Indian Journal of Science and Technology, Vol 9(48), DOI: 10.17485/ijst/2016/v9i48/107967, December 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

# Genetically Modified Ant Colony Optimization based Trust Evaluation in Cloud Computing

**J. Bharath* and V. S. Shankar Sriram**

School of Computing, SASTRA University, Thanjavur - 613401, Tamil Nadu, India;
bharath525113@gmail.com, sriram@it.sastra.edu

## Abstract

**Objectives:** Cloud computing is a virtualized and scalable platform which helps the users to reduce the cost incurred for setting up and maintaining an IT infrastructure. Despite the various benefits offered by cloud, it faces stringent challenges with respect to security and trust management. Trust covers the security aspects of cloud. The proposed system focuses on selection of optimal parameters used to ensure trust value. **Methods:** The hybridized techniques for cloud Trust Management works with a pre defined set of parameter values. Hence the trust value computed using these optimal parameters has a great impact on the overall accuracy of the trust score. We intend to use Genetically Modified Ant Colony Optimization GM-ACO technique to identify best Trust Metric Parameters (TMPs) with respect to Cloud Service Providers and it significantly outperforms compared with existing techniques. **Findings:** The genetically modified ACO algorithm will reduce the complexity of calculating the trust score of many service providers in the cloud environment. **Applications:** Managing trust in peer-to-peer systems, social network based systems, recommendation based systems, policy based systems, reputation based trust systems, trust mining systems etc.

**Keywords:** Ant Colony Optimization, Cloud Computing, Genetic Algorithm, Trust Evaluation, Trust Management

## 1. Introduction

Cloud computing gives us a scalable, dynamic, elastic and shared resources (e.g. Software, storage, computing power etc.) on the internet from data centres around the globe to the users (e.g. individuals, government organizations, business organizations, etc). The services offered by the Cloud computing are too attractive to the users which we cannot ignore on today's competitive environment on service. However, it is not free of threats[1] or obstacles. The non-transparent and highly distributed nature of Cloud computing shows a considerable immediate threat to acceptance of cloud services. The reputed users of this cloud services are often recognizing that they leave the control on their data and they do not have trust over their cloud service provider. The recent survey conducted among the consumers, about 84% are concerned on their data storage[2] on some location and 88% are worried that possibility of others on accessing their data. The business on Cloud computing are growing rapidly, the new players are entering the Cloud computing market on competition to provide service to consumers with primary functionalities. But, there are huge differences on quality of services, features and security. Therefore, it increases the choice of Cloud Service Providers in the market, making it very difficult for the user to select an appropriate service provider. This problem is addressed by the Trust Management system which ranks the service providers according to the user requirements. Trust Management helps to select the Cloud Service Providers based on their trust values. We know that Cloud computing is a dynamic environment; to improve the trust accuracy the trust degree is maintained also it sets the trust threshold. The access denied to the cloud user and cloud service provider who has trust value below the threshold. Here trust computing has a major difficulty, the characteristics of trust is dynamic in nature as Cloud computing is a dynamic environment. The changes happen to cloud resources, users, service

*Author for correspondence

providers at any instance of time, so the trust degree is maintained on frequent monitoring of these changes based on certain parameters. The existing trust follows a mutual trust algorithm that will ensure trust on both user and service provider. The behaviour model of service provider is processed by Ant Colony Optimization, because in cloud the trust degree between the interacted elements is based on the pheromone concentration in Ant Colony Optimization. The users are made to select elements with more credibility to provide resources or services, because the ant will choose the path with high pheromone concentration. The other reason to choose ACO is it is an intelligent and efficient population solving algorithm that can be easily combined with other techniques. So, it is acceptable to use ACO on trust management. Also, the parameters setting of α, β has more influence on the performance of ACO. Our proposed system focuses on selection of optimal parameters used to ensure trust value. We intend to use genetically modified Ant Colony Optimization GM-ACO technique to identify best parameters with respect to Cloud Service Providers and it significantly outperforms compared with existing techniques. The proposed algorithm enhances the classification rate with reduced complexity in the Trust Management systems and also select best appropriate cloud service node based on the user requirement.

