A Trustworhty Model for Reliable Cloud Service Discovery

Akinwunmi A.O.,
Department of Computer Science and Information Technology Bowen University Iwo Nigeria

Olajubu E.A.,
Department of Computer Science and Engineering Obafemi Awolowo University Ile-Ife Nigeria

Aderounmu G.A.,
Department of Computer Science and Engineering Obafemi Awolowo University Ile-Ife Nigeria

ABSTRACT
Cloud computing is a new model for delivering new applications and services. Its adoption is gaining ground because most of the services provided by the cloud are of low cost and readily available for use. Despite many promises by the cloud service providers, users remain much concerned about the general risk associated with the adoption of the cloud. The availability of many cloud service providers on one hand promotes competition in the cloud market and gives end users more freedom to choose the best cloud provider however it became a tedious and time consuming task for potential cloud users to evaluate and compare the available cloud offerings in the market. Hence, discovering a reliable service is a daunting task. This research proposed a trustworthy model for reliable cloud service discovery.

General Terms
Service Discovery, Cloud Computing, Bayesian Network

Keywords
Trust, Providers, Users, Cloud Services, Resources

1. INTRODUCTION
Cloud Computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) [1] that can be rapidly provisioned and released with minimal management effort or service provider interaction[2].

Cloud Computing is an evolving generation in computing. This new service model is related to previous, popular distributed computing initiatives such as web services and grid computing. It relies on providing and sharing computing resources rather than having local servers or personal devices to handle applications [3]. Cloud Computing [4,5,6] has emerged as the next generation platform for hosting business and scientific applications. It offers infrastructure, platform, and software as services that are made available as on-demand and subscription-based services in a pay-as-you-go model to users.

The main idea is to use the existing computing resources in order to bring all available services to the cloud and make it possible to access those services regardless of time and location[3]. The main idea in Cloud Computing is that it covers all the range of users, from home users that use Cloud Computing to approve their works better and IT staffs and enterprise managers that use Cloud Computing for optimizing, planning and implementing their enterprises [3].

All these cloud services and service providers have varied levels of quality and also, due to the anonymous and heterogeneous nature of the cloud systems, some unscrupulous providers may tend to cheat unsuspecting users. In order for clients to access resources they must be discovered. Clouds and their computational resources are not easy to discover and it is difficult to select and use services [7]. Hence, discovering a reliable cloud service provider is prone to much vulnerability. We trust a system less if it gives us insufficient information about its abilities, and trust simply means act of faith that relies on confidence that something will surely be delivered as promised [8,9]. Mere claims such as “secure cloud” or “trust me” cannot help as such to indicate the trust level of users in the providers unless sufficient information is presented with the services. Trust between the service provider and the user is one of the main issues cloud computing faces today. There is no way for the user to be sure of whether the management of the service is trustworthy, and whether there is any risk associated with insider attacks, hence this research.

This work was organized into sections. Section one gave the background to the problem. Section two focused on the related work. Section three presented the theoretical framework for the work while the proposed model was dealt with in section four. Section five discussed the simulation and evaluation of the model while section six concluded the work.

2. RELATED WORK
Service discovery is a subdivision of resource discovery in such a way that service discovery should be seen as specifically as capability to find specific services such as applications or well defined network services that are not pure abstractions [10].

Locating resources in a large scale, heterogeneous network is a not a simple task. There may be several providers, each one with different attributes, thus offering a variety of different levels of user-requirement fitness [11]. In order to get notification about the status and the availability of these resources, a Service Discovery (SD) mechanism must be installed. All Service Providers will require registering in such a mechanism, along with all the necessary information such as URL, rates, compatibility and interface, so that their services are advertised to the end user [11]. The information service then makes all of these available to potential clients, by matchmaking the request with the available resources and returns back the results [11].

Service discovery is required to support any service infrastructure like cloud computing. A large-scale, multi-domain service infrastructure requires a service discovery system that is open, scalable, robust and efficient to a greater extent than a single-domain system [12]. In a multi-domain

environment such as cloud that hosts thousands of service instances, both correctness and completeness of discovery are more important than in a small, restricted environment [12]. Completeness of service discovery requires the system to retrieve relevant results across the various domains that constitutes the system.

Service discovery in cloud environment is made challenging by the potentially large number of available heterogeneous services and a large number of service providers. Service discovery is further made complex by varied level of quality of service (QoS) offered by the different providers. Security concern is also a big challenge while looking for an appropriate cloud service. Several researches attempted to address the challenge of cloud service discovery.

