The following publication C. T. Yu, H. C. B. Chan and D. W. K. Kwong, "Discovering Resources in an Intercloud Environment," GLOBECOM 2017 - 2017 IEEE Global Communications Conference, Singapore, 2017, pp. 1-6 is available at https://doi.org/10.1109/GLOCOM.2017.8254488.

Discovering Resources in an Intercloud Environment

Chin Ting Yu*, Henry C. B. Chan*, and Daniel Wai Kei Kwong†
*Department of Computing, The Hong Kong Polytechnic University
†CITIC Telecom International CPC Limited

***Abstract*—With the growing number of cloud users and the popularity of cloud services, there has been considerable interest in studying Intercloud models and architectures. Inspired by the existing IEEE P2302 Intercloud framework and standards, this paper presents an Intercloud architecture for Intercloud communications, focusing on resource discovery. An Intercloud system seeks to facilitate the sharing of resources among different clouds so that cloud resources can be used more efficiently and effectively. In general, resources sharing involves three main steps: discovering resources, selecting resources and allocating resources. This paper seeks to study a DNS-based method for discovering cloud resources. Based on an Intercloud architecture, we also develop analytical models by using queuing theory, and examines the impact of different parameters on the performance of the resource discovery process. The performance of two models, sequential search and broadcast search, are compared and analyzed. The main objectives of the modeling are to evaluate the performance and investigate the constraints of these two models, and determine a better resources discovery method that minimizes the average response time.**

***Index Terms*—Resource discovery, Intercloud, Cloud computing**

I. Introduction

Cloud computing seeks to facilitate the provision of computing resources over the Internet through a flexible service-oriented platform and a utility-based model (i.e., pay-on-demand). These computing resources include virtual machines (VMs), storage, and network resources [1]. With the growing number of clouds, the concept of Intercloud has been proposed. With the aim of utilizing cloud resources more efficiently, Intercloud seeks to facilitate collaboration among cloud providers, allowing them to share cloud resources effectively and efficiently [2] [3]. According to a recent survey [4], one of the major challenges in Intercloud is to find or discover the required resources in a heterogeneous cloud environment that can fulfill diverse user requirements. In other words, resource discovery is one of the fundamental issues in supporting Intercloud.

Today, many companies use cloud services to support their daily operations. While using cloud services is simple, there are also some risk and reliability issues. For example, once a company subscribes to a cloud service provider, the switching cost may be high because all the company data and applications are associated with a single cloud service provider. Furthermore, due to the high dependence on a single cloud service provider, system failure and data loss are also important issues or considerations. Therefore, from a cloud user perspective, it is desirable to use resources from multiple cloud service providers (i.e., data and applications can be moved between heterogeneous cloud service providers in an Intercloud environment using a standard protocol). Resource discovery is an important process to support Intercloud operation. From the cloud providers' perspective, resource discovery can facilitate maintenance of resources (e.g., virtual machines) and provision of better quality of service to users. For example, when a cloud provider cannot provide certain resources, it can seek the help (i.e., resources) from its cloud partners. Furthermore, as it is not cost-effective for cloud providers to set up datacenters in all geographical locations, they can form a cloud alliance to provide better services to their users through an Intercloud environment. Again, successful Intercloud operation relies on effective resource discovery.

In an Intercloud environment, resource discovery is a process of finding available resources to fulfill certain user requirements. In the discovery process, requested resources may include memory and CPUs usage of virtual machines, data storage, geographic location of the datacenter, and other quality of service parameters. Some resource discovery schemes have been proposed in [5] [6] [7], which are based on a centralized architecture. For example, cloud-based agents are used for discovering resources. A centralized database or processing node is employed for discovering and matching the resources by the cloud providers. That means all the resource information are maintained in a centralized database. Hence there is a scalability problem, and performance and reliability issues. Apart from the centralized approach, peer-to-peer approaches have also been proposed for cloud resource discovery [8] [9] [10] [11]. Using a decentralized architecture, resource information is stored in multiple nodes in a distributed manner. Compared to the centralized approach, this is more scalable. However, the operation is more complex as it requires better coordination. Regardless of the centralized approach or decentralized approach, a common issue is related to the management of up-to-date resource information as some resources (e.g., data storage) are highly dynamic. One common assumption in the previous works is that resource information can be published and retrieved by all cloud providers. This assumption may not be applicable in an Intercloud environment. In a competitive environment, it is unlikely for a cloud provider to disclose certain information to competitors. The assumption only holds when the cloud providers have a common goal (i.e., not competitive with one another). In order to solve this problem, we divide the resource information

