Ant Colony Optimisation and the Traveling Salesperson Problem – Hardness, Features and Parameter Settings

[Extended Abstract]

Samadhi Nallaperuma
School of Computer Science
The University of Adelaide
Adelaide, SA 5005, Australia

Markus Wagner
School of Computer Science
The University of Adelaide
Adelaide, SA 5005, Australia

Frank Neumann
School of Computer Science
The University of Adelaide
Adelaide, SA 5005, Australia

ABSTRACT

Our study on ant colony optimization (ACO) and the Travelling Salesperson Problem (TSP) attempts to understand the effect of parameters and instance features on performance using statistical analysis of the *hard*, *easy* and *average* problem instances for an algorithm instance.

Categories and Subject Descriptors

D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*; F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity

Keywords

Ant Colony Optimisation, Traveling Salesperson Problem, Features, Parameters

1. INTRODUCTION

Ant Colony Optimisation (ACO) is a recent heuristic algorithms approach that performs very well on NP-hard combinatorial optimisation problems. Although ACO is successful in practice, it is hard to understand when and why this type of algorithms work. Throughout the history of heuristic optimisation, attempts have been made to analyse ACO algorithm performance theoretically [3, 8] and experimentally [6, 10]. However, much less work has been done towards the goal of explaining the impact of the problem instance structure and the algorithm parameters on performance.

We study ACO on hard problems in a novel perspective based on problem hardness features. Our study presents hardness analysis for one of the most famous ACO algorithms, the Max-Min Ant System, on the well-studied Travelling Salesperson Problem (TSP). The aim of this research is to understand the impact of problem features and of algorithm parameters on the algorithm performance. An evolutionary algorithm approach similar to the one by Mersmann et al. [4] is utilised to generate easy and hard instances for different parameter settings. Statistical features of these instances are investigated in order to determine the impact of problem structure on performance for a particular algorithm instance with a specified configuration. Furthermore, hard and easy instances of algorithm instances with different parameter settings are compared in order to understand the impact of parameters on performance. With this understanding, we will contribute to ACO research in the areas of algorithm design and parameter prediction.

GECCO'13 Companion, July 6–10, 2013, Amsterdam, The Netherlands.
ACM 978-1-4503-1964-5/13/07.

2. PRELIMINARIES

The TSP is one of the most famous NP-hard combinatorial optimisation problems. Given a set of n cities $\{1, \ldots, n\}$ and a distance matrix $d = (d_{ij})$, $1 \leq i, j \leq n$, the goal is to compute a tour of minimal length that visits each city exactly once and returns to the origin.

In general, the TSP is not only NP-hard, but also hard to approximate. Therefore, we consider the still NP-hard Euclidean TSP, where cities are given by points in the plane and the Euclidean distance is used for distance computations.

This study is focusses on a well-known ACO algorithm called Max-Min Ant System [9]. Individual solution tours are constructed at each iteration by the set of ants considered in the algorithm. These tours are constructed by visiting cities sequentially, according to a probabilistic formula representing heuristic information and pheromone trails. Let us assume that ant k is in node i and needs to select the next city j to be visited. The applied selection formula is called random proportional rule and it is defined as $p_{ij} = [\tau_{ij}]^\alpha * [\eta_{ij}]^\beta / (\Sigma_{h \in N_k} [\tau_{ih}]^\alpha * [\eta_{ih}]^\beta)$. Here N_k represents the set of unvisited nodes of ant k, $[\tau_{ih}]$ and $[\eta_{ih}]$ having exponents α and β represent pheromone and heuristic information respectively. A detailed description and analysis of this algorithm on TSP can be found in the textbook of Dorigo and Stützle (Chapter 3) [2].

3. EXPERIMENTAL INVESTIGATIONS

We use an evolutionary algorithm to evolve easy and hard instances for the ant algorithms, which is similar to previous work done by Smith-Miles and Lopes [7] on the Lin-Kernighan heuristic, the work by Mersmann et al. [4] on the local search 2-opt, and the work by Nallaperuma et al. [5] on two approximation algorithms. Evolutionary algorithms are based to a large extent on random decisions. We use several runs of an algorithm to create a diverse set of hard and easy instances. The search is guided by the approximation quality of the solution. Hence by minimising and maximising the fitness it is possible to evolve easy and hard instances.

The approximation ratio $\alpha_A(I)$ of an algorithm A for a given instance I is defined as

$$\alpha(I) = A(I)/OPT(I)$$

where $A(I)$ is the tour length produced by algorithm A for the given instance I and OPT(I) is the value of an optimal solution of I. $OPT(I)$ is obtained by using the exact TSP solver Concorde [1].

Our experimental set up is as follows. For each parameter setting a set of random TSP instances is generated, and an evolutionary algorithm runs on them for 5000 generations to generate extreme sets of instances. In each iteration, the ACO algorithm is run several times on a single instance, and the average is taken and divided by the optimal solution to the obtain approximation ratio. This entire process is done for instances of sizes 25, 50, 100 and 200, and with the goal of generation easy and hard instances respectively.