Kang and Sim [13] presented a four-stage, agent-based Cloud service discovery protocol. Software agents involving negotiation agents, brokering agents and information agents were used for bolstering the resource management system because of their scalability and adaptability with high level abstraction for modelling of complex software systems. Zhao *et al*. [14] proposed a Service Provider Search Engine (SPSE) innovative service selection algorithm that could find the appropriate service considering the user's multiple QoS (Quality of Service) requirements. Cortázar *et al*.[15] proposed a cloud computing ontology that facilitates a semantic identification, discovery and access to the services in the cloud. Wang *et al*. [16] proposed a mixed integer programming model to select optimal services. The model first computed the QoS uncertainty to prune redundant services in order to extract reliable services. Reshma and Balaji [17] proposed ontological model for service publication, discovery, and selection using Software as a Service (SaaS). Chen and Li [18] proposed a service registry model named as SRC (Service Registry on Cloud) which was an extension of the keywords based service registry model and deployed as a cloud application to provide behaviour-aware and QoS-aware service discovery services. Kang and Sim [19] introduced Cloudle—a multi-criteria Cloud service search engine that supported matching algorithm of three kinds of requirements which are (i) functional requirement, (ii) technical requirement, and (iii) cost requirement.

These related works had proposed different approaches to address the problem of cloud service discovery but they were not adequate in enough in handling the uncertainty associated with the cloud environment, hence the need for this research.

3. THEORETICAL BACKGROUND

The theoretical background for the model was based on Bayesian Network. Bayesian Networks give an intuitive way to depict the joint probability distribution over a set of variables. The random variables can be depicted as nodes in a directed acyclic graph, and links express causality relationships between the variables. This is thus a representation particularly well-suited to look for correlations among random variables in cloud environment to ensure trustworthiness of the environment.

4. PROPOSED MODEL

Having a cloud service discovery model that is reliable is essential for a global adoption of cloud paradigm. How the cloud computing is adopted does not depend only on technical issues but also on socio-technical issues of security. Hence, the challenges that must be addressed before cloud computing adoption are: i) to provide adequate access to cloud services and ii) to ensure that a cloud service discovery process is reliable. Cloud computing should take into consideration the issue of gaining access to the enormous opportunities created by the technology which is dependent on the ease of access as well as the reliability of the cloud environment.

Cloud computing was viewed as a complex aggregation of computing resources from different domains with different administrative policies but having immense benefits that could enhance the mode of computing. Analyzing the problem of discovering a reliable cloud service reveals that the process involves a series of events which are related. The process of service discovery can be broken down into the following sub-functions.

i. Publish function: a service provider uses this function to publish information about the provided services. This function advertises the availability of the services for possible discovery.
ii. Mediating function: It provides an intermediary for the users and provider to interact. Service Oriented Architecture (SOA) is needed to realize this function.
iii. Search function: this function does the actual location of the availability of the service requested by interacting with the various service directories for a possible occurrence of the service needed.
iv. Comparison function: this function ascertains the degree of match between the service requested and services found.
v. Reputation consideration function: this function determines the level of trust of the service provider of the exact service found for the service requested so that users can be assured of reliable service.
vi. Access Granting function: this function makes the trusted service available to the user for actual use.

All of these functions are required for effective cloud service discovery which enable user to request, discover, and use cloud services that are reliable. A careful analysis of reliable cloud service discovery mechanism revealed that it entails a lot of complexity because of the dynamism of the cloud environment and this must be addressed to achieve the aim of this research.

4.1 Model Description

A model that can handle the complexity of the cloud is a probabilistic model which is a description of an uncertain situation. The probabilistic model captures the relationship between each of the variable or functions identified above.

The model variables were identified and the links between them were established as stated below. Model variables are the functions that are relevant to achieving reliable cloud service discovery and these are: Publish, P_f; Mediating, M_f; Search, S_f; Comparison, C_f; Reputation Consideration, R_f and Access Granting; A_f functions as described in the earlier section. Hence the model is stated as

$$M = f(P_f, M_f, S_f, C_f, R_f, A_f) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 1$$

For the model to work all the functions must be available. Then establishing links among the variables of M is viewed as an occurrence of P_f, followed by the occurrence of M_f, then followed by the occurrence of S_f, followed by the occurrence of C_f, followed by the occurrence of R_f and lastly followed by

the occurrence of A_f. The model flowchart is as shown in Fig 1 below in the next column.

The Bayesian network was adopted for formulating the model. The probabilistic dependencies between the cloud service discovery sub-functions were captured as conditional probability. These sub-functions are the variables in the cloud service discovery domain. Hence the model was pictured as a serial connected Bayesian Network which is represented as an acyclic directed graph G[20].

Where G= G(V,E) , this consist of V= $\{p_f, m_f, s_f, c_f, r_f, a_f\}$, the set of nodes and E= $\{p_f m_f, m_f s_f, s_f c_f, c_f r_f, r_f a_f\}$, the set of edges or arc. The graphical representations of the model that capture the relationships between the model's variables is as shown in Figure 2 below in the next column.

A node represents a sub-function. An edge or arc represents a causal relationship or dependency between two sub-functions. Bayesian network makes it possible to model and reason about the uncertainty involved in cloud service discovery.

