Task Scheduling using Ant Colony Optimization Algorithm

Ranjani D.[1] Soumiya M. V.[2] Suganthi M.[3] Suvalakshmi B. P.[4]
[1]Assistant Professor [2,3,4]UG Student
[1,2,3,4]Department of Information Technology
[1,2,3,4]Sri Krishna College of Technology, Coimbatore, India

Abstract— Ant colony algorithm is a multi-agent algorithm that can be used to solve optimization problems. In this paper an ant colony optimization algorithm is used as in for travelling sales man problem and quadratic assignment problem. The cost and performance of the problems can be optimized using ant colony optimization for any real time application.
Keywords: Task Scheduling, Ant Colony, Algorithm

I. Introduction

As the evolution of distributed computing, grid computing and other technologies, cloud computing is more defined as a business computing model with the development of the Internet [1]. Due to potential computation, storage and processing capacity, cloud computing becomes primary computing paradigm to supported the IoT scenario and to leverage a massive heterogeneous set of devices that can access internet anywhere, anytime [5]. Tasks can execute in sequence or at the same time on two or more processors. Task mapping distributes the load of the system among its processors so that the overall processing objective according to given criteria is maximized. An efficient allocation strategy avoids the situation where some processors are idle while others have multiple jobs queued up. The task scheduling activity determines the execution order. To meet the computational requirements of a larger number of current and emerging applications, a satisfactory algorithm for task matching and scheduling is able to strengthen the parallelization functions. Ant colony optimization (ACO) is a metaheuristic, which is inspired by adaptation of a natural system. The medium used by ants to communicate information regarding shortest paths to food, consist of pheromone trails [7]. The travelling salesman problem (TSP) is a deceptively simple combinatorial problem. It can be stated very simply: a salesman spends his time visiting n cities (or nodes) cyclically. In one tour he visits each city just once, and finishes up where he started. The question is: in what order should he visit the cities to minimize the distance travelled? TSP became a benchmark for many new approaches in combinatorial optimization [1–6]. A large number of publications have been dedicated to study the TSP problem, together with some of its variations. The generalized TSP (GTSP) is a very simple but practical extension of TSP [2]. The quadratic assignment problem (QAP) is a combinatorial optimization problem has been introduced by Koopmans and Beckman in 1957. The QAP is a NP-hard (nondeterministic polynomial time complete) optimization problem [9]; even finding a solution within a factor of (1 + e) of the optimal one remains NPhard [10]. This paper presents a new method for combination of TSP and, the QAP, this method is a hybridization of the ant colony optimization with a local search method based on simulated annealing (SA).

II. Literature review

In order to exert advantage of the cloud and fog computing, a cloud and fog cooperation architecture has been found by the authors Jiuyun Xu, Zhuangyuan Hao, Ruru Zhang, and Xiaoting Sun. For the purpose of obtaining the most benefit from such an architecture, one must allocate computing tasks strategically at each processing node of cloud or fog layer. For the scheduling problems of complex tasks with priority constraints in IoT applications. The associated task scheduling in hybrid cloud-fog computing has been addressed. The associated task scheduling strategy based on laxity and ant colony system is proposed in cloud-fog environment, which takes into account the energy consumption and tries to fulfil reduce energy consumption on the condition of satisfying the mix deadline. Simulations and numerical results have shown that their work can show a better performance than other existing methods.[5]

Focused on the generalized travelling salesman problem, the ant colony optimization method from TSP is based on the basic extended ACO method, an improved method by considering the group influence has been found by the authors Jinhui Yang, Xiaohu Shi, Maurizio Marchese, and Yanchun Liang. To avoid locking into local minima, a mutation process is also introduced into this method. The numerical results show that the consideration of group influence in GTSP problem improves the performance of the basic extended ACO method. Their proposed methods could get fairly good results when the problem scale is less than 200 cities. [2]

The resource utilization in hybrid cloud computing with multiple considerations of cost, task completion time, and resource provider interests, and developed by the authors Liyun Zuo, Lei Shu, Shoubin Dong , Yuanfang Chen ,and Li Yan. Firstly, they proposed the cost-first and time-first single-objective hybrid cloud scheduling optimization strategies, which respectively prioritize cost with deadline constraints and task completion time with cost constraints. The MOSACO algorithm was developed by combining the two single-objective scheduling optimization approaches with entropy optimization to establish a multi-objective scheduling model optimized using an improved ant colony algorithm. Compared with other similar resource scheduling methods, MOSACO demonstrated the highest optimality according to considerations of task completion time, cost, number of deadline violations, and the degree of private cloud resource utilization [6].

