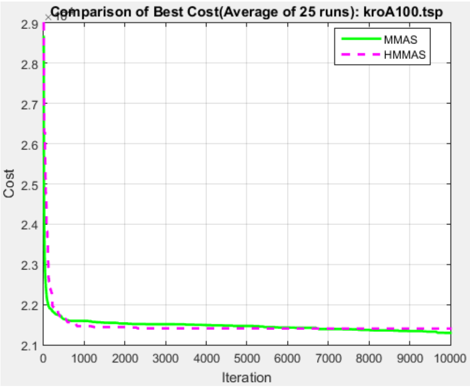


Figure 10: Comparison of MMAS & HMMAS of average best cost (kroA100.tsp) over 25 trials, each trial = 10 000 iterations.

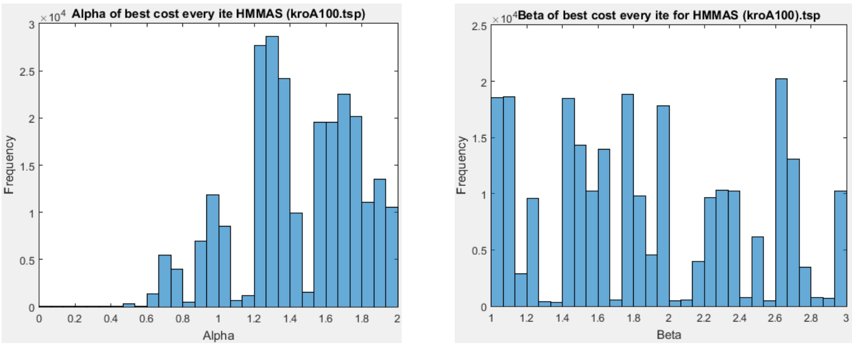


Figure 11 (a) Figure 11 (b)

Figure 11: Frequency of Alpha & Beta values that managed to find the best fitness solution in every iteration for HMMAS (kroA100.tsp) (*Note: 25 trials x 10 000 iterations each trial = 250 000 iterations*)

Figure 10 shows that the average best cost of MMAS is slightly better compared to HMMAS for 100-city problem. Both Figure 8 and Figure 10 suggest that MMAS performs considerably well for eil51.tsp and kroA100.tsp due to the small problem size. Figure 11 illustrates the alpha and beta values related to the best cost in every iteration over 25 trials. Alpha values peak around 1.3 and beta has a peak of 2.65 respectively.

Table 9: Best, average and worst cost comparison between MMAS & HMMAS for d198.tsp (optimum: 15780). Results in bold represents the best value in the table.

Method	Best	Average	Worst	# Optimum Found	1% dev of opt	2% dev of opt
MMAS	15846	15961.12	16137	0	10	22
HMMAS	**15795**	**15871.68**	**16006**	0	**21**	**25**

Table 9 summarizes the comparison made between MMAS and HMMAS for 198-city TSP. HMMAS has a best cost of 15795 compared to 15846 of MMAS and HMMAS also has a lower average and lower worst cost compared to MMAS. Meanwhile, HMMAS also managed to find fitness solutions 21 times within the 1% range of optimum compared to that of MMAS of 10 times.

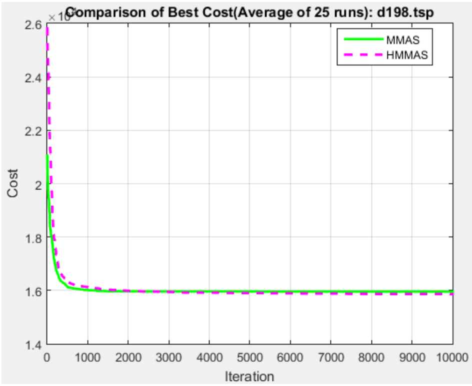
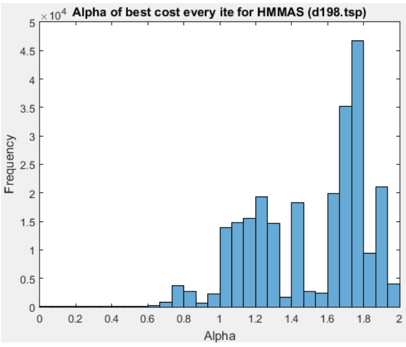


Figure 12: Comparison of MMAS & HMMAS of average best cost (d198.tsp) over 25 trials, each trial = 10 000 iterations.