## 2. Evolution of Trust Computing

Trust computing an additional security that stands behind every successful transaction on today internet based computing; the evolution of such secured computing is explained as follows:

### 2.1.1 Formalizing Trust

Marsh is the first person to introduce Trust Management as computational concept; he formalizes the trust and made a system that embedded an artificial agent[3]. The agent that makes decision based on trust in the area of Distributed Artificial Intelligent (DAI) or Multi-Agent Systems (MAS). Then, he defines the aspect of trust, methodology and discussion on values like subjectivity and sensitivity. He deeply defines the trust from the starting point explained by the authors, the generality like risk, benefits, costs, cooperation and confidence. Then, he define about a tool which is formulizing, way of approaching the problem, some traditional methods and defines the use of DAI as a research tool. Finally, he showed a path on future work as his proposed method had some limitations on values (range) that are chosen for trust, because it is not valid always as prescribed.

### *2.1.2 PGP and X.509*

The best existing system so far used in security application, generally referred as certificate systems. Some of the commonly known certificate systems are Pretty Good Privacy and X.509 framework[4], which is based on cryptographic technique. In PGP user generates a key pair (Public Key, Secret Key) with respect to one's unique ID, Public Key and Secret Key were stored in a respective key rings. These rings were stored and managed by each user. If one user has a good Public Key record of another user which he/she is confident then he/she share it to other user, here the first user is act as introducer to the third user. Each user must sign the introducer's Public Key record with his/her Secret Key and share it to PGP system. It is specifically designed to secure email for individuals and inefficient for wider range of network services. X.509 framework is Public Key infrastructure based on ITU-T standard, a Certification Authority (CA) issues certificate; the certificate contains information more than PGP. It differs from PGP by centralization of data, in PGP anybody can sign a Public Key and acts as an *introducer* but here everyone can get certificates from CA. This framework depends on assumption that the CA is arranged as "certifying authority tree" and users within "group of interest" have the keys that are signed by the CA[5] with a previous authority in a global certifying tree. Though, there are differences in certification systems they are similar in operation.

### *2.1.3 Decentralized Trust*

In proposed a Trust Management model which is based on Policymaker Approach which is derived from the concepts of basic certificate systems. PGP and X.509 are restructured and used in this approach. This approach towards Trust Management is based on certain principles. They are flexibility, unified mechanism and locality of control and separation of mechanism. Policymaker starts at certificate based security model that cantered at wrapping the identities to keys and it allows a secure service requested by user to show that they had a certificate information that allow them to use services. The Policymaker provides simple language to express condition under which an authority or individual is trusted. These services

are appeared as database query engine to the application, the inputs to Policymaker is a set of local policy statement and string represent a proposed trusted action, a collection of credentials. The basic function of Policymaker is to process queries. It is to determine that whether a represented Public Key is permitted to do or perform particular action according to local policy. Though it has several advantages they had some issues like the functions that is encapsulated only in some components (certificates, string management function and policies), they developed only a prototype system there is no formal model of Trust Management system.

#### *2.1.4 Right type of Trust*

Audun examines and tells about the type of trust relationships and trust which is more relevant to data security, he did not attempts to describe a formal trust model. He defines present of trust as a phenomenon is based on the existence of the malicious behaviour of the user, also tells about the trust from the malicious point of view, also describes about the trusted third party and trusting party. Then, he defines trust as knowledge about security; here knowledge is referred as information about the system that relates to the threat to that system. It must be as complete as possible otherwise it will lead to insecure or security breaches. He also compares security and reliability. He defines the importance of trust and its importance in the field of information security and encourages for the future work like he wants to extract parameters for the trust from the real world, defining principles and a formal trust model. Later, in 1999 he defines algebra model for accessing trust at certification chains.

#### *2.1.5 Trust Model for Ubiquitous System*

In proposed a trust model for the system of ubiquitous which is depend on trust values, this model requires all entities to save all the trust values and then evaluate trust between two entities who want to interact. The metrics are developed based on following in sections as Peer recommendation that will obtain trust vector, peer set and common peer vector. The confidence, history of past interaction and time based evaluation are also taken into consideration for trust evaluation in metric. It is taken uncertainty of trust and defines precise model, beside basic factor they additionally added time based evaluation to calculate trust value and efficiently handle false recommendation. This model can handle two types of system having past interaction history and system which are communicating for the first time. Certain tuning parameters are suggested in order to meet security requirements in distributed system. He also suggests some future work to implement the proposed model in ubiquitous environment. He wants to add additional risk analysis on security measures which is proportional and also define work to exchange values among principles. So, that certain principle can force their trust values on other users.