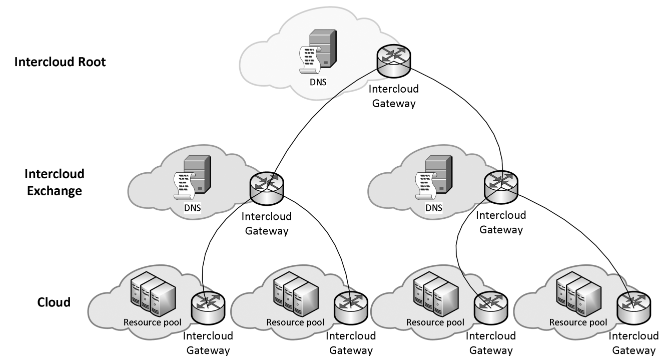

Fig. 1. Intercloud Architecture

into two categories. Static and public information, such as the provider's name, geographic location of the datacenter, and the quality of service parameters, is hosted on the Domain Name System (DNS) in the form of a short for text record (TXT record) [12]. On the other hand, the relatively dynamic and private resource information is not published and cannot be queried directly. A requesting cloud needs to send a request specifying the resource requirements, and receive a positive or negative response from a supplier cloud. Hence, cloud providers effectively form a fully meshed network, because queries must be sent to and processed by the supplier cloud provider. In this paper, two approaches are presented and discussed to facilitate resource discovery in the Intercloud system.

The rest of the paper is organized as below. Section II presents the Intercloud architecture based on IEEE P2302 topology [13]. Section III describes the details of an extension of Intercloud Communications Protocol (ICCP) to support resource discovery. Based on the Intercloud architecture, Section IV presents a queuing model of two resource discovery methods. Analytical and simulation results are shown and discussed in Section V and the paper concludes in Section VI.

II. Intercloud Architecutre

Intercloud seeks to realize the concept of cloud federation, supporting cloud interconnection, resources sharing and resources utilization. With the growing number of various cloud services, there is a need for different cloud service providers to collaborate to provide better services to their users. Inspired by the Internet concept, clouds should communicate and interact with each other using a standard set of protocols. Based on the Intercloud framework proposed by Bernstein [13], our resource discovery protocol is developed based on the following Intercloud architecture. In general, Intercloud architecture consists of three levels of cloud entities (Intercloud Root, Intercloud Exchange and Cloud) and an Intercloud Gateway for communications between different cloud entities. (See Fig. 1)

A. Intercloud Root

Intercloud Root is made up by a cluster/group of root servers, which functions like the Domain Name System (DNS) root servers of the Internet. These root servers are responsible for providing DNS services and working with Intercloud Exchanges to manage clouds to facilitate Intercloud operations, such as resource discovery. In addition, Intercloud Root also functions as a certificate authority for providing trust and authentication services (e.g., issuing and signing digital certificates to cloud service providers in the Intercloud system).

B. Intercloud Exchange

Operating under Intercloud Root, Intercloud Exchange mediates the communications between its associated clouds and Intercloud Root. One of the major functions of Intercloud Exchange is to facilitate the provisioning of resources for clouds. With authority from Intercloud Root, Intercloud Exchange hosts a DNS server with a sub-domain of Intercloud Root to manage resource information for different clouds. It is proposed that the resource information for the clouds can be stored in short for text records (TXT records) in the DNS server. TXT record is a type of resource record like Address Record, so it can be retrieved by simple DNS queries. Some basic and static resource information, such as services provided, vendor, and geographic location of a cloud, can be stored in multiple TXT records. Examples of the TXT records are given below. As a result, a cloud can communicate with its corresponding Intercloud Exchange to discover potential clouds that are capable of providing the required resources.

```
c1.e1.r1.iccp.us. IN TXT "Intercloud_Service=ObjectStorage:Minio;VM:HyperV"
c1.e1.r1.iccp.us. IN TXT "Intercloud_Vendor=ABCCompany"
c1.e1.r1.iccp.us. IN TXT "Intercloud_Geolocation=CountryCode:HK"
```

C. Cloud

Cloud (or cloud service provider) provides a pool of computing resources such as virtual machines and storage resources for cloud users over the Internet. In practice, the resources provided by a cloud service provider are limited so different cloud service providers can form an alliance. A cloud service provider can discover computing resources and communicate with other cloud service providers through the Intercloud system.