This model was then expressed as a Bayesian Network and its joint probability density function was written as a product of the individual density functions, conditional on their parent sub-functions.

$P(x_1,\ldots,x_n) = \prod_{i=1}^{n} P_i(X_i \mid \text{Parents}(X_i))$ ……..…….....2

The model must fulfill the local Markov property in which each sub-function is conditionally independent of its non-descendants given its parent sub-function.

Using the chain rules, the full joint probability distribution for the model is:

$$P(P_f, M_f, S_f, C_f, R_f, A_f) =$$

$$P(A_f \mid R_f) * P(R_f \mid C_f) * P(C_f \mid S_f) * P(S_f \mid M_f) * P(M_f \mid P_f) * P(P_f) \ldots 3$$

For the sake of notation

Let $X_1 = P_f$, $X_2 = M_f$, $X_3 = S_f$, $X_4 = C_f$, $X_5 = R_f$ and $X_6 = A_f$

Hence P(M)= $P(X_1, X_2, X_3, X_4, X_5, X_6)$

$$= P(X_6 \mid X_5) * P(X_5 \mid X_4) * P(X_4 \mid X_3) * P(X_3 \mid X_2) * P(X_2 \mid X_1) * P(X_1) \ldots 4$$

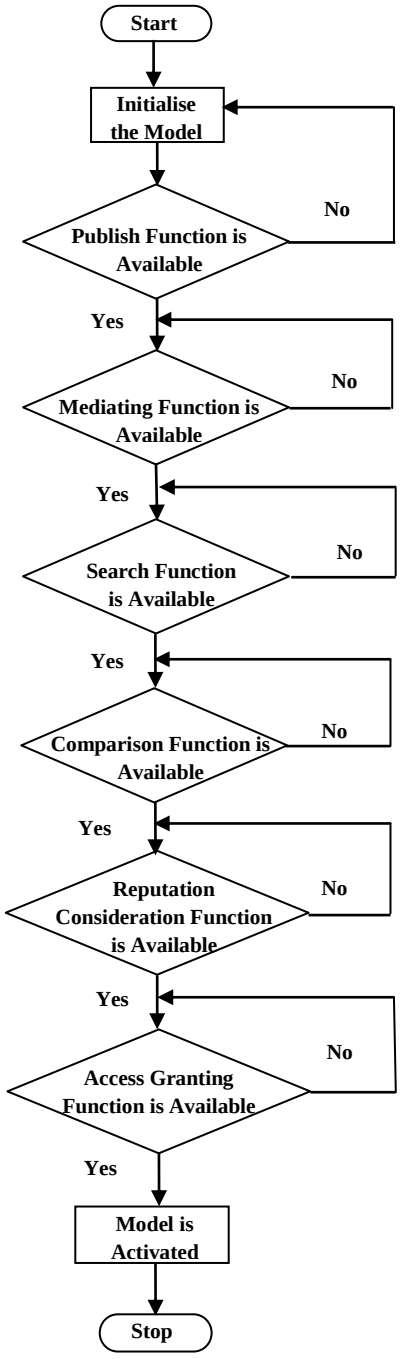

Fig 1: Model flowchart

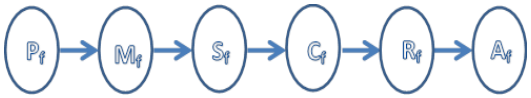

Fig 2: Cloud service discovery Bayesian model

4.2 Model Parameterizing

This entails assigning states and probabilities to each variable. The states for each node depict the potential values or conditions that the node can assume. It was assumed that all nodes represent binary variables with values in the set {0,1}. The nodes can be in either of the two states namely 0 or 1 which indicates the availability of the node. Each node X_i has a conditional probability distribution P(X_i | Parents(X_i)) that quantifies the effect of the parents on the node. The

parameters are the probabilities in these conditional probability tables (CPT).

A convenient option of the parameters in this case was given by $\Theta_{i\,j\,\prod} = P_i(X_i=j|\prod i=\pi)$, $j \in \{0,1\}$ where π is any possible instantiation of the parents of X_i . The first subscript in $\Theta_{i\,j\,\prod}$ refers to the node number, the second subscript referred to the state of the node, and the remaining subscripts referred to the parents' instantiations. To simplify the notation in cases where a variable Xi does not have parents, we use $\Theta_{i\,j}$ to denote Pi(Xi = j); $j \in \{0,1\}$. The structure of the probability distributions in table 1 below imply that the joint probability distribution of the six variables depends on 22 parameters $\Theta=\{\Theta_{i\,j\,\prod}\}$.