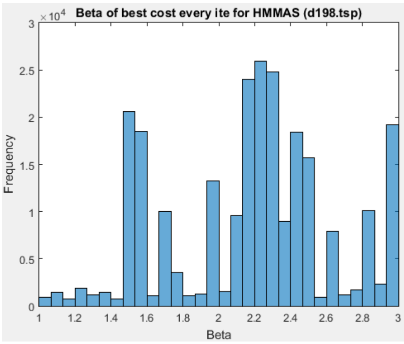


Figure 13 (a) Figure 13 (b)

Figure 13: Alpha & Beta values that managed to find the best fitness solution in every iteration for HMMAS (eil51.tsp) (*Note: 25 trials x 10 000 iterations each trial = 250 000 iterations*)

Figure 12 shows the comparison of the average best cost between MMAS and HMMAS over 25 trials for eil51. HMMAS have a better average best cost compared to MMAS in a medium-sized tsp. Figure 13 shows the alpha and beta values with a peak of 1.7 and 2.2 respectively.

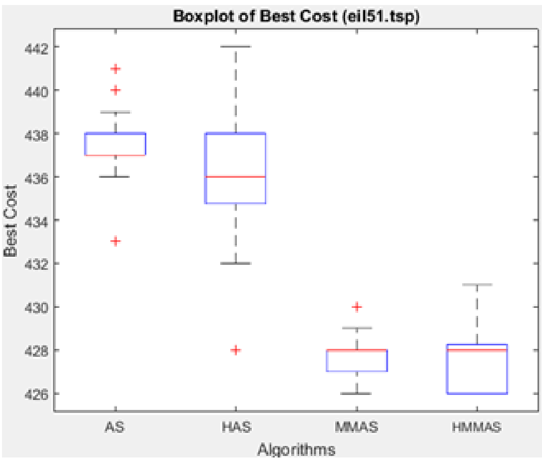


Figure 14: Boxplot of best cost for 25 independent trials of 4 different algorithms namely AS, HAS, MMAS and HMMAS for eil51.tsp. Each trial were conducted for 10 000 iterations.

Figure 14 shows that both HAS and HMMAS have a better performance compared to its base algorithm in terms of best cost found in each of the 25 independent trials for eil51.tsp. HAS has a lower median and lower inter-quartile (IQR) values compared to AS. Furthermore, HMMAS has a worst cost larger than MMAS, but more of the best costs are at the optimum of 426 for eil51.tsp.

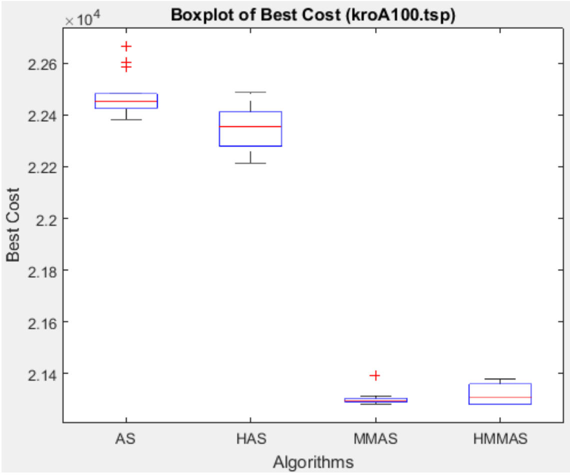


Figure 15: Boxplot of best cost for 25 independent trials of 4 different algorithms for AS, HAS, MMAS and HMMAS for kroA100.tsp.

Figure 15 shows the boxplot for the best cost found by all 4 algorithms in test in each of the 25 trials. It can be seen that HAS has a better performance compared to AS in terms of best cost with a lower median as well. On the other hand, HMMAS has a slightly higher median compared to MMAS. It can also be observed from Figure 14 that both HAS and HMMAS have a larger IQR and this can be attributed to the variance in terms of best cost found caused by the heterogeneous approach introduced.

