Dynamic Task Migration Mechanisms in Cloud Environment: Literature Review and Future Trends

Ankit Tomar
Uttarakhand Technical University, Dehradun

J. C. Patni
University of Petroleum and Energy Studies, Dehradun

Amit Dixit
Quantum University, Roorkee

Pramod Kumar
Krishna Engineering College, Ghaziabad

ABSTRACT

IT sector is adopting novel cloud computing to converge the business and technology platforms in terms of ease of services which allow users to access the resources in pay as go fashion anytime and anywhere. To accomplish this goal several challenges have to face in which balancing the load among the nodes is one of them. Despite the significance of load balancing algorithms there is no organized literature which could cover or analyze the dynamic migration techniques, their scope, limitations and challenges, so this article focus on those dynamic load balancing techniques in which lot of work has done to reduce the migration time of tasks form one VM to other. Task migration is an important load balancing metric in cloud computing by relocating active virtual machines (VMs) from one candidate node to another. To achieve better system performance load must be distribute evenly across the servers, for this under loading and overloading of VM should be avoid by migrating the extra tasks in shortest period. In this literature we have given the detailed overview of qualitative and quantitative analysis of existing task migration schemes, also merits, demerits and important challenges are addressed so that more resourceful and scalable migration algorithms could be develop in near future.

Keywords

Load Balancing, Task Migration, Data Center, VM, Cloud Computing, Virtualization

1. INTRODUCTION

The field of Cloud computing is in the area of computer engineering that offers the access of pools of resources such as data, network and servers etc. Every industry is enforcing for the use of cloud based services that works on ‘Pay as Go’ scheme. Cloud services provide user to access the remote servers with enhanced availability, scalability and tolerance of fault. It basically a service which is collection of servers and nodes subjected to the third party under which the data is stored [1]. Job Migration is an important load balancing metric in which live tasks are relocated from one physical machine to another of probably dissimilar design. Hence it is required to save the current state of the task and convalescing it on the destination machine [35]. Task scheduling is used to distribute the load evenly among the pool of resources to enhance the performance, reduced the cost by bringing the processes closer together. This paper highlights major migration techniques, classification, challenges and then future scope. Amount of data and number of user requests are increasing every day, which require more computing and processing of the servers. It distributes the computing tasks to the resource pool made from a large number of computers. In section 2 we have given detailed overview of task migration and its categories, section 3 describes the comprehensive literature review of existing dynamic load balancing techniques for reducing the migration overhead, from where the articles we have taken on the basis of certain criteria and what is the research design explained in section 4, in section 5 we focused on loopholes, disadvantages and shortcomings of existing migration techniques and also explained about future guidelines for reducing the migration time as well as power consumption.

1.1 Cloud Services 1

We have three basic three layers of Cloud computing. First one is (SAAS) software as a service, that facilitates users to access the pooled resources remotely forms the cloud. The services of this layer are provided by the service providers without knowing about its management services. A software service makes the services available to the users and applications are hosted by third party. For examples- Google Apps. Second one is (IAAS) infrastructure as a service defines the architectural design like networking, access of resources, power and bandwidth [2]. In this model the third party provider hosts the virtualized resources and storage as well as makes them available to the users. For examples Web Services, Third layer (PAAS) Platform as a Service is concerned with application of clouds line hardware resources and operating systems through which user can take advantage of pooled resources on working machines. In this model third party provider host applications and its infrastructure and make the service available to the users.

