An Efficient Multiple Object Resource Allocation Using Hybrid GA-ACO Algorithm

Senthil Kumar, A.M. and M. Venkatesan

ABSTRACT
In Cloud computing the major aspect is to maintain the data and application which is used in remote location. The difficult task in larger industries and global networks are demand driven which utilizes the properties of data and its application that facilitates to obtain the remote systems. The aim of multi objective optimization problem is to simultaneously optimize total no of resource available and total no of destination available to allocate the data's and software etc. The possible solution for resource allocation problem is obtained in a effective manner using Ant Colony Optimization algorithm (ACO). Moreover, the algorithm provides effective solution when we handle the larger amount of data and larger network over the globe and it also deals with the potential solution for position and cost. Since, the proposed algorithm ACO is integrated to Genetic Algorithm (GA) for optimized resource allocation. In larger data and network the software allocation from source to destination node is integrated using two algorithms obtained with the maximum feasible solution and also the computation time should be minimized.

INTRODUCTION
A platform and type of application is used to distribute the large level networks and search engines. This platform and application is known as Cloud computing. In cloud, the Servers in physical machines or virtual machines which configure dynamically and reconfigures in remote server and client nodes thus provide a cloud computing platform (Weiss, A., 2007; Vaquero, L., 2009). In recent trends the clouds include typically other computing resources like as Storage Area Networks (SANs), Network Equipment, Firewall and other security devices. Cloud computing also access and allocate resources and applications that are extended to make it accessible through the Internet around the globe (Rajkumar Buyya, 2010). In large data centers, Industries and Servers that web based applications and web host services are applied to obtain the feasible solution in cloud application (Armbrust, M., A. Fox, et al., 2009). Since, the user accesses the application and data with a suitable Internet connection a standard browser can access a cloud application. In cloud computing the two major components in cloud architecture are as follows,

Service Level Agreements/Resource Allocator:
The Service Level Agreements (SLA) Resource Allocator acts as the interface between the source node and destination node such as the Data Center/Application center through the cloud service provider and remote users/brokers. The interactions between the source and destination nodes require support from the resource allocator and also from service level agreements oriented resource management.

The most powerful search engines (Google, Yahoo etc.,) in web application should allow to run the web application in web server and the application written in standard programming language with libraries. The application providers support and allow the application programming interfaces for account management, data access, web storage, mailing services and host allocation. However, the remote server provides the application and data storage for every user’s access around the world. In source the destination nodes or the remote users should be able to access and share the application and data (upload and download) through the application programming interfaces. The application of cloud initiative can be divided into three phases such as scheduling, executing and resource recovery. In addition, to the service requests process through the cloud in application center or data center.

Users/Brokers:
Over the globe through the internet users or brokers access the application and data and submit

Corresponding Author: Senthil Kumar, A.M., Research scholar & HOD, Dept. of CSE, Tejaa Shakti Institute of Technology for women, Coimbatore, Tamilnadu, India.
E-mail: Senthil1185@gmail.com.
finding the best match between the available resources and set of users is known as NP-hard Problem (Klaithem Al Nuaimi, 2012; Ian Foster, 2008). The computational complexity of cloud computing is as follows:
1. Computational performance is based on time to time updation of resource variations.
2. The unreliable connection between the devices like computers to toggle, router and mobile phones.
3. The resources can be connected or disconnected over the cloud network at any time.
4. The unavailability of resources without any notification.

However, several real time problems involve taking multiple criteria for resource allocation. In this case, current research is apt to look at the multi-objective position. Therefore, the difficulty of resource allocation is formulated as a multi-objective combinatorial optimization problem aiming to concurrently optimize best position, time and power consumption. A modified version of the Ant Colony (AC) algorithm is proposed and designed to deal effectively with large solutions for large scale data centers, Industries etc. when a multi-objective resource allocation problem arises where time consumption, resource best position and cost should be minimized (Qinghua Zheng and Rui Li, 2015). The performance of the ant colony algorithm is evaluated using multi-objective genetic algorithm.