### 2.2 Trust based Access Control Techniques

Access Control is an important mechanism to fulfil the security needs of networks; research on this area received more interest in past years. Many scholars put access control models that are given as follows.

### 2.2.1 Role based Access Control

In[6] analyzed the requirement of access control which is dynamic in nature on Cloud computing and Role Based Access Control (RBAC) is applied to Cloud computing environment. This model is based on identification mechanism and it is enclosed, so it can be applied only to enclosed network. RBAC cannot be applied to large environment, distributed network, particularly to cloud environment because it is open and dynamic in nature. RBAC model verifies only user's authentication identity without considering user's trust. To resolve this problem, they[6] proposed in which they can insert Trust Management into the existing access control model and developed new RBAC model a dynamic trust-based RBAC against its disadvantages. It gives a trust calculation process, access control policy based on user's role data and trust degree. This would reduce the risk in cloud environment and increases the security.

#### *2.2.2 Trusted Cloud Computing Platform*

In[7] introduce a Trusted Cloud Computing Platform (TCCP), where Infrastructure as a Service (IaaS) providers can give a type of close execution environment to the users and make sure the confidentiality of other virtual guest system. It also allows the users to check whether the service provided to them through the IaaS is secured or not before accessing the virtual machine. It also guarantee that no administrator of cloud provider's can view or modify the user's content. This also has the notion of attestation on entire service; it is allow a user to verify whether its process of computation will run securely.

### 2.2.3 Privilege Chain for Access Control

In[8] proposed a model based on authorization chain known as privilege chains. It is very significant that the information used for security management on cloud. It is used to confirm trust of resources and to guard cloud resources from an unauthorized access. Here the resources are not trusted if the privilege chain is broken, it shows there is missing subject in the chain. The chain path flow from the creator to user, it turns that the resources as an actors that are not trusted. Mainly, this privilege model uses the data about data about resources and access control rule to create authorization chain. If the chain is traceable and complete then the resources is trusted. But building authorization chain will take certain amount of time due to large number of cloud resources. So, it will surely reduce the computational efficiency and decrease the performance of services.

### 2.2.4 Mutual Trust Model

In proposed a new access control that ensures mutual trust on user and service provider in Cloud computing. As we know Cloud computing is a virtualized environment that will provide scalable web services to the users, which also faces stringent security challenges. However, access control will solve this problem but we cannot directly apply the traditional access control model. Because it will not handle the vulnerability and uncertainty caused by open environment on Cloud computing. It guaranteed the data security by ensuring the security and reliability. So combining Trust Management with Mutual Trust model, MTBAC is introduced. This model, initially quantifies cloud user's trust by framing different trust attributes and sub attributes by using AHP. Then, it evaluates the cloud service behaviour mechanism which is based on the ACS and MTBAC Structure. However the performance is based on the parameter setting of (α, β), it can be further optimized to increase the accuracy. Our proposed system optimizes these parameter values further based on genetically algorithm is described as follows.

## 3. Proposed GM-ACO Model

The proposed GM-ACO model classifies the trusted service provider based on the rule generated by ant in ACO and parallel Genetic Algorithm optimizes the control parameter values to improvise the trust accuracy rate, as mentioned.

### 3.1 ACO for Classification

In this classification model, ACO defines a heuristic search technique to find the rule information of data sets and a step by step covering strategy to rule discovery[9]. After its discovery, the dataset set is further grained by removing the dataset examples covered by the discovered rule. This process continues iteratively until the training dataset is almost empty or empty. Every rule in ACO has a predicted class[10] and antecedent condition part and the condition part is the combination of tuples like (attribute, operator and value). The 'equal to' operator is used in all the experiment, assuming that all the attributes in ACO is categorical. Here, we assume a rule condition approximately as which is equal to, where is the attribute, attribute value is represented as in domain. The probability of adding this condition to ant constructing rule is given by the Equation (1), as:

$$P_{i,j}(t) = \frac{\tau_{i,j}^{\alpha}(t) \cdot \eta_{i,j}^{\beta}}{\sum_{i}^{a} \sum_{j}^{b_i} \tau_{i,j}^{\alpha}(t) \cdot \eta_{i,j}^{\beta}, \forall i \in I} \quad (1)$$

From above Equation it shows that, represents the pheromone concentration that is currently available between the connection from attribute i to value of j, attribute sets that are not used by ants are represented as 'I', then represents the heuristic value for which is problem dependent, 'a' gives the total attribute and shows the total number of attribute values at attribute i domain. The necessary part of ant's pheromone is initialized equally at all the cells of pheromone table based on the Equation (2), given as follows:

$$\tau_{i,j}(t=0) = \frac{1}{\sum_{i=1}^{a} b_i} \quad (2)$$

- 'A' represents number of attributes.
- ''Represents the number of values in the attribute i domain.