D. Intercloud Gateway

Intercloud Gateways are responsible for communications between clouds. Each cloud entity (Intercloud Root, Intercloud Exchange, and Cloud) is associated with an Intercloud Gateway to communicate with other cloud entities using the same protocol. An Intercloud Communications Protocol (ICCP) was proposed to support communications between heterogeneous cloud providers [14]. In the next section, an extension of ICCP to support cloud resource discovery is presented.

III. Intercloud Resource Discovery Protocol

An XML-based Intercloud protocol ICCP was proposed for facilitating Intercloud communications [14]. The protocol design is based on a simple request and response pair, and is highly extensible by adding new commands. An extension of ICCP that supports resource discovery in the Intercloud architecture is proposed and an “InquireResource” service is added. A pair of “InquireForResource” request messages and “ReplyForResource” response messages is shown below. A request is sent to a cloud, containing the requirements of the resources from the requesting cloud, including a specification for a virtual machine. After receiving the request, the receiver cloud checks whether it can meet the resource requirements. A positive response is sent if the cloud can provide the resources. The details of the supplied resources are included in the response message. A negative response is sent if the cloud cannot provide the resources. The “InquireForResource” request can also be sent to Intercloud Exchange and Intercloud Root, and a list of recommended Cloud domains is replied.


```
<Request Version="1.0" ID="1487744976553">
<GeneralInformation From="c1.e1.r1.iccp.us" To="c2.e1.r1.iccp.us"
Date="2017-02-21" Time="22:29:36">
</GeneralInformation>
<RequestInformation Service="InquireResource" Command="InquireForResource">
<Service>VM</Service>.
<Memory>2GB</Memory>
<CPU>2</CPU>
<Disk>100GB</Disk>
<Geolocation>HK</Geolocation>
</RequestInformation>
<AdditionalInformation>
<Signature>dfvxTdx/zJLqkNzQiuDdM3PJ40xsC7WXwTxnE46yRM6DQra</Signature>
<SignatureAlgorithm>SHA1/RSA</SignatureAlgorithm>
</AdditionalInformation>
</Request>
```



```
<Response Version="1.0" ID="1487744976553">
<GeneralInformation From="c2.e1.r1.iccp.us" To="c1.e1.r1.iccp.us"
Date="2017-02-21" Time="22:29:38">
</GeneralInformation>
<ResponseInformation Service="InquireResource" Command="ReplyForResource">
<Vendor>XYZLimited</Vendor>
<Service>VM</Service>
<ServiceProvider>VMware</ServiceProvider>
<Memory>2GB</Memory>
<CPU>2</CPU>
<Disk>100GB</Disk>
<Geolocation>HK</Geolocation>
</ResponseInformation>
<AdditionalInformation>
<Signature>aD7g/HvcdEhXHYMOtamQfazQQJcVGpfE33/x+9cd/92V5o+</Signature>
<SignatureAlgorithm>SHA1/RSA</SignatureAlgorithm>
</AdditionalInformation>
</Response>
```


Fig. 2 illustrates a resource discovery process initiated by a cloud. When a cloud cannot satisfy users’ requirements, it may initiate an “InquireForResource” request to its Intercloud Exchange for resources. The discovery process consists of two steps. The first step is to obtain a recommended Cloud list from an Intercloud Exchange. The resource requirements are divided into two categories, and specified in the request. The static and public resource information, such as the type of services required, vendor, and the geographic location of the supplier cloud, is considered, however, the relatively dynamic and private resource information, such as memory and CPUs of a virtual machine, is ignored by the Intercloud Exchange. Intercloud Exchange applies the preferences and constraints to its affiliated clouds, and selects the clouds that meet the requirements according to the DNS TXT records. If the Intercloud Exchange cannot find any suitable cloud, it redirects the request to Intercloud Root. Otherwise, the domain names of clouds are returned. When Intercloud Root receives the request, it applies the preferences and constraints to all the clouds in the Intercloud system. It returns a list of domain names of clouds, unless all clouds in the system do not meet the requirements. After receiving the Cloud list, the second step of the discovery process is to send an “InquireForResource” request to the clouds that are recommended