**II. Problem Formulation:**

**Ant Colony Optimization:**

ACO algorithm and the Monte Carlo (MC) systems take parallel simulation for resource allocation. The general stochastic simulation systems is known as MC systems, since the stochastic components contain transition rules and state sampling, making use of sampling experiments to perform the MC systems better (Medhat A. Tawfeek and Ashraf El-Sisi, 2013). The experimental results are updated to gain statistical knowledge about the problem to obtain the minimum feasible solution. In this technique data can be also iteratively used to decrease the difference in the estimation of the described nodes and directing the simulation process over the cloud network for resource allocation (Kun Li, 2011; Yue Zhou and Xinli Huang, 2013). ACO algorithms the ants sample the problem’s solution cloud network by frequently applying until a feasible solution of the allocate resources from the request is used to obtain a stochastic decision policy. In sampling the repeated ants is realized concurrently by a collection of differently request replicas of the same ant. The ACO recursive algorithm each ant “experiment” allows to adaptively changing the numerical information on the problem construction (Madhurima Rana, 2014).

Ant Colony optimization is stirred by the study of actual ant colonies and based mostly upon their combined forage performance. In the Ants area unit, social insects and board colonies, their behavior rule is based on the goal of colony endurance. The supply of resources and during checking out food, ants often travel between their nest and food resources, the ants travel round the space neighboring their nest in an exceedingly random manner. While moving, ants drop distinctive substances known as pheromones and these pheromones determines length. Ants will smell pheromones and by robust secretion awareness they will identify their path from source to destination and vice versa (Lin Wang and Lihua Ai, 2013). The ant finds the food resources and it evaluates the standard and amount of the food to hold back to the nest. However, throughout the comeback, the ant drops some food in an exceedingly ground and therefore the quality and amount of pheromones of the food. The ACO formula contemplate the 3 major parameters contemplate as follows,
**Pheromone Update:**

In ACO algorithm the pheromone updating the trails to obtain the minimum feasible solution for identifying the source and destination and they use two kinds of strategies to update the pheromone trails.

**Definition of Pheromone and Heuristic Information:**

At each step two parameters such as pheromone and a heuristic factor is taken for consideration. There are two approaches to define and use one or more sources. When only one source is employed, the pheromone information associated with each destination is shared to reduce the multiple sources into a single node or source. If multiple sources are used, usually each source matches to destination. With respect to the pheromone information, each source may contain different values depending on the implementation strategy applied.

**Pheromone and Heuristic Aggregation:**

Whenever multiple sources are used, one must use some form of aggregation procedure to aggregate the pheromone/heuristic matrices. There are three common strategies for this: (1) The weighted sum, (2) The weighted product, and (3) Random, where at each step a random objective is selected to be optimized.

**Basic Ant Colony Algorithm:**

1. **Initialize the variables**
2. **Begin**
3. **Initialize the pheromone**
4. **While (stopping criteria) do**
5. **Position at each ant starting on source**
6. **While (building a solution) do**
7. **Chose source to allocate the destination**
8. **End for**
9. **End while**
10. **Update the pheromone**
11. **End while**
12. **End**

**Genetic Algorithm:**

Genetic Algorithm (GA) is an evolution process and natural selection is being simulated, the resource is followed by the requests, evaluation, selection, crossover and regeneration. The sequences of requests are randomly generated for resource allocation problem that needs to be evaluated. The GA will select the sequence of requests timing based on certain probability that will mate in the next process. The best value obtained from Crossover and Mutation function will be performed. In order to generate a new minimum possible solution compared to the previous one. This repeated process is achieved until the best solution for minimum cost, position and time to be achieved (Zainudin Zukhri 2013; Santanu Dam and Gopa Mandal, 2015).

GA is greatly suitable in many optimization problems, as this technique does not require a early information of the problem to be solved in nature, it is measured the in support of GA. In addition, this algorithm is probable to find the global optimum and it is well tailored to the problem. On the other hand, GA is easy to drop into early convergences that create the best solution which is hard to achieve.