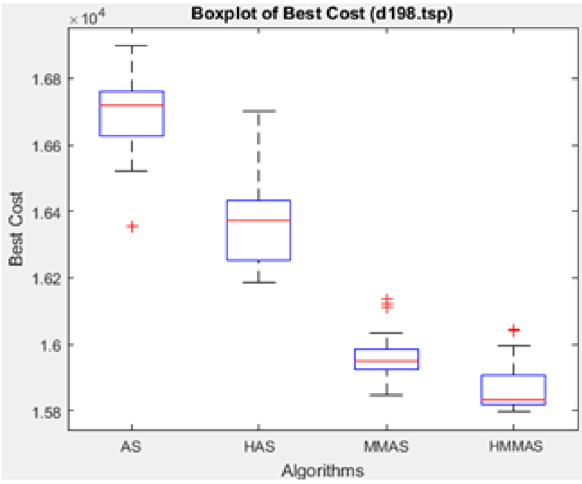


Figure 16: Boxplot of best cost for 25 independent trials of 4 different algorithms namely AS, HAS, MMAS and HMMAS for d198.tsp.

Figure 16 shows the improvement of HAS and HMMAS over its base algorithms. Both heterogeneous algorithms have a lower median compared to AS and MMAS. In both cases, the improvements are statistically significant thus the algorithms clearly benefitting from the heterogeneous approach.

Table 10: p-values and z-values of Wilcoxon rank sum test for best cost of HAS and HMMAS against its respective base algorithm.

TSP	HAS vs AS	HMMAS vs MMAS
eil51	0.0143	0.8796
kroA100	2.03E-0.6	0.3078
d198	2.87E-08	1.27E-04

A two-tailed Wilcoxon rank sum test with confidence level of 95% was conducted for HAS against AS and HMMAS against MMAS with $p < 0.05$ as the threshold level where the difference is significant. The table above shows that the best cost found by HAS for the 25 trials are significantly better when compared to AS for all the three instances. The test shows that the best cost of HMMAS is not significant over its base algorithm for eil51 and kroA100. First of all, these two tsp instances fall under the category of small instance problem where even the base algorithm performs moderately. Secondly, the effect of individual variance or heterogeneity is limited in HMMAS due to algorithm's limitation of only a single agent to modify the pheromone limiting the overall heterogeneity advantage. Furthermore the performance of the base algorithm MMAS is clearly superior to that of AS meaning that it is also more difficult to for heterogeneity to show an improvement. However, despite this, HMMAS is statistically significant when compared to MMAS in terms of best cost found for d198.tsp.

7 Discussion, Conclusion & Future Work

In summary, a heterogeneous ACO has been introduced which implements artificial ants that have different 'behavioural traits' compared to the traditional homogeneous approach. This computational work in ACO is in relation to the biological aspect of real ants where ants are known to have diversity in their population. The results clearly show that the heterogeneous approach in ACO produce improved performance over the standard, parameter tuned algorithms on which they are based. The performance difference was particularly marked when implemented on Ant System. This is likely to be due to the greater contribution of each ant to the pheromone trail, highlighting the effect of diversity. The smaller gains made with HMMAS can be explained by the increased performance of the base algorithm, locating solutions closer to the optimum and also that only the best ant contributes to the pheromone update reducing the effect of population diversity on algorithm progression.

The implemented approach, by varying the alpha and beta values shows that even though prior work [3] suggests a range of optimal α and β values to choose from, determining a certain value is not easy as the parameters are problem-dependant. The results here show that the heterogeneous approach is able to overcome this problem by being robust to parameter settings by effectively exploring the parameter space in conjunction with optimising the problem. Having a variety of 'behavioural traits' rather than a single behaviour shows the advantage in the performance of the algorithm. Recording the best performing alpha and beta values provides some support for the parameter values suggested by both Dorigo [3] and Stützle [6], but also highlighted instances where these parameter settings were not optimal. The discovery of distinct distributions of parameter settings for alpha and beta is interesting and demonstrates the algorithms' sensitivity to these parameters. These distributions remained stable despite being tested on multiple problem sizes. The work here has explored the hypothesis that heterogeneity is able to improve the performance of an algorithm and the results have gone some way to showing that heterogeneity applied to ACO can improve performance on the TSP and robustness to parameter settings. The next focus is on implementing Gaussian distribution towards heterogeneity and greater biological plausibility.