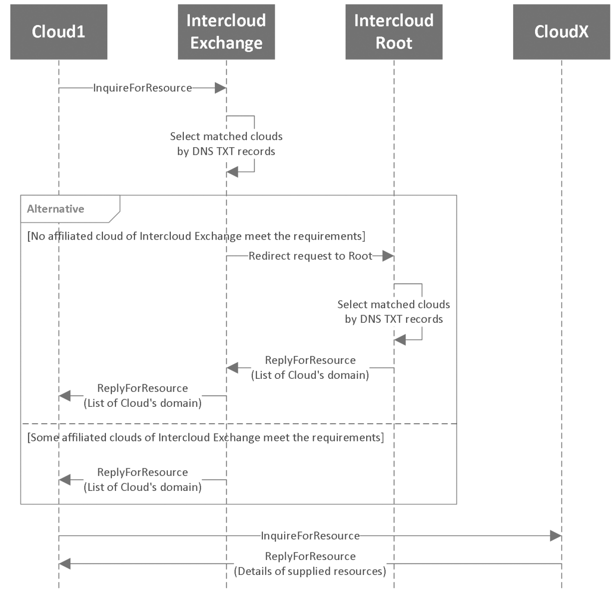

Fig. 2. Sequence Diagram of Resource Discovery Process

by the Intercloud Exchange. The detailed requirements of the resources, such as the amount of memory, processing units, and disk space of a virtual machine are all considered by the supplier cloud. If the supplier cloud can provide the services, it replies with a positive response to the requesting cloud.

There is a certain probability that a cloud will not be able to provide the required services or resources, so the requesting cloud may need to send its request repeatedly to multiple cloud providers in the second step of the discovery process in order to obtain resources. Two methods, sequential search and broadcast search, are proposed to solve this problem. By using a sequential search, the requesting cloud sends the request to the clouds on the list iteratively. It waits for a response from one cloud before sending a request to another cloud. The procedure repeats until a certain cloud returns a positive response, or all clouds on the list have been requested once. The number of requests sent to clouds in the system is minimized, but it may result in a long wait time if many iterations are required to successfully obtain resources. By using a broadcast search, a requesting cloud sends the request to all clouds on the list simultaneously, without waiting for any response. Therefore, it is possible to minimize the waiting time; however, the number of requests generated is increased. In the next section, analytic models based on queuing theory are presented. The constraints and response time of the two proposed methods are compared and investigated. The response time of obtaining a recommended Cloud list from Intercloud Exchange and Intercloud Root is not included in the model, since it remains relatively constant regardless of the use of a sequential search or a broadcast search.

IV. Queuing Model

Suppose there are N clouds, denoted as $C_1, C_2, \ldots, C_N$, in the Intercloud architecture. Every cloud is modeled as an independent queue that receives and processes resource