**III. Proposed Algorithm:**

Integration of two algorithms, such as ACO and GA in resource allocations is done to obtain the minimum feasible solution in multiple node resource allocation from multiple sources to destinations. In this concept the ant colony system and genetic algorithm combined to achieve the best cost, position and minimum computation time between the source and destination. There are several works based hybridization techniques that are available but the proposed hybrid GA and ACO as a new algorithm to solve the problem of resource allocation. This hybridization will simplify in selecting the adjustable parameter in ACO where human experience is needed and in most cases depending on coincidence. The two algorithms based on GA and ACO hybridization covers up the weaknesses of both algorithms. ACO is employed to help GA to eliminate the appearance of invalid tour, while GA is used to overcome the dependency the matrix of pheromone in ACO.

The hybrid ACO-GA algorithm can be described as follows:

Step 1. Initialize parameters.
Step 2. Initiate the iteration
Step 3. Loop /*An iteration */
Step 4. Each ant is positioned on a starting node.
Step 5. Each ant applies a state transition rule to incrementally build a solution and a local pheromone updating rule until all ants have built a complete solution.
Step 6. Update the each ant.
Step 7. Global pheromone updating rule is applied until end condition.
Step 8. Stop further iterations Each edge between node (i, j) has a distance or cost associate δ (i, j) and a pheromone concentration τ (i, j). The equation 1 is the state transition rule, which is a probabilistic function for each node u, which has not been visited by each ant placed on the node r.
Step 9. If minimum feasible solution is not obtained repeat the step 2 to 8.
Step 10. End

\[
S_r(i, j) = \frac{[(i, j)][\eta(i, j)]^\beta}{\sum [(i, u)][\eta(i, u)]^\beta}
\]  

(1)

The parameter \( \beta \) determine the relevance of the pheromone concentration compared with the distance or cost, \( \tau (i, j) \) Global pheromone updating rule can be applied as:
\[ T(i, j) = (1 + \alpha)T(i, j) + \Delta \tau_k(i, j) \]  

Where \( \alpha \) is the pheromone evaporation factor between 0 and 1 and \( \Delta \tau_k(i, j) \) is the reverse of the distance or cost is 1 for ant \( k \), if \((i, j)\) is its path and if the value is 0 if it is not in the path. The steps can be modified to manage cloud architecture. The cloud is visualized is the collection of clustered services, hence the live services of cloud behaves like an ant, when it finds its file object, the task of the ant is completed. Subsequently, considering the prime component of cloud computing, the compute cloud and storage cloud can be modeled as virtual services of cloud. Every time a request is processed on a cloud cluster site, \( \tau \) is updated for all the site connections and thus the “(2)” can be modified by associating a parameter \( \tau \).

\[ d(i, j) = (1 - \alpha)\tau(i, j) + \sum \Delta d_{i\tau}(i, j) \]  

**Fig. 2:** The Rule for Genetic algorithm used for resource allocation.

The dot operator represents time for each cloud scheduling service. Therefore, the \( \alpha_{tcs} \) is introduced which expresses the evaporation factor under the time slot of cloud service. The heuristic can be divided into two categories for cloud-based services e.g. On-Line Mode Service and the Batch Mode service. In online mode, whenever a request arrive, it immediately allocate to the first free resource allocator. The arrival order of the request in cloud is important in this method. Here, each service request is considered only once for matching and scheduling. In batch mode, the requests are collected; the
scheduler considers the approximate execution time for each task and use heuristic approach to possibly make better decision. The function free \([y]\) will be the return time, when the resource allocators \(M_i\) is free. We consider,

\[
\text{free}[y] = I\Delta + ET_{xy}
\]

where, \(I\Delta\) is the initial time slot of request of service made on the cloud architecture and \(ET_{xy}\) is the execution time matrix of request \(ri\) on resource allocator \(m\). The scheduling of resource allocator on the cloud service proposes the probability of servicing the request:

\[
D_{xy} = \frac{D^{p}_{xy} \eta_{xy} \left( \frac{1}{ET_{xy}} \right)}{\sum D^{p}_{xy} \eta_{xy} \left( \frac{1}{ET_{xy}} \right)}
\]

Where,

- \(\eta_{xy}\) is the attractiveness of the move as computed by heuristic information indicating a prior desirability of that move.
- \(D^{p}_{xy}\) is the pheromone trail level of the move, indicating the fast and accuracy of the cloud service in the past (with lower values) to make that particular move (therefore it represents a posterior service accomplishment indication of the desirability of that request).
- \(ET_{xy}\) - Execution Matrix of service and resource allocator. In this proposed model, we select the highest probabilities ‘i’ and ‘j’ is the next request service \(i\) is executed on the resource allocator \(j\).