Table 1: Probability distributions

Node		Parameters	
X_i	Parents	State=0	State=1
X_1	None	$\Theta_{10}=P(X_1=0)$	$\Theta_{11}=P(X_1=1)$
X_2	X_1	$\Theta_{200}=P(X_2=0\mid X_1=0)$ $\Theta_{201}=P(X_2=0\mid X_1=1)$	$\Theta_{210}=P(X_2=1\mid X_1=0)$ $\Theta_{211}=P(X_2=1\mid X_1=1)$
X_3	X_2	$\Theta_{300}=P(X_3=0\mid X_2=0)$ $\Theta_{301}=P(X_3=0\mid X_2=1)$	$\Theta_{310}=P(X_3=1\mid X_2=0)$ $\Theta_{311}=P(X_3=1\mid X_2=1)$
X_4	X_3	$\Theta_{400}=P(X_4=0\mid X_3=0)$ $\Theta_{401}=P(X_4=0\mid X_3=1)$	$\Theta_{410}=P(X_4=1\mid X_3=0)$ $\Theta_{411}=P(X_4=1\mid X_3=1)$
X_5	X_4	$\Theta_{500}=P(X_5=0\mid X_4=0)$ $\Theta_{501}=P(X_5=0\mid X_4=1)$	$\Theta_{510}=P(X_5=1\mid X_4=0)$ $\Theta_{511}=P(X_5=1\mid X_4=1)$
X_6	X_5	$\Theta_{600}=P(X_6=0\mid X_5=0)$ $\Theta_{601}=P(X_6=0\mid X_5=1)$	$\Theta_{610}=P(X_6=1\mid X_5=0)$ $\Theta_{611}=P(X_6=1\mid X_5=1)$

Hence, the joint probability distribution depends on the collection of the 22 parameters:

$\Theta_1 = \{\theta_{10}, \theta_{11}\}$ ……..…5

$\Theta_2 = \{\theta_{200}, \theta_{201}, \theta_{210}, \theta_{211}\}$ ……6

$\Theta_3 = \{\theta_{300}, \theta_{301}, \theta_{310}, \theta_{311}\}$ ……7

$\Theta_4 = \{\theta_{400}, \theta_{401}, \theta_{410}, \theta_{411}\}$ ……8

$\Theta_5 = \{\theta_{500}, \theta_{501}, \theta_{510}, \theta_{511}\}$ ……9

$\Theta_6 = \{\theta_{600}, \theta_{601}, \theta_{610}, \theta_{611}\}$ 10

4.3 Model Evaluation

It is important to evaluate the model resulting from parameterizing process, and this was done quantitatively using sensitivity analyses. Sensitivity analysis tests the sensitivity of model outcomes to variations in model parameters [21]. Sensitivity analysis in Bayesian Networks (BNs) can detect the sensitivity of outcome probabilities to changes in input nodes or other model parameters, such as changes in node's type of states [20]. Recall that equation 4 is

$$P(M)= P(X_6 \mid X_5)*P(X_5 \mid X_4)*P(X_4 \mid X_3)*P(X_3 \mid X_2)*P(X_2 \mid X_1)*P(X_1)$$

Substituting the collection of the parameters from the table above into

$$P(M)= P(\Theta_1)*P(\Theta_2 \mid \Theta_1)*P(\Theta_3 \mid \Theta_2)*P(\Theta_4 \mid \Theta_3)*P(\Theta_5 \mid \Theta_4)*P(\Theta_6 \mid \Theta_5) \quad \ldots 11$$

The states of the variables are very sensitive to the operation of the model considering the joint probabilities outcomes from the various possible parameters.

Sensitivity analysis tests the sensitivity of model outcomes to variations in model parameters. Sensitivity analysis in BNs can measure the sensitivity of outcome probabilities to changes in input nodes or other model parameters, such as changes in node's type of states and their coarseness. Sensitivity analysis was performed using two types of measures; entropy and Shannon's measure of mutual information [22].

Entropy measure

The entropy measure was based on the assumption that the uncertainty or randomness of a variable *X*, characterized by probability distribution *P(x)*, can be represented by the entropy function [21] H(X):

$$H(X)= -\sum_{x \notin X} P(x).\log P(x)$$

Hence in this work, the joint probability distribution for the random variables X1, X_2, X_3, X_4, X_5, and X_6 depend the collection of the parameters Θ_1, Θ_2, Θ_3, Θ_4, Θ_5 and Θ_6

The entropy measures of these variables were stated as follows:

$$H(X_1)= -\sum_{\Theta_1 \notin X_1} P(\Theta_2).\log(\Theta_2) \quad \ldots\ldots 12$$

$$H(X_2)= -\sum_{\Theta_2 \notin X_2} P(\Theta_2).\log(\Theta_2) \quad \ldots\ldots 13$$

$$H(X_3)= -\sum_{\Theta_3 \notin X_3} P(\Theta_3).\log(\Theta_3) \quad \ldots\ldots 14$$

$$H(X_4) = -\sum_{\Theta_4 \notin X_4} P(\Theta_4).\log(\Theta_4) \quad \ldots\ldots 15$$

$$H(X_5)= -\sum_{\Theta_5 \notin X_5} P(\Theta_5).\log(\Theta_5) \quad \ldots\ldots 16$$

$$H(X_6)= -\sum_{\Theta_6 \notin X_6} P(\Theta_6).\log(\Theta_6) \quad \ldots\ldots 17$$

Shannon's measure

Shannon's measure of mutual information was used to assess the effect of collecting information about one variable (*Y*) in reducing the total uncertainty about variable *X* using: $I(Y, X) = H(Y) - H(Y|X)$ where I(Y,X) = the mutual information between variables [20].