The heuristic value taken and passed as information exists in theory measure for the term quality which is added to the current rule. The quality is calculated in terms of entropy, in order to prefer this term to others and it is evaluated by Equations (3) and (4):

$$\eta_{i,j} = \frac{log_2(k) - infoT_{i,j}}{\sum_{i}^{a} \sum_{j}^{b_j} log_2(k) - infoT_{i,j}} \quad (3)$$

$$info\ T_{i,j} = -\sum_{w=1}^{k}\left[\frac{freqT_{i,j}^{w}}{|T_{i,j}|}\right] * log_2\left[\frac{freqT_{i,j}^{w}}{|T_{i,j}|}\right] \quad (4)$$

Here represents partition cases, partition has the cases there attribute has a value of, 'k' represents the total number of class, 'a' gives the total attribute and shows the total number of attribute values at attribute i domain and gives number of partition cases with class w. If the value of *info* is higher, then the ant will less likely to choose, that is added to the current partial rule. To increase the accuracy and understandable property of rule, rule pruning is initialized immediately after ant completes the construction of rule[9]. Also, the rule pruning process will remove the term iteratively which increases the rule quality further to maximum. The Quality 'Q' of rule is calculated based on the Equation (5), given as:

$$Q = (TruePos/(TruePos + FalsePos)) * (TrueNeg/(FalsePos + TrueNeg)) \quad (5)$$

- *TruePos* → Number of cases covered and having same class predicted by rule.
- *FalsePos* → Number of cases covered and having different class predicted by rule.
- *FalseNeg* → Number of cases not covered and having class predicted by rule.
- *TrueNeg* → Number of cases not covered and having different class predicted by rule.

The next important process is pheromone updating that is taken care by ant after the completion of rule construction and is based on the Equation (6), shown as:

$$\tau_{i,j}(t+1) = \tau_{i,j}(t) + \tau_{i,j}(t).Q \quad \forall i,j \quad (6)$$

The pheromone amount has to be decrease from each, which is not occurred in the rule construction. The pheromone reduction of an unused term is carried out by dividing the value of each by a summarization of all values.

## 3.2 GM-ACO Algorithm

The proposed GM-ACO algorithm is a hybridized prototype based on the procedure of both Ant Colony Optimization and Genetic Algorithm and it proceed as follows:

- Initialize the variable P for population, Pc for crossover probability, Pm mutation probability, NCmax maximum number of cycles and function Rw for roulette-wheel selection.
- Initialize for loop for iteration i = 0 to population size, generate feasible population based on the objective function.
- Select two parents X and Y from the generated population using Roulette-Wheel selection, then generate offspring by applying Crossover operation on two selected parents X, Y based on Pc Crossover rate.
- Mutate the array of X, Y by each bit based on the mutation rate Pm and update the population P, if the best solution is not attain, then increment i+1 and go to step (2).
- Obtain the (Alp,Bet) values from the Pxy for every iteration based on the maximum fitness and pass the values as probability information to ant, which construct rules.
- Initialize Dataset (D) and for condition, if the Dataset (D) is greater than maximum uncovered sets do the operation, Initialize Pheromone, Probability information (Alp,Bet) ant_size( ) and R as Rule set..
- For iteration j = 0 to ant_size do the operation, create rule based on Dataset (D) and save it as Rn, then prune the rule (Rn). After removing unwanted data from Rn, evaluate (Rn) based on the probability selection Pij.
- Store the Rn as an update the pheromone ( ), then replace the global best with iteration best values. On the next generation if the iteration best value exceed the maximum global best value then terminate the loop.
- Calculate D based on the difference from the original training set to the maximum covered set, which is based on the global best iteration value and training set. Return T rate based on the evaluated dataset and global best iteration.