Cloud computing is an emerging area and widely used today as increasing demand of data access. It is a self-service oriented utility to accessing the virtual pool of abstract information. Researchers developed various high performance static and dynamic load balancing algorithms that distribute the load to all candidate nodes uniformly with minimal response. Load balancing is a high performance mechanism in cloud computing that is used to distribute the load equally to the virtual nodes to achieve highly scalable cloud network when the workload is high [5]. Load balancing techniques also designed to achieve green computing under the greedy approach to use the resources in an optimal manner and to avoid the hazards like server failure, high response time and to avoid bottleneck conditions. Load balancing algorithms must have major qualitative metrics for better resource management discussed as follows:

(i) Throughput (ii) Migration time (iii) Utilization of resources

(iv) Response time (v) Scalability (vi) Fault tolerance

2. BACKGROUND STUDY

Load Balancing Algorithms are basically falls under two categories Adaptive and Non Adaptive. Non adaptive load balancing algorithms are also called static, non-preemptive approaches under which every node has at least one task to implement. In this article we emphasized on Mata Heuristic, non-adaptive and dynamic load balancing approach that is inspired by nature and based on natural selection strategy.

Task scheduling is an NP hard problem hence either heuristic or Meta heuristic algorithms will be best suited for getting better throughput and scalable service. When we go through the article Milani et al, (2016) it has found that a lot of work (about 23% of total Load Balancing) has been done to improve the task migration using dynamic load balancing mechanism but majority had not shown any desire to improve it through hybrid load balancing schemes [10]. Whatever the reason were, but not cleared or there is misguidance that why the important load balancing metric (migration time) was not discussed in any of the hybrid techniques this is actually the gray area or research gap. We have worked in this grey area and try to review comprehensively of famous dynamic load balancing algorithms. In this article we found that there is no single dynamic load balancing technique which could reduce the task migration time with a highly scalable mechanism.

Cloud computing is the most emerging technologies and adopted at a big level in software companies to develop or host various platforms. Virtualization is also the key factor of cloud computing used to deploy the concurrent execution of parallel tasks. At every level the lifetime of a task is maintained for example: first task is uploading then assignment then execution then migration if needed and finally downloading, in order to maintain this sequence of tasks, task scheduling has to be done which is called load balancing in cloud environment. Task migration is considered very important aspect for healthy load balancing, the flow chart of task migration is shown in the following figure 1 [22].

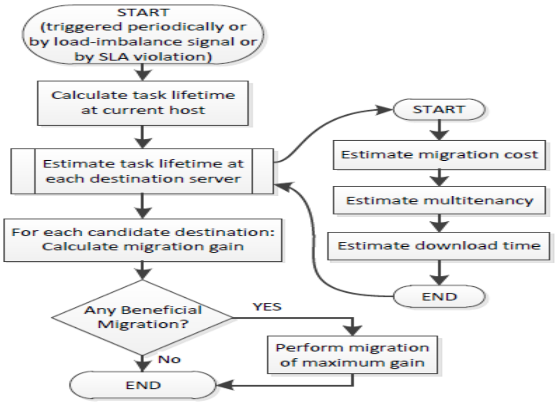

Fig 1: Flowchart of Task Selection and Migration

3. RELATED WORK

One of the basic building blocks of datacenters are Virtual machines (VMs), due to cost savings, elasticity, and ease of administration they are used to provide information as a service (IAAS) to enhance high performance computing (HPC) and to backup cloud computing. The scale of live migration can range from migrating VMs across only a few physical machines to entire racks of physical machines. Live migration traffic also consumes the bandwidth at the source and target network interfaces and competes with the bandwidth requirements of applications running within the VMs. Task scheduling the term of operating systems where multiple processes run concurrently to share the resource. In distributed computing task scheduling is a big challenge from the point of view to managing the requests of the server by different clients. As requests being an overhead for a server due to number of requests crosses the limit of server in that case there may be a chance of hardware failure. Cloud, an enhanced distributed services that is availed by users anywhere and anytime by third party also needed a synchronized mechanism to distribute the load evenly to the servers.

In cloud computing, Load balancing is very burning area where a lot of work has been done in recent years. Here author used scheme that improves the migration time of tasks during dynamic load balancing using particle swarm optimization technique. There are several disadvantage of all VM migration first one is majority attempted overloaded VMs for migration, second it prepares dirty space for Precopy mechanism for online migration of VM’s, third, it requires a large memory for both physical and for host machine, another drawback is its lower downtime due to pausing the primary VM. Ramezani et al, (2014) proposed Task-based system load balancing method which not only attempts overloaded VM for migration but migrate only the extra tasks but also overcome previous disadvantages [11]. This mechanism does not provide the pausing mechanism of VM so less time will be required for task migration. However this approach only migrates the independent arrival jobs from overloaded VM. The problem of finding optimum solution for allocating extra tasks from overloaded VM to other VM is an open issue, so need to solve in future.