The Shannon's measures of mutual information for these variables were stated as follows:

$$I(X_1, X_2) = H(X_1) - H(X_2 \mid X_1) \quad \ldots\ldots 18$$

$$I(X_2, X_3) = H(X_2) - H(X_3 \mid X_2) \quad \ldots\ldots 19$$

$$I(X_3, X_4) = H(X_3) - H(X_4 \mid X_3) \quad \ldots\ldots 20$$

$$I(X_4, X_5) = H(X_4) - H(X_5 \mid X_4) \quad \ldots\ldots 21$$

$I(X_5, X_6) = H(X_5) - H(X_6 \mid X_5)$ ……22

The mutual information between the variables must be such that $I(X_1, X_2) \geq I(X_2, X_3) \geq I(X_3, X_4) \geq I(X_4, X_5) \geq I(X_5, X_6)$ for effective model evaluation.

4.4 Model Estimation

Learning about a specific model, M, that best accounts for all the states and probabilities was accomplished by maximizing the parameters distribution over the model which according to Bayes' rule is

$P(M|\Theta) \alpha P(\Theta|M) P(M)$ .....…..……..23

Determining the prior, $P(M|\Theta)$ was full of uncertainty, then the model that maximizes the likelihood, $L(\Theta|M)$ was chosen. The likelihood is proportional to the probability of observing the model, treating the parameters of the distribution as variables and the model as fixed.

The best estimator $\hat{\Theta}$, is whatever value of $\hat{\Theta}$ that maximizes

$L(\hat{\Theta}|M) = * P(M|\hat{\Theta})$ ……24

One is typically looking for the parameter, $\hat{\Theta}$ that maximize the likelihood of observing the model

Based on the proportional relationship as expressed in 24, $\hat{\Theta}$ that maximizes $L(\hat{\Theta}|M)$ will also maximize $P(M|\Theta)$ which is the probability of the observed model.

The likelihood of the model is the product of the likelihood of the individual parameter item

$L = L_1 * L_2 * L_3 * L_4 * L_5 * L_6 = \prod_{k=1}^{6} L_k$ ……..…25

$P(M \mid \hat{\Theta}_1) * P(M \mid \hat{\Theta}_2) * P(M \mid \hat{\Theta}_3) * P(M \mid \hat{\Theta}_4) * P(M \mid \hat{\Theta}_5) * P(M \mid \hat{\Theta}_6)$
……..……26

$= \prod_{k}^{6} P(M \mid \hat{\Theta}_k)$ ….....27

Where $\hat{\Theta}_k$ denotes the best individual parameter item estimator for each of the model variables

The likelihood function in 27 was expressed as log likelihood function as shown in 28

$\ln L = \Sigma_{k=1}^{6} InP(M \mid \hat{\Theta}_k)$ ……...…...28

5. MODEL SIMULATION AND EVALUATION

The main entities in the cloud system were the users, brokers and providers. The interaction for consideration in this work is focuses on the users and the providers. The concept supposes that each entity in the model has different qualities which are not completely independent and these qualities are correlated together with using specification and generalisation i.e. some qualities or attributes in lower abstraction level can create one or more common qualities in higher abstraction level.

In order to test the behavior of the proposed model with trust integration in the process of cloud service discovery, CloudAnalyst which is a tool useful to model and analyze large scale cloud computing development [23] was used. The existing model was referred to as the process of cloud service discovery without trust integration while the proposed model has trust integrated in the process of cloud service discovery. The user entities were modelled as User Bases. Provider entities were modelled as Data centres. Broker entities were modelled as Service Broker Policy type of reconfigure dynamically with load. The Trust integration in the process was implemented using the Throttled load balancing policy for the proposed model . The round robin policy was used to model lack of trust integration in the process of cloud service discovery in the existing model.

The evaluation parameter used was reliability which is the measure of how consistency the model will operate repeatedly under the same given operating conditions. The output from the system was used to evaluate this metrics. Determining the reliability of our proposed and existing models required that one observed how the models performed specified function under specified conditions for a specified period of time. Hence, the reliability in this regard was expressed as a function of how consistent are the models in their roundtrip time (RTT) versus the users load during operation. Hence the equation 29 below

Reliability = f (RTT Consistency, Users Load)……29

To determine this, 10 different trials were ran for both the existing and proposed models using the roundtrip time (RTT) to verify the behaviour of the models. RTT is the time taken for the user request to get to the service provider and its response back to the user. In the simulation we had 5,10,15,20, 25, 30 and 35 user entities with a cloud provider entity. The detailed parameter settings for the experiment are as shown in the Table 2 below.