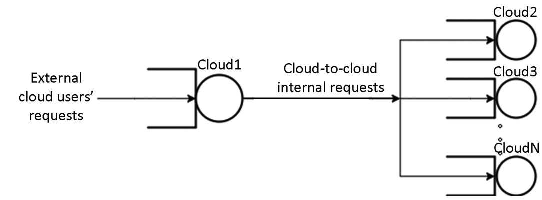

Fig. 3. External and Internal Requests

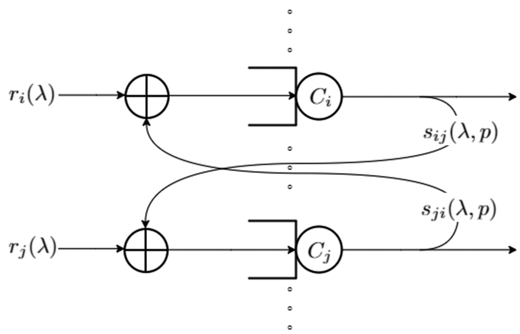

Fig. 4. Generalized Queuing Model of the system

discovery requests. It is assumed that a cloud cannot provide the required resources of a request with a probability p. Unlike the common cloud computing model, clouds not only provide services to cloud users, but also consume resources from other clouds. Fig. 3 illustrates the two types of requests, which are initiated by users and clouds. The requests from ordinary cloud users are defined as external requests, since they are generated outside the Intercloud system. Consecutive arriving users' requests may be sent from different users, so the inter-arrival time is an exponential random variable. Therefore, we assume that the external requests follow a Poisson Process with arrival rate λ and the external arrival rate of C_i denoted as $r_i(\lambda)$. When a cloud receives a user's request that the corresponding services cannot provide, it generates a request, which it sends to another cloud. The requests generated by clouds are defined as internal requests, since they are generated by entities in the Intercloud system. The generation of internal requests depends on the result of the external request, so the internal arrival rate depends on λ and p and the internal arrival rate from C_i to C_j is denoted as $s_{ij}(\lambda, p)$. Additionally, we assume the service time of a cloud to process a request follows an exponential distribution with service rate μ so the internal requests also follow a Poisson distribution. Owing to the characteristics of Poisson distribution, the total arrival rate of a cloud can be calculated by equation 1. Hence, every cloud can be modeled as an M/M/1 queue and the generalized queuing model is shown in Fig. 4.

$$n_i = r_i + \sum_{j=1, j \neq i}^{N} s_{ji} \tag{1}$$

A. Sequential Search

The basic principle of sequential search is to search for resources iteratively with a given sequence. The searching sequence for clouds is not the same, in order to avoid system overwhelming by the immense number of requests received by a cloud provider. For example, C_1 searches from C_2 to C_N. In general, C_i searches from C_{i+1} to C_N and then from C_1 to C_{i-1}. With this searching sequence, the internal arrival rate from C_i to C_j is shown in equation 2.

$$s_{ij} = \begin{bmatrix} 0 & p\lambda & p^2\lambda & \cdots & p^{N-1}\lambda \\ p^{N-1}\lambda & 0 & p\lambda & \cdots & p^{N-2}\lambda \\ p^{N-2}\lambda & p^{N-1}\lambda & 0 & \cdots & p^{N-3}\lambda \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p\lambda & p^2\lambda & p^3\lambda & \cdots & 0 \end{bmatrix} \tag{2}$$

By the formula of M/M/1 queue, the expected response time of a request τ is given by

$$\tau = \frac{1-p}{(1-p)\mu - (1-p^N)\lambda} \tag{3}$$

The response time of the entire process of resources discovery that we want to measure is calculated from the request initiated by a user to the corresponding response received by the user. The response could be positive, with the ability to obtain the required resources, or negative, if the required resources cannot be obtained. Hence, the expected response time T of the entire process is given as follows.

$$T = \frac{1-p^N}{(1-p)\mu - (1-p^N)\lambda} \tag{4}$$

B. Broadcast Search

The basic principle of broadcast search is to search for resources by sending the requests to all target clouds simultaneously. The main difference between broadcast and sequential is that the order or sequence of searching clouds is ignored. Therefore, the internal arrival rate from a cloud C_i to all other clouds C_j, $1 \leq j \leq N$ and $i \neq j$, are the same. The internal arrival rate from C_i to C_j is shown in equation 5.

$$s_{ij} = \begin{bmatrix} 0 & p\lambda & p\lambda & \cdots & p\lambda \\ p\lambda & 0 & p\lambda & \cdots & p\lambda \\ p\lambda & p\lambda & 0 & \cdots & p\lambda \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p\lambda & p\lambda & p\lambda & \cdots & 0 \end{bmatrix} \tag{5}$$