The comprehensive feature set introduced by Mersmann et al. [4] is used for this study to analyse the hard and the easy generated TSP instances. These features include distances of edge cost distribution, angles between neighbours, nearest neighbour statistics, mode, cluster and centroid features as well as features representing minimum spanning tree heuristics and of the convex hull. The parameters considered in this study are the most popular and critical ones in any ACO algorithm, namely the exponents α and β that represent the influence of the pheromone trails and of the heuristic information respectively. We consider three parameter settings for our analysis (setting 1 with $\alpha = 1$ and $\beta = 2$, 2 with $\alpha = 0$ and $\beta = 4$ and 3 with $\alpha = 4$ and $\beta = 0$) The feature values for each instance set are analysed in detail with reference to the parameter setting it is based on. By comparing feature values of instances with different parameter settings, it is possible to understand the impact of parameter settings on the performance of the algorithm. Moreover, cross comparison of different parameter configurations can reveal any complimentary capabilities of parameter settings. This is possible through running algorithm instances on each others' hard and easy instances.

Our initial experimental results for the Max-Min algorithm with default parameters($\alpha = 1$ and $\beta = 2$) show the following. The distances between cities on the optimal tour are more uniform in the hard instances than in the easy ones. The approximation ratio is very close to 1 for all the generated easy instances whereas for the hard instances it is higher ranging from 1.04 to 1.29. Optimal tours for the easy instances lead to higher angles than in optimal tours of the hard instances. The standard deviation of angles of the easy instances are significantly smaller than the values of the hard instances. These values for both the hard and the easy instances slightly decrease with the instance size. Instance shapes for small instances structurally differ from the respective shapes of larger instances.

Furthermore, results of the second ($\alpha = 0$, $\beta = 4$) parameter setting support the above patterns. For example, the standard deviation of angles to the next two nearest neighbours follows a similar pattern for the second parameter combination. However, the third combination ($\alpha = 4$, $\beta = 0$) shows an increasing pattern (over increasing instance size) of standard deviation values in contrast to the decreasing pattern (over increasing instance size) in other parameter settings . This clearly shows that the heuristic information parameter β has significant influence, as $\beta = 0$ in the third setting.

We have compared the performance of ACO with the three considered parameter settings on each others hard and easy instances. In summary, our comparisons suggest that the first and second parameter settings complement each other, performing better on each other's hard instances. The third parameter setting has performed poorly compared to the other two settings. This provides insights into the impact of the heuristic information through the parameter β, as $\beta = 0$ effectively disables the heuristic information. These results reveals that, compared to pheromone information, heuristic information tends to play a more important role in ant tour construction. The second parameter setting , which lacks parameter α representing pheromone information, could still perform competitively against the default parameter setting. However, we cannot argue on the missing parameter as the sole reason for performance. For example, the extremely large value of α could impact tour construction significantly, thus causes the worse results of the third parameter setting.

4. FUTURE WORK

We have carried out an evolutionary algorithm approach to generate easy and hard instances for several parameter settings for a standard ACO algorithm and the Travelling Salesperson Problem. Hard and easy instances of several parameter settings are analysed and compared. Future work will concentrate on using our insights for algorithm design and parameter prediction in ACO.

References

[1] D. Applegate, W. J. Cook, S. Dash, and A. Rohe. Solution of a Min-Max Vehicle Routing Problem. *INFORMS Journal on Computing*, 14(2):132–143, 2002.

[2] M. Dorigo and T. Stützle. *Ant Colony Optimization*. Bradford Company, 2004.

[3] T. Kötzing, F. Neumann, H. Röglin, and C. Witt. Theoretical analysis of two ACO approaches for the traveling salesman problem. *Swarm Intelligence*, 6:1–21, 2012.

[4] O. Mersmann, B. Bischl, H. Trautmann, M. Wagner, J. Bossek, and F. Neumann. A novel feature-based approach to characterize algorithm performance for the traveling salesperson problem. *Annals of Mathematics and Artificial Intelligence*, pages 1–32, 2013.

[5] S. Nallaperuma, M. Wagner, F. Neumann, B. Bischl, O. Mersmann, and H. Trautmann. A Feature-based Comparison of Local Search and the Christofides Algorithm for the Travelling Salesperson Problem. In *Proceedings of the International Conference on Foundations of Genetic Algorithms (FOGA)*, 2013. To appear.

[6] P. Pellegrini, D. Favaretto, E. Moretti. On MAX-MIN ant system's parameters. In *Proceedings of the 5th International Conference on Ant Colony Optimization and Swarm Intelligence (ANTS)*, pages 203–214. Springer, 2006.

[7] K. Smith-Miles, J. van Hemert, and X. Y. Lim. Understanding TSP difficulty by learning from evolved instances. In *Proceedings of the 4th International Conference on Learning and Intelligent Optimization (LION)*, pages 266–280. Springer, 2010.

[8] T. Stützle and M. Dorigo. A short convergence proof for a class of Ant Colony Optimization algorithms. *IEEE Trans. on Evolutionary Computation*, pages 358–365, 2002.

[9] T. Stützle and H. H. Hoos. MAX-MIN Ant system. *Future Generation Computer Systems*, 16(9):889–914, 2000.

[10] T. Stützle, M. López-Ibáñez, P. Pellegrini, M. Maur, M. Montes de Oca, M. Birattari, and M. Dorigo. Parameter Adaptation in Ant Colony Optimization. In *Autonomous Search*, pages 191–215. Springer, 2012.