Table 2: Parameter settings for reliability test

User Entities	Provider Entity	User Growth Factor	Request Growth Factor	Execution Instruction Per Length
5	1	10	10	100
10	1	10	10	100
15	1	10	10	100
20	1	10	10	100
25	1	10	10	100
30	1	10	10	100
35	1	10	10	100

In this simulation the number of user entities was varied while the number of the provider entity was fixed in order to observe the behaviour of the existing model without trust integration and proposed model with trust integration in terms of reliability of their operations.

Tables 3 and 4 below showed the results obtained from the simulation for 10 different trials of RTT ran for both the existing and proposed models respectively.

Table 3: Ten different trials of Round Trip Time (RTT) ran for the existing model

No of User Entities	RTT Trial 1 (ms)	RTT Trial 2 (ms)	RTT Trial 3 (ms)	RTT Trial 4 (ms)	RTT Trial 5 (ms)	RTT Trial 6 (ms)	RTT Trial 7 (ms)	RTT Trial 8 (ms)	RTT Trial 9 (ms)	RTT Trial 10 (ms)
5	299.88	299.96	299.98	299.98	299.98	299.98	299.98	299.98	299.96	299.98
10	300.55	300.22	300.56	300.57	300.56	300.56	300.56	300.56	300.62	300.56
15	300.56	300.57	300.49	300.50	300.49	300.49	300.49	300.49	300.28	300.49
20	301.88	301.68	301.57	301.58	301.57	301.57	301.57	301.57	301.70	301.57
25	300.24	300.57	300.43	300.20	300.43	300.43	300.43	300.43	300.54	300.44
30	300.67	300.69	300.72	300.87	300.72	300.65	300.72	300.72	300.58	300.74
35	299.92	300.03	299.96	300.00	299.96	300.03	299.96	299.96	300.03	299.93

Table 4: Ten different trials of Round Trip Time (RTT) ran for the proposed model

No of User Entities	RTT Trial 1 (ms)	RTT Trial 2 (ms)	RTT Trial 3 (ms)	RTT Trial 4 (ms)	RTT Trial 5 (ms)	RTT Trial 6 (ms)	RTT Trial 7 (ms)	RTT Trial 8 (ms)	RTT Trial 9 (ms)	RTT Trial 10 (ms)
5	299.83	299.83	299.83	299.83	299.83	299.83	299.83	299.83	299.83	299.83
10	300.51	300.09	300.51	300.51	300.51	300.51	300.51	300.51	300.51	300.51
15	300.40	300.43	300.40	300.40	300.40	300.40	300.40	300.40	300.40	300.40
20	301.45	301.58	301.45	301.45	301.45	301.45	301.45	301.45	301.46	301.45
25	300.35	300.48	300.35	300.35	300.35	300.35	300.35	300.35	300.35	300.35
30	300.58	300.57	300.58	300.58	300.58	300.58	300.58	300.58	300.44	300.58
35	299.82	299.97	299.82	299.82	299.82	299.82	299.82	299.82	299.95	299.82

In order to elicit the needed information from the various trials of both existing and proposed models, the Pearson's Product Moment Correlation Coefficient was used to estimate the strength and direction of association between the various trials for each of the models and this result of Pearson's Product Moment Correlation Coefficient computation for the existing and proposed models are as shown in Table 5 below

Table 5: Pearson's product moment correlation coefficient computation for the existing and proposed models

Model	TRIAL 1_2	TRIAL 2_3	TRIAL 3_4	TRIAL 4_5	TRIAL 5_6	TRIAL 6_7	TRIAL 7_8	TRIAL 8_9	TRIAL 9_10
Proposed	0.940725	0.940725	1.000000	1.000000	1.000000	1.000000	1.000000	0.990307	0.990307
Existing	0.952083	0.960058	0.979821	0.979821	0.997532	0.997532	1	0.975389	0.972607

The Graph in Fig.4 showed the Plot of the Pearson's Product Moment Correlation Coefficient against the RTT Trials to verify the behaviour of the proposed and existing models in terms of reliability. From the results, the proposed model shows a high level of correlation in its Pearson's Product Moment Correlation Coefficient than the existing model. This implies that, the proposed model shows a high level of consistency in its behaviour while the existing model's behaviour was less consistent. It is observed that integration of trust into the process of cloud service discovery ensures a high level of reliability. The reliability of the proposed model was better than the existing with closer look at the descriptive statistics of the simulation result in terms of R^2 squared value with the proposed model having a value of 0.393 while the existing model has 0.255.

The result showed that integrating trust into the process of cloud service discovery improves the quality of service in terms of response time, scalability and reliability.

