A Novel Approach for Resource Discovery using Random Projection on Cloud Computing Environments

M.N.Faruk [1], Dr.D.Sivakumar [2]
Research Scholar, Bharath University, Chennai.[1]
Dean & Professor, Dept of ECE, Arunai College of Engineering, Tiruvannmalai [2]
justdoit.fff@gmail.com **1**
dgsivakumar@sify.com [2]

***Abstract* - Cloud computing offers different type's utilities to the IT industries. Generally the resources are scattered throughout the clouds. It has to enable the ability to find different resources are available at clouds, This again an important criteria of distributed systems. This paper investigates the problem of locating resources which is multi variant in nature. It also used to locate the relevant dimensions of resources which is avail at the same cloud. It is also addresses the random projection on each cloud and discover the possible resources at each iteration, the outcome of each iteration updated on collision matrix. All the discovered elements are updated at the Management fabric. This paper also describes the feasibility on discovering different types of resources available each cloud.**

Keyword - Resource discovery, Multi-variant, Resource subsets, Service level agreement (SLA), Virtual Pool.

I. INTRODUCTION

In a modern computing world virtualization and cloud computing provides fundamental phenomena of decoupling applications and services from underlying hardware infrastructure. It provides feasibility on multi-variant resources like CPU memory and network bandwidth etc.,[1] All these resources are promoted by demand basis and adopted in responds to changes in system work loads. In cloud computing

Resources allocation can be sealed on both the extremes and provides elasticity by reflecting load intensity and resource demands of running applications.[2] More over virtualization provides the platform on resource utilization by reducing number of physical computing on cloud environment. Cloud environment has sustainable by data centers which incorporated by Virtual Machine (VM) on same physical hardware. VM improves energy efficiency and cost savings for both service provided and infrastructure provided. This dynamic environment has challenging platform on QOS by means of availability and performance.[1,5] During resource utilization 1.how much resource are allotted on new service request by virtualization at the same time it should guarantees SLA on new service and existing service. In order to discover the resource on cloud environment .The discovery mechanisms provides manual progression on cloud resources. It's one of the key challenges of cloud environment discovery mechanisms.[1] In most of the cases the search results provides invalid information on virtualized resource that has false information. It is quite impossible to work on multiple false information which is derived from the search engine. The search engines and manual process of searching virtualized resources on cloud environment requires an supervised learning mechanism.

This paper focuses on the case of discovering an unknown resources (Motif's) which is multi variant in nature. These discovery mechanism has multi dimensional view on multi variant resource on cloud. However this technique extracted from multi sensor system.[4,8] In this approach we are interested in addressing on broader view on multi variant resource which we represented as sub dimensional resource discovery. Each resource has multiple criteria's to fulfill such as similarity basis, equalant basis and cost basis. In this paper we presented an algorithm to discover the resource which is previously unknown and multi variant time series.[6] It also guarantees the SLA violation Our algorithm naturally handles more restrictive problems which we call all dimensional resource discovery on clouds it gives an approach on Discovering Multi variant resources.

II. RELATED WORKS

One of the first work in which event based resource discovery has been done by wei yan, Soglin Hu et al [1]. In this work authors have proposed a framework to enable decentralized discovery of both static and dynamic service and resource attributes to enable the continues monitoring of resource updates.

In the second work in which R.Raman et.al [6] established a centralized architecture. In this paper centralized matching server incorporated with no of clients the discovery requests are sent to the matching server. According to the maximum matching utility the resource discovery has been established. However it could feasible for small scale networks. For large scale networks, it requires central administrators which may cause central platform failure.

In other work in which Globus MDS is defines the resource discovery by hierarchical architecture.[7] Index services have become a vital part in this architecture. It is a grid model in which multiple information sources are grouped and transferred into index servers via registered protocol. The Index servers are incorporated with query servers. The registered protocol will compare the service requests with query servers and derives the valid information with the help of information servers. However its limits with scalability which is difficult with administrating different types of servers.