Task scheduling is an NP hard problem in cloud environment, the load has to be balanced whenever some VM are under loaded and some are overloaded to achieve efficient utilization of machines. Krishna et al, (2013) introduced new approach (HBB-LB) honey bee behavior inspired load balancing that removes the tasks from the heavily loaded machine by deciding the priorities of jobs [12]. Author compared the honey bee from removed tasks. This method reduces the time of jobs that are waiting in the queue, hence lesser number of migrations and response time also minimized though this mechanism. Frequent Migration of task affects the performance of cloud environment, so here author proposed a modified bee colony algorithm to optimize the make span time and number of migrations. In section of experimental results author compared the make span time before and after the load balancing of HBB-LB with other two approaches DLB and FIFO, also compared the make span time of HBB_LB, FIFO, DLB and WRR. Author makes the graphs here between number of tasks and task migrated and compared HBB_LB, FIFO, DLB and WRR in all graphs it is clear that HBB_LB is the best load balancing approach. However it may not highly scalable mechanism since it considers the tasks independently. In future we can improve the scalability of this algorithm. In order to maintain the QOS of cloud eco system the request from overloaded system must be transferred to under loaded VM. The algorithm tries to achieve minimum response time and completion time.

To improve the system stability Babu et al, (2016) modified the older version of nature inspired bee colony optimization technique by considering lower priority of tasks while migration is needed [13]. Author basically focus that lower priority task will have more chance of migration so higher priority task will unaffected hence this algorithm always selects the least priority task for migration to reduce the load disparity so that no tasks need to wait for a long time. In this article author compared the task migration, degree of imbalance and make span time between the Bee colony and enhanced Bee colony algorithms. The future scope of this approach is further enhancement by hybridization using nature inspired algorithms like Ant colony algorithm.

Cloud data centers having heterogeneous service using several VMs with different specifications, due to dynamic requests they fluctuates with resources consumption. In the reference of dynamic load balancing some servers may overload some under loaded so this may cause performance degradation. Gutierrez G et al, (year) proposed a cooperative agent based problem solving technique for load balancing using live VM migration [14]. Here VM are determined the destination of migrated task, the exact time when to migrate and VM hosted policies. This is novel approach that is very helpful to distribute the load evenly across the servers to get high performance of cloud environment or getting better QOS. This work contributes agent based load balancing for data centers in distributed fashion with deciding the priorities over the cloud during the VM migration. Future work will focus on estimating VM migration overhead and developing a resource. However this approach is not fully scalable, having high VM migration and it is centralized approach.

In this article author a market-based control method (MBA), to address the load balancing in cloud database using allocation and dynamic VM migration [15]. It basically a market based control approach that works on phenomenon of buyers and traders like in cloud system server and host respectively. In MBA, database nodes are treated as traders in a market, and certain market rules are used to intelligently decide data allocation and migration Buyer: buys the load from seller (or receives the data) Seller: Sells its load or sends its records A buyer is constrained to purchase a price no more than its limit price, and a seller to accept a price no lower than its limit price. If buyer pays over cost than its private value or seller takes lower trading considered to be at loss. So for this scheme work properly a buyer always gives a lower bid price than its private & seller always gives the higher ask price than its monotonic function.