Acknowledgement

We would like to thank the Faculty of Electronics and Computer Engineering (FKEKK), Technical University of Malaysia Malacca (UTeM) and the Ministry of Higher Education (MoHE) Malaysia for the financial support under the SLAB/SlAI program.

References

[1] Christian Blum, “ACO Applied to Group Shop Scheduling: A Case Study on Intensification and Diversification,” in *ANTS ’02 Proceedings of the Third International Workshop on Ant Algorithms*, 2002, pp. 14–27.

[2] G. (Gerhard) Reinelt and Gerhard, *The traveling salesman : computational solutions for TSP applications*. Springer-Verlag, 1994.

[3] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant system: Optimization by a colony of cooperating agents,” *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 26, no. 1, pp. 29–41, 1996.

[4] A. P. Modlmeier and S. Foitzik, “Productivity increases with variation in aggression among group members in Temnothorax ants,” *Behav. Ecol.*, vol. 22, no. 5, pp. 1026–1032, Sep. 2011.

[5] M. Collett and T. S. Collett,"*Spatial aspects of foraging in Ants and Bees*," Cold Spring Harbor Monograph Series *49*, pp 467 - 502, 2007.

[6] T. Stutzle and H. Hoos, “MAX MIN Ant System and Local Search for the Traveling Salesman Problem,” *Ieee Int. Conf. Evol. Comput.*, pp. 309–314, 1997.

[7] G. Gutin and A. P. Punnen, Eds., *The Traveling Salesman Problem and Its Variations*, vol. 12. Boston, MA: Springer US, 2007.

[8] A. Engelbrecht, “Heterogeneous particle swarm optimization,” *Swarm Intell.*, pp. 191–202, 2010.

[9] O. Blight, G. Albet Díaz-Mariblanca, X. Cerdá, and R. Boulay, “A proactive–reactive syndrome affects group success in an ant species,” *Behav. Ecol.*, vol. 27, no. 1, pp. 118–125, Jan. 2016.

[10] J. W. Lee and J. J. Lee, “Novel ant colony optimization algorithm with path crossover and heterogeneous ants for path planning,” in *Proceedings of the IEEE International Conference on Industrial Technology*, 2010.

[11] C. Chira, D. Dumitrescu, and C. M. Pintea, “Heterogeneous sensitive ant model for combinatorial optimization,” *Genet. Evol. Comput.*, p. 163, 2008.

[12] A. Hara, S. Matsushima, T. Ichimura, and T. Takahama, “Ant colony optimization using exploratory ants for constructing partial solutions,” in *2010 IEEE World Congress on Computational Intelligence, WCCI 2010 - 2010 IEEE Congress on Evolutionary Computation, CEC 2010*, 2010.

[13] M. Yoshikawa, “Adaptive Ant Colony Optimization with Cranky Ants,” Springer Netherlands, 2009, pp. 41–52.

[14] P. Zhang and J. Lin, “An adaptive heterogeneous multiple ant colonies system,” in *Proceedings - 2010 International Conference of Information Science and Management Engineering, ISME 2010*, 2010.

[15] L. Melo, F. Pereira, and E. Costa, “Extended experiments with ant colony optimization with heterogeneous ants for large dynamic traveling salesperson problems,” *Proc. - 14th Int. Conf. Comput. Sci. Its Appl. ICCSA 2014*, pp. 171–175, 2014.

[16] T. Stutzle *et al.*, “Parameter Adaptation in Ant Colony Optimization IRIDIA – Technical Report Series Parameter Adaptation in Ant Colony Optimization,” 2010.