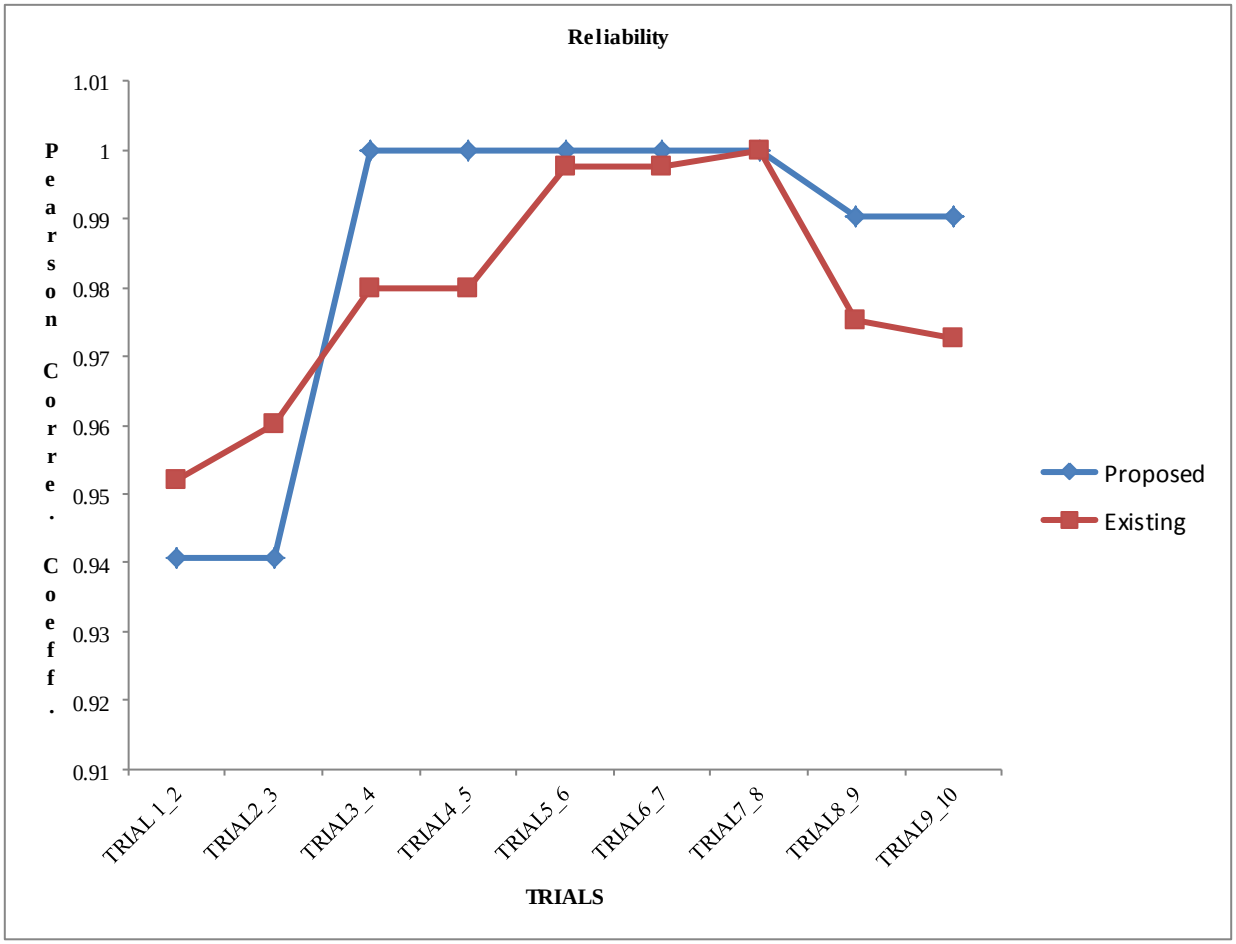

Fig 4: Pearson's product moment correlation coefficient against the round trip time (RTT) Trials

6. CONCLUSION

The research clearly showed that the integration of trust mechanism into the process of cloud service discovery has greatly improved the process and as a result of this, cloud adoption rate is going to be greatly hastened. Users' anxiety about using the cloud will reduce and at the same time, users concerns about the general risk associated with the adoption of the cloud such as security will be less. Trust management as one of the important components in the cloud security when properly addressed will act as an impetus for the growth of cloud computing. Achieving reliability among the various entities within the cloud service discovery system is healthy for its development. Trust among entities in cloud was considered in part in this research. There is the need to also consider other security issues as they affect the cloud service discovery process.