**IV. Result and Output Discussions:**

In the developed hybridization technique cloud network optimization algorithm (GA-ACO) scheduling request service is resource allocation. Practically, in this algorithm simulation by varying the number of nodes or the service requests with the parameters crossover and keepover respectively changed to obtain the optimum solution in cloud network. The crossover and keepover parameters are considered with each iteration of nodes and or the request in resource allocation is mainly based on overall computation time, best cost and position to be obtained.

The Figure 3 shows the performance of the service request in resource allocation with respect to the no of nodes with cost. The probability of the coefficient update in the ant colony optimization \(P_{xy}\) fetches the resource allocation in cloud service. In multi object resource allocation problem, the hybrid genetic algorithm is achieved using ant colony Optimization (GA-ACO) to obtain the best values such as cost, position and the overall compute time for resource allocation. It is shown below in the table 1 and table 2. In Table 1 & 2 varying the number of nodes to obtain the best cost, position and computing time with the same iteration.

![Fig. 3: Resource Allocation Under GA-ACO.](image3.png)

![Fig. 4: Comparison of Best Cost Value.](image4.png)
Table 1: Best cost and Position to obtain the Keep percentage is 0.9.

<table>
<thead>
<tr>
<th>No of Iteration</th>
<th>pop size</th>
<th>Best Cost</th>
<th>Best Position</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>5</td>
<td>25.8136</td>
<td>9.2589</td>
<td>8.0038</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>8.528</td>
<td>8.2533</td>
<td>8.0226</td>
</tr>
<tr>
<td>100</td>
<td>20</td>
<td>16.1614</td>
<td>13.0198</td>
<td>7.9083</td>
</tr>
<tr>
<td>100</td>
<td>50</td>
<td>16.7541</td>
<td>11.2763</td>
<td>8.0238</td>
</tr>
</tbody>
</table>

Table 2: Best cost and Position to obtain the Crossover percentage is 0.9.

<table>
<thead>
<tr>
<th>No of Iteration</th>
<th>pop size</th>
<th>Best Cost</th>
<th>Best Position</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>5</td>
<td>34.9980</td>
<td>11.7121</td>
<td>7.9811</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>22.4139</td>
<td>6.8082</td>
<td>7.8986</td>
</tr>
<tr>
<td>100</td>
<td>20</td>
<td>15.2919</td>
<td>10.5266</td>
<td>7.6256</td>
</tr>
<tr>
<td>100</td>
<td>50</td>
<td>18.0545</td>
<td>10.9970</td>
<td>8.0125</td>
</tr>
</tbody>
</table>

Fig. 5: Comparison of Best Position.

Fig. 6: Comparison of Computation Time.

V. Conclusion

In this paper, the hybrid genetic algorithm with ant colony optimization has been proposed to obtain the resource allocation over the global network under cloud computing technique. The pheromone update mechanisms used to support and minimize the cost and position of the service request and the ACO coefficient is updated with respected to $\tau$. In time to time service due to absence of any restore time in service and resource allocation distribution, the continuous process in ant colony is integrated with genetic algorithm in cloud network to obtain a better optimization service request and resource allocation.

REFERENCES


Swarm Based Algorithms” International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT).

Medhat A. Tawfeek and Ashraf El-Sisi, 2013. “Cloud Task Scheduling Based on Ant Colony Optimization” IEEE.


Santanu Dam and Gopa Mandal, 2015. “Genetic Algorithm and Gravitational Emulation Based Hybrid Load Balancing Strategy In Cloud Computing” IEEE.


Yue Zhou and XinLi Huang, 2013. “Scheduling Workflow in Cloud Computing Based on Ant Colony Optimization Algorithm” IEEE, Sixth International Conference on Business Intelligence and Financial Engineering.