This article describes the Extremal optimization approach for CPU load balancing in distributed environment. The EO approach detects the appropriate task for migration as well as steered collection of the best computing nodes to receive the migrating tasks that basically reduces the migration overhead by using two fitness functions based on unambiguous models which estimate relations between the programs and the hardware. Extremal optimization algorithm uses best fit selection strategy to choose the task for migration and also considers the best placement on destination host. Author proposed an optimal load balancing scheme for faster convergence to heuristical as well as better quality of service and tried to minimize the task migration overhead by applying guided search for best possible target selection (VM placement) by implementing standard EO algorithm with guided state change (EO-GS) [16]. Load balancing quality with EO improved by the guided search of the migration targets is showed by comparison in most cases better than that of the other algorithms. Through graphs author tries to build the relationship between average speed up and migration number for the same four algorithms as discussed earlier. In future we can more simplify the EO algorithm since it suffers from low scalability and response time is very high.

The paper reviews comprehensively the application of metaheuristic techniques in the area of scheduling in cloud environments. In this paper author focused on one of the important issue that is scheduling, goal of scheduling is much optimized resource mapping. In cloud computing, Task scheduling is considered NP Hard problems due to having a large domain of solution, since there is no algorithm that provides the optimal solutions in polynomial time. Hence metaheuristic approaches (nature inspired algorithms) are supposed to be the best algorithms that produces the nearest optimal results in reasonable time for these NP hard problems. The main focus of this paper is to reduce the energy consumption, since according to survey 50% energy is consumed for cooling the data centers because a lot of heat is produced by resources [17]. To reduce the heating of overloaded VM, tasks are migrated to under loaded VM. However it is very challenging to reduce the energy consumption without losing the performance in terms of when and how the task is migrate to keep environment cool and safe. Security and privacy aware scheduling is another area which needs to be explored using metaheuristic techniques. Future investigations are required to perform scheduling in a way that it protects the sensitive and/or private information associated with tasks/users.

To meet the ever increasing demands of dynamic load by relocating VMs in data centers virtualization is empowered by VM migration. In this article author proposed state of art bandwidth optimization methods to minimize the energy consumption. Through an exhaustive literature review Ahmad et al, (2015) elaborate the live VM migration techniques, thematic classification of VM migration is investigated through a set of parameters, the commonalities and variances are highlighted to get some research issues that originates the necessitate future consideration of optimal VM migration schemes [18]. This paper elaborate VM migration, server consolidation, dynamic voltage and other power optimization schemes. Here VM migration area highlights some open challenges and future trends regarding developing best optimal VM migration, server consolidation domain emphasized power optimization in data centers using miscellaneous modes. Further lively workload behavior in distributed VM migration schemes design can reduce the overall effort and speedup processing of VM migration. There are various challenges in field of VM server consolidation framework. Very first challenge alarms maintaining accuracy during resource demand estimation in cloud environment, second challenge concerns that in majority of research articles the issue of security was ignored and third one is regarding optimal distribution of VM in dynamic resource demand applications. Finally conclusion of this paper is to design the dedicated, fast communicative and heterogeneous nodes for VM storage for reducing service downtime and migration time.

Another optimal solution for migration is presented in Beloglazov et al, (2010) by switching off the ideal physical nodes to minimize the energy consumption in cloud VM migration environment [19]. This is a decentralized scheme of resource management system for cloud data centers handled by global policies applying to live migration to relocate the VMs. This article actually presented VM resizing, scheduling and migration policies in command forms, at earlier optimization stage the resource utilization is observed and to minimize the energy consumption. Since there is trade-off between performance and energy saving, to solve this problem, author implemented the policy to set the upper and lower threshold for CPU utilization. If utilization is greater than upper limit some VM has to migrate, if it goes underutilization then switch off the node just after the migration of all VMs. However there are two main problems one is switching off the nodes in case of underutilization but there is no comment about when the node is switched on or restarted if there is requirement, since if nodes will off then it may be possible that some of nodes again suffer from overutilization in this case we need more nodes. Other problem is author has not given any clue to determine particular values of utilization threshold.

VM migration is very burning area of cloud computing, many system pause VM, copy the state data and then resume the VM on destination host. These systems cause application to become unavailable during the migration process. Author proposed ZAP that could achieve lower down time of the service by transferring a process group to move the VM among hosts in LAN without disrupting its services [20]. ZAP uses partial OS visualization to allow the process domains (PODS) using a modified kernel. This approach is isolating every process to kernel interfaces like sockets and file handlers, into a namespace that could be migrated. This scheme is considered to be faster than results in the collective work due to migration. However ZAP do not address the problem of maintaining open connections for exiting services as well as uses stop and copy mechanism.