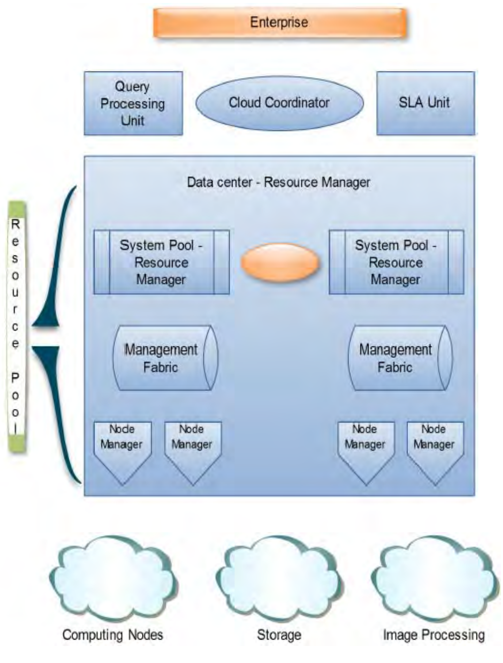
In a distributed environment the federated UDDI consists of multiple repositories that are synchronized periodically.[8] It gives a better solution for service discovery. However it is quite expensive to replicate frequently updated information. Publish/subscribe has been leveraged for service discovery. In this approach the publish/Subscribe servers received a translated message in the form of service attributes and domain requirements.

The combination of multiple nodes may face big complexity with two models. But it quite suit for small scale individual nodes it is implemented in both the forms static and dynamic.

III. ARCHITECTURAL FRAMEWORK

Cloud aims to provide datacenters for upcoming technology. The framework includes virtual machines and services in the form of (hardware resources and Software resources) by which user can deploy their application anywhere.[5] Each cloud client may undergo with individual SLA's (Service Level Agreement) for their task. For public clouds the resources are collocated by demand basis. The frame work may include with three layers.

- Enterprise layer
- Resource pool layer
- VM pool



III.A. ENTERPRISE LAYER

The Enterprise layer is having three major components.

Query processing unit: Each cloud client may come up with service requests. The service request must be properly analyzed and forwarded into next level of processing. The cloud client must undergone with query processing unit to confirm the service request to be valid and must incorporated with SLA unit.

The SLA unit: Service Level Agreement (SLA) each cloud client must digitally signed with SLA which is derived from each cloud server. The cloud server is responsible for providing different types of resources to each individual cloud client. The SLA defines the level of usage and maximum usage frequency with each individual resource.

Cloud coordinator: The cloud coordinator unit does an authentication service. the cloud client should verify their service request by query processing unit. Once the query processing unit validated the service request then it will forwarded to SLA unit. The cloud should accept the terms and conditions followed by SLA unit. Finally the cloud coordinator is forward the request to Resource pool manager for further level of arrangements.

III.B. RESOURCE POOL LAYER

System Pool Resource manager: The management service segregate the available resources present in the different cloud environment. Each cloud environment includes different types of resources. The overall resource management is done at this level.

Management Fabric: The outcome collision matrix updated on management fabric unit which contains radius R only the relevant dimensions. All the resource availability is being monitored here and updated frequently with resource pool manager.

IV. RESOURCE DISCOVERY CRITERIA

In contrast to the discussed studies we propose the multi variant resource available at different clouds. It gives a brief review of existing algorithmic framework and provides some recent enhancements to discover the multi variant resources. This paper simplifies an algorithm that adopts at cloud environment and discovers the available resources and its related subsets.[4] Once the available resource and are identified, using random projections. The related subsets are identified in a linear time sequences. The algorithm determines the relevant dimensions for each resource available in a cloud.

IV. A. RANDOM PROJECTIONS

The Random projection is a mechanism that is to locate the resource available in a different clouds. However it derives approximately equal subsets in a linear time. After extracting subsets that named and converted in a single group.[4] The algorithm provides through several iterations of random projection. Each iteration selects subsets of its positions and projects unique resource by eliminating the remaining resources and its positions.