By the formula of M/M/1 queue, the expected response time of a request τ is given by

$$\tau = \frac{1}{\mu - (1 + (N-1)p)\lambda} \tag{6}$$

In order to have a fair comparison between the methods of sequential and broadcast search, the response time of the entire process of resources discovery is calculated from the request initiated by a user to the corresponding response received by the user. By adopting a broadcast search, the first positive response received among clouds is replied to the users, so the expected response time T of the discovery process cannot be calculated by the summation of the mean delay of each cloud directly. There are two cases that we must consider. First, a user's request is satisfied by a subset of clouds in the system. In this situation, the minimum of the mean delay of the clouds subset should be considered. Second, a user's request cannot be satisfied by any cloud in the system. In this case, the maximum of the mean delay of all clouds is considered. From the result in [15], the mean delay of an M/M/1 queue is an exponential random variable. The expectation of the minimum of $N-1$ exponential random variable is given as $\frac{1}{(\mu-(1+(N-1)p)\lambda))(N-1)}$. The expectation of the maximum of $N-1$ exponential random variable is given as $\sum_{k=1}^{N-1}\frac{1}{(\mu-(1+(N-1)p)\lambda))k}$. Hence, the expected response time T of the whole process is given as follows.

$$T=\sum_{k=1}^{N-1}(\binom{N-1}{k}(1-p)^k+p^k)\frac{p^{N-k}}{(\mu-(1+(N-1)p)\lambda))k}+\frac{1}{\mu-(1+(N-1)p)\lambda} \quad (7)$$

V. Results and Discussion

In this section, the analytical and simulation results of the two proposed methods of discovering resources are presented and analyzed. The simulation results are used to validate the equations presented in Section IV. An event-based simulation program is used to emulate an Intercloud system where there are numerous clouds forming a fully-meshed structure. In the simulation, parameters are varied to demonstrate the performance of using different discovery methods. The number of clouds N and the external arrival rate λ are varied, and the expectation of the entire response time T of the discovery process is measured. In each simulation run, there are at least 10,000 external requests that have been processed by the system. In order to make a fair comparison between two proposed approaches, a set of external requests is generated at the beginning of each simulation, and the two approaches are adopted with the same set of requests.

Fig. 5 demonstrates the comparison of the total response time between sequential and broadcast search by varying the number of clouds in the system. When there is a small number of clouds in the system, a broadcast search performs better than a sequential search. As the number of clouds in the system increases, the response time when using broadcast searches increases more significantly than when using a sequential search. Therefore, a sequential search performs better than a broadcast search when the number of clouds in the system increases. Fig. 6 depicts the comparison of the total response time between the two approaches by varying the external arrival rate with a fixed number of clouds. It is also obvious that the response time of a broadcast search increases more rapidly compared to a sequential search when the external arrival rate increases. Therefore, a sequential search performs better than a broadcast search when the external arrival rate increases. In addition, Figs. 5 and 6 show that the analytical results and simulation results are matched with each other. Therefore, the equations 4 and 7 that we derived are valid.

As a further investigation, the total number of external and internal requests processed by the system in the above simulation is analyzed. In Figs. 7 and 8, the total number of external requests, internal requests generated using sequential