The procedure is illustrated in Figure 1 and it proceeds based on the given steps. This proposed algorithm works on the background and calculates cloud service nodes trust value, the trust dataset are loaded from the node behaviour trust database. The pheromone update procedure, updates the calculated trust value in order to maintain the accuracy because trust environment is dynamic. As mentioned earlier, the control parameter (α and β) values are optimized using Genetic Algorithm and this algorithm applies all the values of (α and β) as prob-

ability information to Equation (1) of ACO, in order to evaluate and obtain the trust rate with higher accuracy.

**Input:** Dataset **D**, with Trust Values;

**Output:** Trust Accuracy **T**;

**Genetically modified – Ant Colony Optimization:**

i←0; j←0;

Initialize P (t) = 0, Pc, $R_{best}$ =0, Pm, NCmax=1000, Rw ( ) =0,

**For each i=0** to Pop_Size

Generate P, Tournament-selection Ts (X, Y) ← P;

Pxy ← Crossover (X, Y); Mutate (Pxy) at rate of Pm;

P←Pxy; update P;

**get** (Alp, bet)← Pxy; i ←i+1;

While (D) > uncovered **do**

**Initialize** Pheromone ( ), probability information (Alp, Bet), Rule set← R, ant_size ( );

**While i** < NCmax and nostagnation **do**

**For j←1** to ant_size **do**

Rn ← create Rule (D), Prune (Rn);

Evaluate (Rn), $R_{iter-best}$ ←Rn ;

**End for**

Update pheromone ($R_{iter-best}$); Evaluate ($R_{iter-best}$);

$R_{global-best}$ ← $R_{iter-best}$ ;

i ← i+1;

**end while**

**D** ← D – covered ($R_{global-best}$, D);

**T** ←calculate (D, $R_{global-best}$ );

**End while**

**Return T;**

**Figure 1.** GM-ACO algorithm.

## 4. Experimental Result

The experimental result of the proposed GM-ACO model is validated by classification accuracy of real time data, which is cloud Armor dataset[11] that contains the trust values of different cloud service provider based on the consumers' feedback. Initially, the ACO is process with the default parameter setting[12] as (α = 1, β = 5) and accuracy rate for 10 fold cross validation are observed and plotted in a line graph is shown in Figure 2. Then, the same cloud Armor dataset is processed on proposed GM-ACO[13] model, here the parameter values (α and β) of ACO which influence the performance are optimized and set by Genetic Algorithm[14] and the cross validation result of test set is observed and plotted is shown in Figure 3. The accuracy rate for both ACO[15], GM-ACO model are observed and tabulated and all the cross validation accuracy rate are plotted in a graph against each other as shown in Figure 4. From the graph it is observed that ACO starts with the higher accuracy rate[16] at first fold of cross validation than GM-ACO model and maintains similar accuracy at some points, then the overall accuracy rate of GM-ACO is slightly higher than existing ACO model. It shows that Trust accuracy rate[17] is improved on compared with the existing ACO.

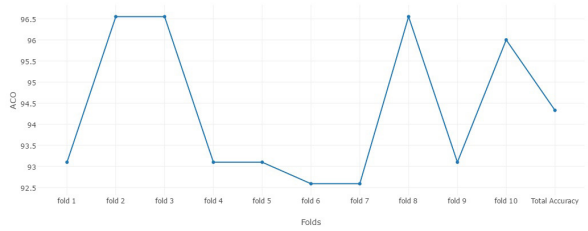


**Figure 2.** ACO trust accuracy.

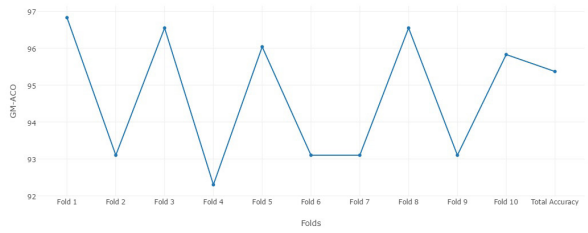


**Figure 3.** GM-ACO trust accuracy.

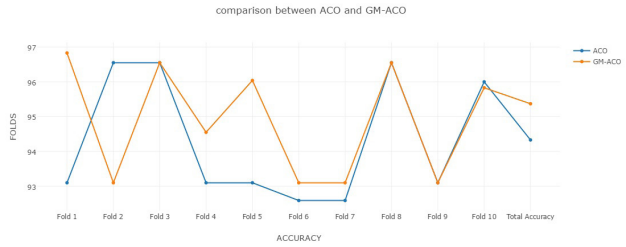


**Figure 4.** Comparison between ACO and GM-ACO.

## 5. Conclusion

The proposed GM-ACO model will solve the problem of traditional access control model, by enabling the users to select the trusted service node with higher accuracy rate from the cloud environment. As mention earlier, the trust accuracy rate of ACO is improved by using the Genetic Algorithm to optimize the parameter values (α and β), because the performance of ACO is based on these parameter values. It is validated based on classifier accu-

