Chapter 1

Ant Clustering Algorithms

Yu-Chiun Chiou  
National Chiao Tung University, Taiwan

Shih-Ta Chou  
Feng Chia University, Taiwan

ABSTRACT

This paper proposes three ant clustering algorithms (ACAs): ACA-1, ACA-2 and ACA-3. The core logic of the proposed ACAs is to modify the ant colony metaheuristic by reformulating the clustering problem into a network problem. For a clustering problem of N objects and K clusters, a fully connected network of N nodes is formed with link costs, representing the dissimilarity of any two nodes it connects. K ants are then to collect their own nodes according to the link costs and following the pheromone trail laid by previous ants. The proposed three ACAs have been validated on a small-scale problem solved by a total enumeration method. The solution effectiveness at different problem scales consistently shows that ACA-2 outperforms among these three ACAs. A further comparison of ACA-2 with other commonly used clustering methods, including agglomerative hierarchy clustering algorithm (AHCA), K-means algorithm (KMA) and genetic clustering algorithm (GCA), shows that ACA-2 significantly outperforms them in solution effectiveness for the most of cases and also performs considerably better in solution stability as the problem scales or the number of clusters gets larger.

1. INTRODUCTION

Clustering, so-called set partitioning, is a basic methodology widely applied in the fields including statistics, mathematical programming (e.g., location selecting, network partitioning, routing, scheduling and assignment problems) and computer science (e.g., pattern recognition, learning theory, image processing and computer graphics). The core logic of clustering is to classify all objects into several mutually exclusive groups so as to achieve some optimal conditions. Most conventional clustering methods rapidly become computationally intractable as the problem scale
Ant Custering Algorithms

gets larger due to the combinatorial nature of the
methods. The mathematical model of clustering
for a given number \((K)\) of clusters can be formu-
lated as:

\[
[CA] \quad \text{Max } F(X)
\]

Subject to:

\[
\sum_j x_{ij} = 1 \quad \text{for all } i
\]

\[
\sum_j x_{jj} = K
\]

\[
x_{ij} \leq x_{jj} \quad \text{for all } i, j
\]

\[
x_{ij} = \{0, 1\} \quad \text{for all } i, j
\]

where \(x_{ij}\) denotes the relationship among objects. \(x_{ij}=1\) indicates that object \(i\) is assigned to object \(j\), (also cluster \(j\)). \(x_{ij}=0\) otherwise; \(i, j=\{1,2,\ldots,N\}\), \(N\) is the number of objects, \(K\) is the number of clusters. Accordingly, \(x_{ij}=1\) represents that object \(j\) is assigned to itself, indicating that cluster \(j\) has been formed. \(F(X)\) is the objective function. \(X\) is

The total number of feasible solutions of \([CA]\) is:

\[
|\pi_K| = \left(\frac{1}{K!}\right) \sum_{j=0}^{K} (-1)^{K-j} \binom{K}{j} j^N.
\]

That is, the solution space of \([CA]\) is exponential to the problem size and number of clusters. For instance, there are a total of 511 feasible solutions for 10 objects \((N=10)\) to be divided into 2 clusters \((K=2)\). There are a total of 42,525 feasible solutions for 10 objects to be divided into 5 clusters. Brucker (1978) and Welch (1983) proved that, for specific objective functions, clustering becomes an NP-hard problem when the number of clusters exceeds three, if one aims to find the optimal clusters. Hansen and Jaumard (1987) pointed out that even though the best algorithms developed for some specific objective functions, there would exhibit complexities of \(O(N^3\log N)\) or \(O(N^6)\), leaving some room for further improvement.

The heuristic algorithms for clustering can generally be divided into five categories: statistics clustering, mathematical programming, network programming, neural network and metaheuristics. The algorithms for conventional statistic clustering (e.g. Ward, 1963; MacQueen, 1967) include agglomerative hierarchical clustering method, divisive hierarchical clustering method, and \(K\)-means. The algorithms for mathematical programming (e.g. Mulvey & Crowder, 1979; Trick, 1992; Moshe, et al., 1996; Savelsbergh, 1997) range from dynamic programming, Lagrangian relaxation, linear relaxation, column generation and branch-and-price. The algorithms for network programming (e.g. Ali & Thiagarajan, 1989; El-Darzi & Mitra, 1995) include graph theoretic relaxations and network relaxation. The algorithms for neural networks mainly include self-organizing map (SOM) (e.g. Vesanto & Alhoniemi, 2000; Wu and Chow, 2004) and adaptive resonance theory (e.g. Bartfai, 1996). The algorithms for metaheuristics have been rapidly developed recently, including genetic algorithms (e.g. Lunchian, et al., 1994; Jiang, et al., 1997; Lozano, et al., 1998; Chioi & Lan, 2002; Garai & Chaudhuri, 2004), Tabu search (e.g. Al-Sultan, 1995; Sung & Jin, 2000), simulated annealing (e.g. Klein & Dubes, 1989; Selim & Al-Sultan, 1991), and ant colony optimization (e.g. Shelokar, et al., 2004; Tsai, et al., 2004; Kuo, et al., 2005; Yang & Kamel, 2006; Azzg et al., 2007).

Ant colony optimization (ACO) is a newly developed metaheuristic technique proposed by Dorigo et al. (1996) and Dorigo and Gambardella (1997). It is inspired by the behaviors of ant