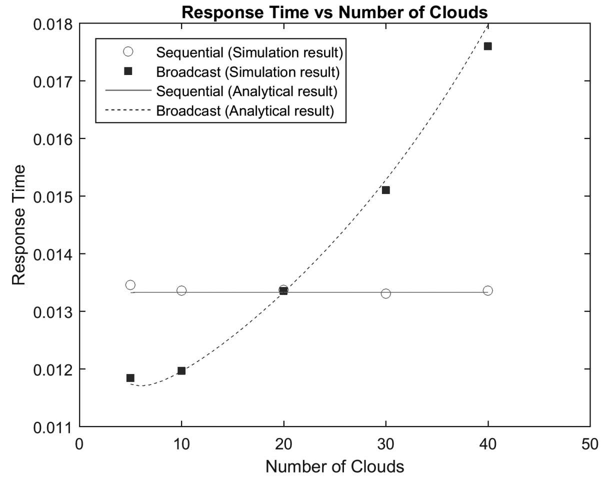

Fig. 5. Response time vs number of clouds

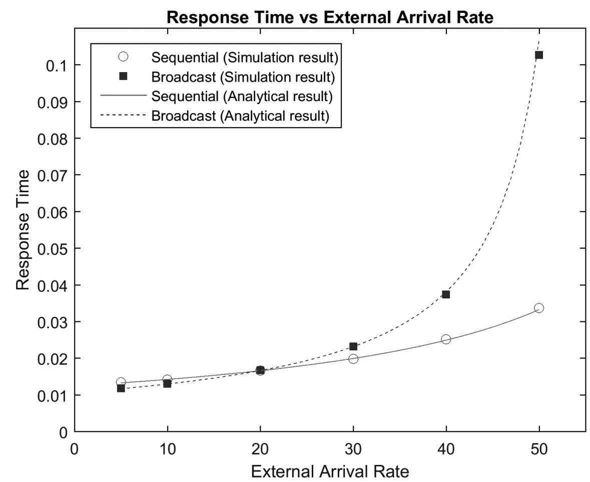

Fig. 6. Response time vs external arrival rate for a fixed number of clouds

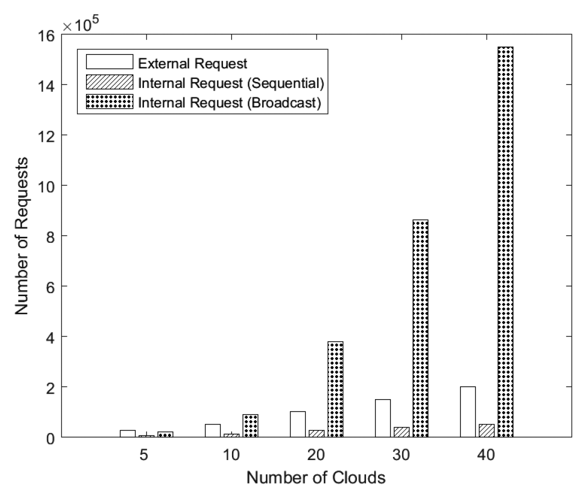

Fig. 7. Number of requests for different number of clouds

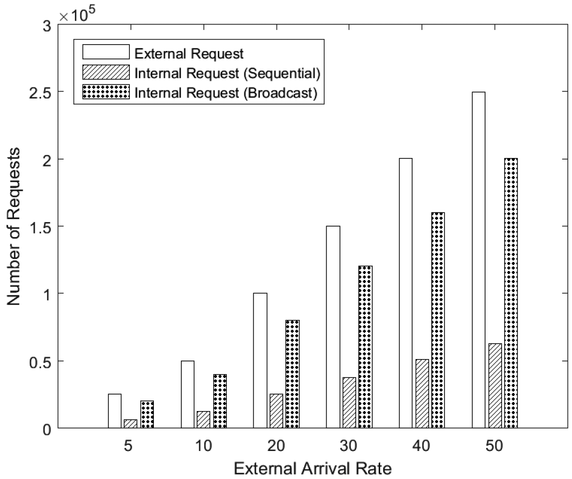

Fig. 8. Number of requests for different external arrival rate

search, and internal requests generated using broadcast search are shown. The results shown in Figs. 5 and 7 are obtained from the same simulation tests. In Fig. 7, the number of external requests increases linearly as the number of clouds increases. This result is trivial, because the total number of cloud users served by the Intercloud system increases. In addition, the number of internal requests generated by using sequential search is always less than the number of external requests; however, the internal requests generated by using a broadcast search are far greater than when using a sequential search. When the number of clouds increases to 40, the number of internal requests is already seven times more than the number of external requests. The results shown in Figs. 6 and 8 are obtained from the same simulation tests. In Fig. 8, the number of external requests increases linearly as the number of clouds increases. This result is trivial, due to the increase in the external arrival rate. In addition, the number of internal requests when using two methods also increased linearly, but the number of internal requests in a sequential search is always less than in a broadcast search.

To summarize the results in Figs. 7 and 8, many more internal requests are generated by using a broadcast search, compared to a sequential search; because of this, a broadcast search induces a heavier load to the system, with a higher internal arrival rate. The utilization ρ of a cloud is calculated by the ratio of total arrival rate, which is the sum of external arrival rate and internal arrival rate, to its services rate. In queuing theory, if ρ increases, the queuing delay increases. Therefore, from Figs. 5 and 7, and Figs. 6 and 8, a positive correlation can be observed between the number of internal requests and the response time. From the above result, we may conclude that a sequential search is relatively scalable compared to a broadcast search when the number of clouds in the system and the external arrival rate increase.