Series (S) of Resources on clouds (Private, Public & Hybrid).Computing Capacity (C) The Active resources are already connected with cloud (w).Max no of random projections and iterations (Maxrp).SLA violation Limit (threshrel).Time taken and utilization Measure (D(s)). Once the resources has discovered it should updated on collision matrix. After implementation of collision matrix after each iteration the collision matrix details updated on management fabric layer. The total time complexity of this resource discovery algorithm is calculated by No of resources in the cloud (T). No of iteration for the discovery (I). Then the collision matrix will be $C = \sum_{i=1}^{n} C_i$

If T will increase then the total complexity will increases. In order to detect similar resource that available on different clouds the collision matrix will be maintained. If there are T subsets, then the collision matrix size is T x T and stores the no of iterations. The matrix is updated after each iteration by having the projected resources and then incrementing the matrix entry by each equivalent subsets.

ALGORITHM: RESOURCE DISCOVERY

C – Computing Capacity

c – Location and availability of different resource that lead to sparse collision matrix.

R- Discovered resource collected from no of clouds connected to the VM pool Si

Each resource in cloud with different computing capacity are arranged in a series.

S: $S_i = \{S_i \ldots S_{i+w-1}\}$: $1 \leq i \leq |S|-w+1$

Compute $\hat{P}(D) \simeq p(D(S_i, d, S_j, d)$ an estimation of the distribution over the distance between all non trivial resources on each cloud.

Search for resource which having different computing capacity discovered resource lead to sparse collision matrix.

Discover the resource which connected to the cloud (m), computing capacity for each iteration & with different clouds(s).

Build the collision matrix using random projection over different clouds.

No of iteration = min ((mc), max_{rp}).

Enumerate Discovered resource on the collision matrix.

(a). Find the best collision matrix entry (x^1, x^2)

(b). Find the largest resource entry which has maximum capacity in the collision matrix and extract the set of all collision with this value,

$X=\{(x_1^1,x_1^2), (x_2^1,x_2^2)\ldots(x^1_{|x|}, x^2_{|x|})\}$

(c). Compute the Distance with relevant resource in the clouds xj1, xj2 in each collision, 1<j<|x| and dimension.

$$d : dist_{j,d} = D(S_{xj^1}, d, S_{xj^2}, d)$$

(d). Determine which resource are not relevant to the task assignment

$$rel(d) = I(\int_{-\infty}^{dist_{j,d}} \hat{P}(D)) < thres_{rel})$$

(e). Select the collision matrix with smallest avg distance actively connected resources per iteration

$$(x_j^1.x_j^2) : j = \arg_j \min(\frac{\sum d^{dist} j, d^{rel(d)}}{\sum d^{rel(d)}})$$

(f). Estimate the neighborhood radius R, only the relevant dimension

(g). Locate all other occurrences of the similar cloud

$$\min(S_{x^1}, S_i), D(S_{x^2}, S_i) \leq R$$

(h). Disconnect all the resources that would constitute non trivial matches with the cloud.

The total time complexity of the random projection algorithm is linear with number of resources in a single cloud (T). The number of iteration (I) and the location and availability of different resources (c) and the number of collision

The complexity is dominated by the collision since Ci grows quadratically with the number of each projected resources which can rise as high as T in worst case.

$$C = \sum_{i=1}^{n} C_i$$

$$C_i = \sum_{h \in H} (\frac{N_k}{2}) = \sum_{h \in H} \frac{1}{2} N_k (N_k - 1)$$

Specifically where H is the set of all projected resources N_k equals to the number of resources that project to $h \in E$.

V. Experimental setup

This Framework could be evaluated by running experiments using predefined located resources as well as non synthetic resources captured by on body inertial sensors. Our experiments demonstrate the efficacy of the algorithm as well as scaling properties if the location of different resources and number of dimension increases. This also investigates the effect of different distances metrics and provide a comparison with other multivariant discovery algorithm.