Also Deshpande et al, (2012) proposed an approach for VM migration using Inter rack live migration (IRLM) using Precopy technique that decreases the traffic load during VM migration on primary networks by deduplication of VM storage [21]. Due to several reasons such as maintenance, power saving, system performance and most important is load balancing we need to migrate the requests within the data centers. In this article author proposed a parallel live migration approach using live migration inter rack distributed system using technique in QEMU/KVM. Simulated results showed that total traffic on core networks and migration time is reduced by 43% and 15% respectively. Existing live migration schemes reduces the data transferring either focus on optimizing the single VM migration or multiple VM that are running on same machine but in this article author presented live migration through inter racking. IRLM technique is implemented in two parts one is preparation and other is migration phase. Under preparation phase basically duplicates the contents through hashing after it migration phase is ready under which VMs are migrated in parallel to the destination site. However this scheme could be implemented in future using post copy approach in which target-to-target page transfers can be delayed until after the resumption of VMs at the target rack.

Furthermore Lazaros Gkatzikis et al, [22] proposed a mobile cloud migration approach (MCC system architecture) which brings the cloud nearer to mobile user to avoid communication delay. Virtualization provides the prospective for quick and on demand formation of physical machines to run various tasks, hence to avoiding resource wasting, VM migration tools enable cloud suppliers to adjust the load at every data server. Due to under load or overload of the host, VM may be migrated to another server and may continue execution form the point of view of minimal power consumption also it is called VM migration and may occur multiple times during task execution for resourceful load balancing. Migration should be transparent to the applications, so every VM is migrated along with its current state of execution so that it can resume from its previous state. After the complete execution of task the result is transferred to the user by the data center where the user has submitted task. Here author discussed many challenges like workload uncertainty, Unpredictability of Multitenancy Effects, Unknown Evolution of Accompanying Data Volume, Partial availability of Cloud-Related Information and many more.

3.1 Classification of VM Migration Schemes

The time to relocate or shift the tasks from one specific candidate node to another for execution is known as transfer time of migration time. For better system efficiency migration time will need to be minimized, and we should keep this low as we can. VM migration schemes falls under three categories namely:

3.1.1 Bandwidth optimization: Bandwidth optimization falls under three types (a) Precopy Migration (b) Post copy Migration (c) Hybrid Migration.

Virtual Machine migration techniques monitor either post copy [21, 25, 26], hybrid [27, 28], or Precopy [21, 23, 24] migration designs to migrate VMs across data centers. VM migration methods use minimized network bandwidth so VM can deal with large data (up to hundred GBs). VM Migration deeds deduplication [21], compression [24] and fingerprinting [29] to improve network performance and application.

3.1.2 DVFS enabled Power optimization: DVFS-enabled migration algorithms uses a prototype while considering a single-core [28, 31] or multi core [30, 31] processors. The proposed optimization schemes exploit DVFS (Dynamic voltage and frequency scaling) [28] or power capping to minimize power consumption during hosting different workloads such as e-commerce [30], scientific [31], or integer operation-based applications [32].