VI. Conclusion

To conclude, resource discovery is one of the key challenges in utilizing resources in an Intercloud environment. To address this problem, an extension of ICCP with DNS support is proposed to facilitate resource discovery in the Intercloud system. In addition, two methods, sequential search and broadcast search, are proposed and analyzed based on queuing theory. The analytical results are validated by simulation. According to the simulation results, sequential search performs better, with a relatively short expected response time. Adopting a sequential search is more scalable than a broadcast search, owing to the minimized number of internal requests generated.

VII. Acknowledgements

This project is supported by the Innovation and Technology Fund of the Government of the Hong Kong Special Administrative Region and by CITIC Telecom International CPC Limited.

References

[1] P. Mell, T. Grance *et al.*, "The NIST definition of cloud computing," 2011.

[2] K. Nagin, D. Hadas, Z. Dubitzky, A. Glikson, I. Loy, B. Rochwerger, and L. Schour, "Inter-cloud mobility of virtual machines," in *Proceedings of the 4th Annual International Conference on Systems and Storage*. ACM, 2011.

[3] D. Petcu, G. Macariu, S. Panica, and C. Crăciun, "Portable Cloud applications-From theory to practice," *Future Generation Computer Systems*, vol. 29, no. 6, pp. 1417–1430, 2013.

[4] A. N. Toosi, R. N. Calheiros, and R. Buyya, "Interconnected cloud computing environments: Challenges, taxonomy, and survey," *ACM Computing Surveys*, vol. 47, no. 1, pp. 1–47, 2014.

[5] R. Buyya, R. Ranjan, and R. N. Calheiros, "Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services," in *International Conference on Algorithms and Architectures for Parallel Processing*. Springer, 2010, pp. 13–31.

[6] R. Nikbazm and M. Ahmadi, "Agent-based resource discovery in cloud computing using bloom filters," in *Computer and Knowledge Engineering (ICCKE), 2014 4th International eConference on*. IEEE, 2014, pp. 352–357.

[7] S. Abdelwahab, B. Hamdaoui, M. Guizani, and T. Znati, "Cloud of things for sensing as a service: sensing resource discovery and virtualization," in *Global Communications Conference (GLOBECOM), 2015 IEEE*. IEEE, 2015, pp. 1–7.

[8] A. Celesti, F. Tusa, M. Villari, and A. Puliafito, "How to enhance cloud architectures to enable cross-federation," in *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*. IEEE, 2010, pp. 337–345.

[9] G. Le, K. Xu, and J. Song, "Gossip-based hybrid multi-attribute overlay for resource discovery in federated clouds," in *e-Business Engineering (ICEBE), 2012 IEEE Ninth International Conference on*. IEEE, 2012, pp. 279–284.

[10] P. Khethavath, J. Thomas, E. Chan-Tin, and H. Liu, "Introducing a distributed cloud architecture with efficient resource discovery and optimal resource allocation," in *Services (SERVICES), 203 IEEE Ninth World Congress on*. IEEE, 2013, pp. 386–392.

[11] R. Rahman, N. Nahar, R. Islam, and K. Sakib, "A peer to peer resource discovery scheme for provisioning in cloud using multi attribute range query," in *Informatics, Electronics & Vision (ICIEV), 2014 International Conference on*. IEEE, 2014, pp. 1–5.

[12] P. Mockapetris, "DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION," Requests for Comments, RFC Editor, RFC 1035, November 1987. [Online]. Available: https://rfc-editor.org/rfc/rfc1035.txt

[13] D. Bernstein and D. Vij, "P2302 - Standard for Intercloud Interoperability and Federation (SIIF)." [Online]. Available: http://standards.ieee.org/develop/project/2302.html [Accessed: 14-Jun-2016]

[14] C. K. Law, W. Xie, Z. Xu, Y. Dou, C. T. Yu, H. C. B. Chan, and D. W. K. Kwong, "System and Protocols for Secure Intercloud Communications," in *2016 11th International Conference for Internet Technology and Secured Transactions (ICITST)*. IEEE, 2016, pp. 399–404.

[15] P. G. Harrison, "Response time distributions in queueing network models," in *Performance Evaluation of Computer and Communication Systems*. Springer, 1993, pp. 147–164.