racy rate of dataset which contains the trust data. Existing ACO model with the predefined parameter values of ($\alpha = 1$ and $\beta = 5$) are processed by loading Cloud Armor dataset, which is a real time dataset that contains the trust values of different real Cloud Service Providers on cloud environment and trust accuracy rate are observed. Then, in proposed GM-ACO the default parameter values are optimized and set by the Genetic Algorithm and processed with the same cloud Armor dataset. The number of instances in dataset will affect the trust accuracy rate. So, the dataset should be pre-processed efficiently before loading into the model. The observed trust accuracy rate of GM-ACO is higher than the existing ACO model. Thus, the proposed GM-ACO model increases the trust accuracy and rank the cloud service provider with higher trust rate, the user can select the service provider with higher trust rate and improves the security.

## 6. References


1. Lin G, Wang D, Bie Y, Lei M. MTBAC: A Mutual Trust Based Access Control model in Cloud computing. China Communications. 2014 Apr; 11(4):154–62.
2. Marsh SP. Formalizing trust as a computational concept. 1994. Available from: http://citeseer.ist.psu.edu/viewdoc/download;jsessionid=99A42EFFF9BCB0B1D00976F31858085E?doi=10.1.1.102.8227&rep=rep1&type=pdf
3. Blaze M, Feigenbaum J, Lacy J. Decentralized Trust Management. Proceedings of the IEEE Symposium on Security and Privacy; 1996. p. 164.
4. Zimmarmann P. PGP user's guide. Cambridge: MIT Press; 1995.
5. Jameel H, Hung LX, Kalim U. A trust model for ubiquitous systems based on vectors of trust values. ISM '05 Proceedings of the 7th IEEE International Symposium on Multimedia; USA. 2005. p. 674–9.
6. Jincui C, Liqun J. Role-based access control model of Cloud computing. Energy Procedia. 2011; 13:1056–61.
7. Santos N, Gummadi KP, Rodrigues R. Towards trusted Cloud computing. Proceedings of the Conference on Hot Topics in Cloud computing; USA. 2009. p. 1–11.
8. Yoon JP, Chen Z. Using privilege chain for access control and trustiness of resources in Cloud computing. Springer Berlin Heidelberg; 2010 Jul. p. 358–68.
9. An Ant Colony Algorithm for Classification Rule Discovery. 2001. Available from: https://www.researchgate.net/publication/2377588_An_Ant_Colony_Algorithm_for_Classification_Rule_Discovery
10. Liu B, Abbass HA. Classification rule discovery with Ant Colony Optimization. IEEE Computational Intelligence Bulletin. 2004 Feb; 3(1):1–5.
11. Ant Colony Optimization.2004. Available from: https://mitpress.mit.edu/books/ant-colony-optimization.
12. A Hybrid Approach of GA and ACO for TSP. 2004. Available from: http://ieeexplore.ieee.org/document/1341948/
13. Liang Z, Sun J. A novel multiple rules sets data classification algorithm based on Ant Colony Algorithm. Applied Soft Computing. 2016 Jan; 38:1000–11.
14. Noor TH, Sheng QZ, Ngu A, Alfazi A. Cloud Armor: A platform for credibility-based Trust Management of cloud services. CIKM '13 Proceedings of the 22nd ACM International Conference on Information and Knowledge Management; 2013 Oct. p. 2509–12.
15. Ye Z, Zheng Z. Configuration of parameters α, β, $\boldsymbol{\rho}$ in Ant Algorithm. Geomatics and Information Science of Wuhan University. 2004 Mar; 29(7):597–601.
16. Beulah S, Dhanaseelan FR. Survey on security issues and existing solutions in cloud storage. Indian Journal of Science and Technology. 2016 Apr; 9(13):1–8.
17. Kirubakaramoorthi R, Arivazhagan D, Helen D. Analysis of Cloud computing Technology. Indian Journal of Science and Technology. 2015 Sep; 8(21):1–3.