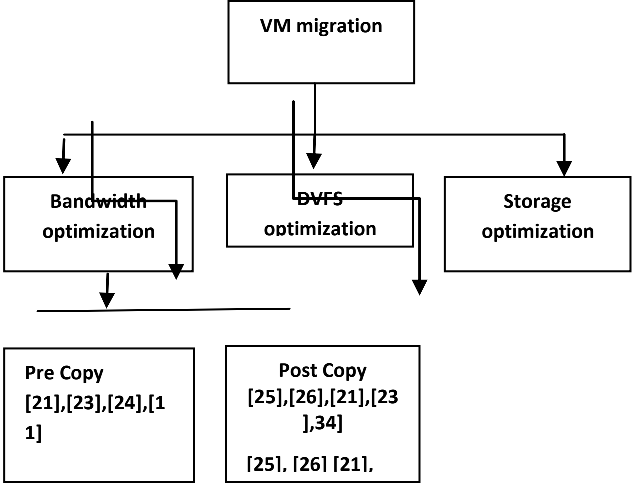

Fig 2: Classification of Migration Schemes

3.1.3 Storage optimization: Storage optimization covers two services target and proxy servers, both are connected to source and destination servers through a network block device connection (NBD) [33]. Prototype implementation of I/O blocked live storage migration [34] rapidly relocates disk blocks with in WAN links with less power on I/O performance.

4. RESEARCH METHODOLOGY

The word Meta-heuristic is used to search optimal solutions in most of the problems in a cloud environment. To achieve high performance for most of the scheduling problems, meta-heuristic techniques are used. Recently, this approach is used to solve the NP-hard optimization problems. Table 1 covered the non-static load balancing schemes which tried to maximize the cloud system by optimize the important metrics. We have critically surveyed about 26 articles of respected publications having good cited index. In table 1 different dynamic load balancing techniques proposed by various researchers have been analyzed with comparative analysis of different existing load balancing techniques with respect to different performance parameters. When we aggregated demerits of existing task migration schemes we get some new to directions towards future.

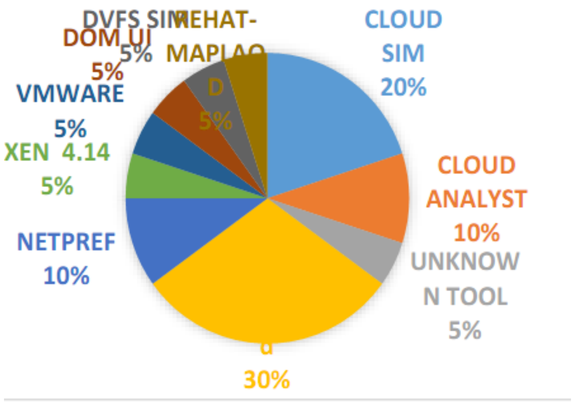

Fig 3: figure of simulation tool used

Table 1: Key Points of Previous Related Survey

Author, Year	**Author's Contribution**	**Algorithm Used**	**Demerit**
[20] Osman, 2002	Less memory required for VM migration	ZAP system	Use pausing mechanism
[28] Akshat , 2008	DVFT cost aware optimization	Power-minimizing Placement	High response time
[25] Hines, 2009	Comparing Precopy and post copy	Self-ballooning compression	
[31] Laszewski, 2009	Reduces wastage of server energy	INTERVAL Algorithm	High transfer time
[34]Hirofuchi, 2009	Storage optimization VM live Migration over WAN		Less Scalable
[29] Xhang, 2010	Reduced Migration Time	MOD algorithm	
[19] Anton , 2010	Decentralized resource management system for Cloud	VM allocation policies	Utilization threshold not defined
[24] Svärd, 2011	Delta compression technique		
[27] Sahani, 2012	Hybrid migration using bandwidth optimization	KVM/QEMU	Less scalable
[21] Deshpande, 2012	Reduced 26% migration time	QEMU/KVM /IRLM	Not efficient using Precopy
[12] Dhinesh, 2013	Reduced task Migration Time	HBB_LB	Not a scalable mechanism
[23] Lazaros, 2013	Improved Precopy live migration	Sonic migration algorithm	
[32] Jeong, 2013	Reduced Migration Time using Precopy scheme		Less downtime
[26] F Yin, 2014	Three stage memory post copy Live VM migration	LZO compression	
[11]Ramezani , 2014	Less memory required for VM migration	TBSLB_PSO	extra tasks allocation
[35] Rani , 2015	Reducing Migration Overhead		
[14]Gutierrez , 2015	23% work done of task migration in dynamic LB schemes	MAPLOAD	Not fully scalability found
[15] Wang , 2015	Bid and Ask policies used to reduce migrations	Market Clearing Algorithm	Overhead to filling buyer's need
[17] Mala , 2015	Reduction of energy consumption by migrating the tasks		High migration overhead
[18] Ahmad , 2015	Optimal VM deploying over datacenters		
[16] Ivanoe, 2015	Reduced the complexity of task selection for Migration	EO-GA	Suffers from low scalability
[10] Milani, 2016	Agent based problem solving with VM migration policies		
[13] Ramesh , 2016	Improved QOS by reducing Migration Time	Enhanced Bee colony	Not fully scalable
[2] Tim Yu, 2017	Reduced memory migration time	WWS	Downtime is not changed
[37] Osanaiye, 2017	Live migration of VM	Stop N Copy	Low latency
[38] Aznoli, 2017	Concluded that majority not worked security issues		high response time