VI. Conclusion

We have described several generalizations of the multivariant resource discovery properties. The key feature is that allows linear time discovery of sub dimensional motifs. We apply random projection independently to each dimension only if the size of the collision matrix is monitored and the relevant parameters are dynamically monitored to limit the density of the matrix. We do researches on other complexities that will occur while applying on real time.

VII. REFERENCES

[1] Wei Yan, Songlin Hu, Vinod muthusamy, Hans-Arno Jacobson, Li Zha: Efficient Even-based Resource Discovery. Proceedings of the Third ACM International Conference on Distributed Event-Based Systems.

[2] Edo Zural and Noam Nisan: An Efficient Approximate Allocation Algorithm for combinatorial Auctions. ACM 2001 Article.

[3] Isah Abdul and Safwana Haque : Resource management in Grid Computing: A Review in Green Journal of science engineering and technology research. Vol.2(1).pp 024-031, April 2012.

[4] T. Bailey and C. Elkan. Fitting a mixture model by expec-tation maximization to discover motifs in biopolymers. In Second International Conference on Intelligent Systems for Molecular Biology, pages 28–36. AAAI Press, 1994.

[5] J. Buhler and M. Tompa. Finding motifs using random projections. In International Conference on Computational Biology, pages 69–76, 2001.

[6] R. Raman, M. Livny, and M. Solomon. Matchmaking: distributed resource management for high throughput computing. In IEEE HPDC, 1998.

[7] Andrew J. Younge, Gregor Von Laszewski et al., : Efficient resource management for cloud computing environments. IEEE.

[8] Anton Beloglazov, Jemal abawajy, Rajkumar Buyya: Energy Aware resource allocation heuristics for efficient management of data centers for cloud computing. In Future Generation computer systems 28(2012), 755-768.

[9] Vijay Kumar, James Metzler et al., : Object Discovery, Identification and Association : Proceeding in SPIE vol.8050,805016.

[10] Zenon Chaczko, Venkatesh Mahadevan et al., Availability and Load Balancing in Cloud Computing, in 2011 International Conference on computer and software Modeling IPCSIT vol. 14 (2011). Singapore.
[11] B. Alexander, "Web 2.0: A New Wave of Innovation for Teaching and Learning?" Learning, vol. 41, no. 2, pp. 32–44, 2006.
[12] R. Buyya, C. Yeo, and S. Venugopal, "Market-oriented cloud computing:Vision, hype, and reality for delivering it services as computing utilities,"in Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC-08, IEEE CS Press, Los Alamitos, CA, USA), 2008.
[13] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud Computing and Grid Computing 360-Degree Compared," in Grid Computing Environments Workshop, 2008. GCE'08, 2008, pp. 1–10.
[14] S. Becker, H. Koziolek, and R. Reussner. The Palladio component model for model-driven performance prediction. Journal of Systems and Software, 2009.
[15] M. N. Bennani and D. Menasc´e. Resource Allocation for Autonomic Data Centers using Analytic Performance Models. In ICAC, 2005.
[16] G. Box and G. Jenkins. Time series analysis: forecasting and control. Prentice Hall PTR Upper Saddle River, NJ, USA, 1994.
[17] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In ACM SIGCOMM, 2001.
[18] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph. Tapestry: An infrastructure for fault-tolerant wide-area location and and routing. Technical report, Berkeley, CA, USA, 2001.
[19] Buyya, R., Yeo, C.S. and Venugopal, S. 2008. Market-oriented Cloud computing: Vision, hype, and reality for delivering it services as computing utilities. Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications, Los Alamitos, CA, USA.
[20] New Datacenter Locations. 2008. http://royal.pingdom.com/2008/04/11/map-of-all-google-data-center-locations/
[21] Bianchini, R., and Rajamony, R. 2004, Power and energy management for server systems, Computer, 37 (11) 68-74.