The authors Nihan¸ Etin Demirel, and M. Duran Toksar has proposed a powerful algorithm (AntSimulated) for the QAP, which is based on the ant colonies. The search process of each ant has been reinforced with a local search procedure based on SA. AntSimulated has a new strategy to reinforce pheromone trails. This strategy includes function of the best three of found solutions for updating the pheromone trails. Comparisons with some of the best heuristics for the QAP have shown that Ant Simulated is among the best as far as irregular and regular problems are concerned. Especially, their results show a noticeable increase in performance compared with previous ant systems for the QAP (HAS-QAP algorithm). Thus demonstrating the complementary gains brought by the combined use of a powerful local search [7].

III. RELATED CONCEPTS

A. Generalized Traveling Salesman Problem

The generalized traveling salesman problem (GTSP) has been introduced by Henry-Labordere, Saksena and Srivastava in the context of computer record balancing and of visit sequencing through welfare agencies since the 1960s. The GTSP represents a kind of combinatorial optimization problem. It can be described as the problem of seeking a special Hamiltonian cycle with the lowest cost in a complete weighted graph. Let $G = (X, E, W)$ be a complete weighted graph, where

$X = (x_1 , x_2 , . . . , x_n) \ (n \geq 3)$

$E = \{ e_{ij} \mid x_i , x_j \in X \}$

and

$W = \{ w_{ij} \mid w_{ij} \geq 0 \text{ and } w_{ii = 0} , \forall i,j \in \{ 1,2, . . . , n \}$

Are vertex set, edge set and cost set, respectively? The vertex set X is partitioned into m possibly intersecting groups $X_1, X_2, ..., X_m$, with $\mid X \mid \geq 1$ and $X = \bigcup_{j=1}^{m} xj$. The special Hamiltonian cycle is required to pass through all of the groups, but not all of the vertices differing from that of the TSP. For convenience, we also call W as the cost matrix and take it as $W = (w_{ij})_{n \times n}$. There are two different kinds of GTSP under the abovementioned framework of the special Hamiltonian cycle: (1) the cycle passes exactly one vertex in each group and (2) the cycle passes at least one vertex in each group [2].

B. Quadratic Assignment Problem

The QAP can be described as the problem of assigning n number of facilities to n number of locations with given distances between the locations and given flows between the facilities. The goal then is to place the facilities on locations in such a way that the sum of the product between flows and distances is minimized [8]. More formally, given n facilities and n locations, two $n \times n$ matrices,

A = $[a_{ij}]$ where a_{ij} is the distance between locations i and j,

B = $[b_{kl}]$ where b_{kl} is the flow between facilities k and 1.

QAP can be stated as follows:

$min\ (\pi) = \sum_{i=i}^{n} \sum_{j=l}^{n} aijbkl$,

$k = \pi_i$ and $l = \pi_j$ where Π(n) is the set of permutations of n elements. Using PǒnÞ QAP can be described as follows:

$\min_{\pi \in \Pi} f(n) = \sum_{i=i}^{n} \sum_{j=l}^{n} aijb\pi i\pi j$.

QAP instances of size larger than 20 are intractable, so a large number of metaheuristic methods for solving them have been used up to now [7].

IV. PROPOSED ALGORITHM

To meet the objectives, we studied the influence of ACO's main parameters on its performance and the interaction with the usage of local search. We measure the time spent by ACO and other algorithms for pheromone update and solution construction. ACO and Ant Colony System (ACS) were considered for the TSP and for the QAP. ACO is much faster for the pheromone update for the TSP. For ACO, a small percentage, allowing it to generate many more solutions. With local search, the advantage of the fast ACO's pheromone update is reduced. The same happens if the computation of solution quality is computationally more demanding, as it is the case for the QAP; for QAP with local search the time spent for the pheromone update by ACO is negligible and, hence, the speed advantage of ACO becomes virtually irrelevant. As a next step, we examined how ACO specific parameters influence the algorithm's behaviour; in particular, we tested the algorithm with different settings of K and τ_{max} .If local search is applied, the best K now becomes clearly one on the TSP; on the QAP, ACO with a larger value of K performs better especially for larger instance. In ACO, the ratio between τ_{max} and τ_{min}, together with the setting of K, determines how strong the search intensification is. However, the changes τ_{max} shows a different behavior when ACO is applied to different problems. For the TSP, the strategies' performance is quite similar independently of whether local search is used. However, the quality-based strategy obtains better results. For the QAP and without local search, the elitist-based strategy is almost good, while the other strategies show worse results. With local search, ACO performs better, where all strategies are competitive with other optimization algorithms.