7. REFERENCES

[1] NIST 2014 NIST Cloud Computing Program Available at :http://www.nist.gov/itl/cloud/

[2] NIST 2009 The nist definition of cloud computing National Institute of Standards and Technology Cloud Computing Project Available at: http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v14.doc, Last accessed 11 August 2011

[3] Boroujerdi, M.M. and Nazem, S. 2009. "Cloud Computing: Changing Cogitation about Computing" World Academy of Science, Engineering and Technology Vol.58, pp1112-1116,

[4] Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. and Zaharia, M. 2009 "Above the Clouds: A Berkeley View of Cloud Computing" Technical Report No. UCB/EECS-2009-28, University of California at Berkley, USA,.

[5] IBM, " Reservoir – An ICT Infrastructure for Reliable and Effective Delivery of Services as Utilities". The Reservoir Seed Team IBM Research Report, H-0262, 2008.

[6] Buyya, R., Yeo,C., Venugopal,S., Broberg, J. and Brandic, I. "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility". Future Generation Computer Systems. Vol. 25, No. 6, pp. 599-616, 2009.

[7] Brock, M. and Goscinski, A. 2010. "Toward Ease of Discovery, Selection and Use of Clusters within a Cloud" *2010 IEEE 3rd International Conference on Cloud Computing* pp. 289-296, 2010.

[8] Costa, C. and Bijlsma-Frankema, K. (2007) "Trust and Control Interrelations," *Group and Organization Management*, 32(4): 392 – 406.

[9] Lund, M. and Solhaug, B. (2010). "Evolution in Relation to Risk and Trust Management," *Computer*, May 2010, pp. 49–55.

[10] Meshkova, E., Riihijärvi, J., Petrova, M. and Mähönen, P. 2008. " A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks**"** Computer Networks: The International Journal of Computer and Telecommunications Networking ,Vol. 52 No.11, pp. 2097-2128, August, 2008.

[11] Kousiouris,G., Kyriazis, D., Varvarigou, T., Oliveros, E. and Mandic, P. 2012. "Taxonomy and State of the Art of Service Discovery Mechanisms and Their Relation to the Cloud Computing Stack". In D. Kyriazis, T. Varvarigou, and K. Konstanteli (Eds.), *Achieving Real-Time in Distributed Computing: From Grids to Clouds* ,pp. 75-93, 2012. Hershey, PA: Information Science Reference. doi:10.4018/978-1-60960-827-9.ch005

[12] Ahmed, R., Limam, N., Xiao, J., Iraqi, Y. and Boutaba, R. 2007. "Resource And Service Discovery In Large-Scale Multi-Domain Networks" IEEE Communications Surveys & Tutorials 4th Quarter, 2007.

[13] Kang, J. and Sim, K.M. 2011. "Towards Agents and Ontology for Cloud Service Discovery" *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery* , pp. 483- 490, 2011.

[14] Zhao, L., Ren, Y., Li, M., and Sakurai, K. 2012. "Flexible Service Selection with User-Specific QoS Support in Service-Oriented Architechture" Journal of Network and Computer Applications, Vol.35, No. 3, pp. 962-973, May, 2012

[15] Cortázar, G.O., Zapater, J.J.S. and Sánchez, F.G. 2012. "Adding Semantics to Cloud Computing to Enhance Service Discovery and Access", *Conference Proceedings EATIS'12*, May 23-25, 2012 Valencia, Spain pp. 231-236, 2012.

[16] Wang, S., Zheng, Z., Sun, Q., Zou, H. and Yang, F. 2011. "Cloud Model for Service Selection" *Proceedings of the IEEE Conference on Computer Communications Workshops*, April 10-15, 2011, Shanghai, China, pp. 666-671, 2011.

[17] Reshma, V.K. and Balaji, B.S. 2012. "Cloud Service Publication and Discovery Using Ontology", International Journal of Scientific & Engineering Research Vol.3, No.5, 2012.

[18] Chen, H. and Li, S. 2011. " SRC: A Service Registry on Cloud Providing Behavior-aware and QoS-aware Service Discovery" School of Software, Shanghai Jiao Tong University Shanghai 200240, P.R. China, 2011.

[19] Kang, J. and Sim, K.M. 2010. Cloudle: A Multi-criteria Cloud Service Search Engine *2010IEEE Asia-Pacific Services Computing Conference* pp. 339-346

[20] Ben-Gal, I. 2007, Bayesian Networks, in Ruggeri F., Faltin F. and Kenett R., Encyclopedia of Statistics in Quality and Reliability, Wiley and Sons.

[21] Kragt, M.E. 2009. A beginners guide to Bayesian network modelling for integrated catchment management. Landscape Logic Technical Report No. 9 Department of Environment, Water, Heritage and Arts Australia

[22] Pearl, J. 1988. *Probabilistic reasoning in intelligent systems: networks of plausible inference*, San Mateo, California, Morgan Kaufmann Publishers.

[23] Wickremasinghe, B., Calheiros, R.N. and Buyya, R. 2010. CloudAnalyst: A CloudSim-based Visual Modeller for Analysing Cloud Computing Environments and Applications.