We restart by re-initializing the pheromone values to τ_0 after r iterations without improvement. This mechanism improves ACO's performance, especially when local search is used. A more extensive comparison showed that (i) for the TSP without local search, ACO performs better on most instances, (ii) for the QAP without local search, ACO performs better was observed. Overall, these results show that ACO with the restart procedure appears to be competitive, regardless of whether local search is used.

The ACO algorithm for TSP and QAP is

1) Step 1

Set *time:=0* {time is time counter}

For every edge *(i,j)* set an initial $\tau_{ij} = c$ for trail density and $\Delta\tau_{ij} = 0$.

2) Step 2

Set *s:=0* {s is travel step counter}

For *k:=1 to l* do

Place ant *k* on a city randomly.

Place the city in $visited_k$.

Place the group of the city in $tabu_k$.

3) Step 3

Repeat until $s \leq m$

Set $s:=s + 1$

For *k:=1 to l* do

Choose the next city to be visited according to the probability $p_{ij}^{\ k}$

Move the ant *k* to the selected city.

Insert the selected city in $visited_k$.

Insert the group of selected city in $tabu_k$.

4) Step 4

For *k:=1 to l* do

Move the ant *k* from $visited_k$(n) to $visited_k$(1).

Compute the tour length L_k traveled by ant k.
Update the shortest tour found.
For every edge *(i,j)* do
For $k:=1$ to l do
Update the pheromone trail density τ_{ij}. *time:=time + 1*

5) Step 5
If (*time* < *TIME_MAX*) then
Empty all $visited_k$ and $tabu_k$
Goto Step 2.
Else
Print the shortest tour.
Stop

1) Step 1
Set $s:=0$ {s is travel step counter}
Determine the initial permutation (π^k; the permutation of ant k), an initial temperature (T_0), a last temperature (T_n), a cooling proportion (r);

2) Step 2
Create new permutation (π_j) from current permutation (π_i) by using selected neighborhood mechanism;
$\Delta = f(\pi_j)-f(\pi_i)$
If $\Delta \leq 0$ Δ Then
$\pi_i := \pi_j$
Else If random (0.1) < exp(- /T)Then
$\pi_i := \pi_j$

3) Step 3
$T_{s+1}=T_s/(1+rT_s)$;{T_s is a temperature of iteration}
$s:=s+l;$
If $s>K$ Then
Stop;
Else Go to STEP 2.

V. Conclusion

In this paper, we proposed combined concept of the ACO algorithm for the TSP and the QAP. Without the usage of a local search has strong impact on parameters settings for ACO applied to the TSP. In addition, we have shown that ACO shows early stagnation behaviour and introduced a restart mechanism which has improved significantly the overall performance of ACO. We conclude that, with the restart procedure and with the right configuration of combined optimization, ACO is competitive to the advantage of finding good solution quality in a shorter computation time.

Reference

[1] Li Liu, Taiyu Luo, and Yuanyuan Du, "A new task scheduling strategy based on improved ant colony algorithm in IaaS layer"
[2] Jinhui Yang, Xiaohu Shi, Maurizio Marchese, Yanchun Liang "An ant colony optimization method for generalized TSP problem", National Natural Science Foundation of China and Chinese Academy of Sciences. Published by Elsevier Limited and Science in China Press. All rights reserved.
[3] Y Dai, Y Lou, X Lu.Task Scheduling Algorithm Based on Genetic Algorithm and Ant Colony Optimization Algorithm with Multi-QoS Constraints in Cloud Computing[C]. International Conference on Intelligent Human-Machine Systems and Cyberneti, 2015:428-431.
[4] Reinelt G. TSPLIB – a traveling salesman problem library. ORSA J Comput 1991; 3:376–84.
[5] Jiuyun Xu, Zhuangyuan Hao , Ruru Zhang , and Xiaoting Sun," A Method Based on the Combination of Laxity and Ant Colony System for Cloud-Fog Task Scheduling"
[6] Liyun Zuo, Lei Shu, Shoubin Dong , Yuanfang Chen ,and Li Yan, "A Multi-Objective Hybrid Cloud Resource Scheduling Method Based on Deadline and Cost Constraints"
[7] Nihan¸ Etin Demirel , M. Duran Toksar, "Optimization of the quadratic assignment problem using an ant colony algorithm"
[8] T. Stutzle, M. Dorigo, AGO algorithms for the quadratic assignment problem, New Ideas in Optimization, McGraw-Hill, 1999.
[9] S. Shani, T. Gonzalez, P-complete approximation problems, Journal of the ACM 23 (3) (1976) 555–565.
[10] M. Queyranne, Performance ratio of polynomial heuristics for triangle inequality quadratic assignment problems, Operations Research Letters 4 (1986) 231–234.