Table 1 covers contribution of author, algorithms used and the most important thing demit (term need to improve in future) is covered so that researchers can work on weak and lower attention area as well as the picture of future trends should be clear like majority of schemes suffers from low downtime, low response time with less failure.

4.1 Selection of literature

Different researches in the area of cloud computing have shown that after 2007 there is rapid change to publish the articles of cloud computing using load balancing metrices. Data selection phase summarize from the selected studies for further revivew is we foumd 467 articles when we search with specific keyword, out of them we studied abstract, conclusion with contribution of author of primary papers from some prestigous journols. By inclusion exclusion criteria finally we gone through full body text of 30 research papers that were considered as the primary for the review of literature.

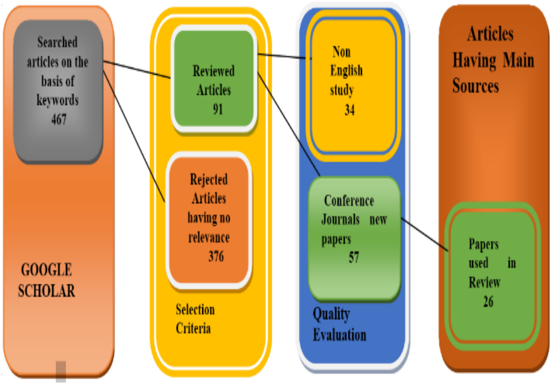
Searched articles on the basis of keywords 467
GOOGLE SCHOLAR
Reviewed Articles 91
Rejected Articles having no relevance 376
Selection Criteria
Non English study 34
Conference Journals new papers 57
Quality Evaluation
Articles Having Main Sources
Papers used in Review 26

Fig 4: Overview of seletcted article documentation process

Figure 3 is showing the overview of the selection process by diagramatic fashion. The inclusion exclusion crieteria of of the slection of sources as as folows:

- **Inclusion 1** Study which is clearly describing load balancing metrics
- **Inclusion 2** Study must follow english lietrature
- **Inclusion 3** Study is published in the field of cloud computing
- **Inclusion 4** Study should be make by reaserchercher in peer revived publictaion.
- **Inclusion 5** Study should be from reputated confrences or journols

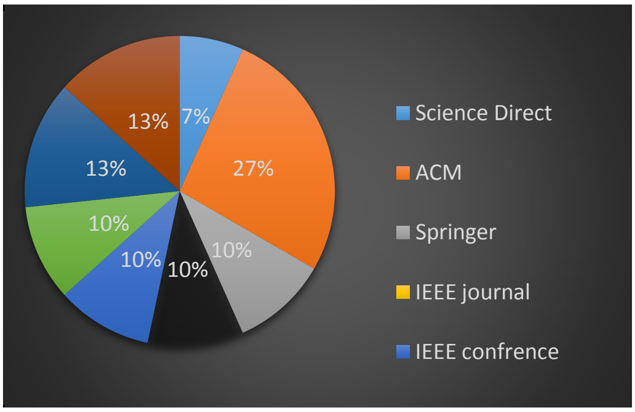

Fig 5: Percentage of published papers in publication

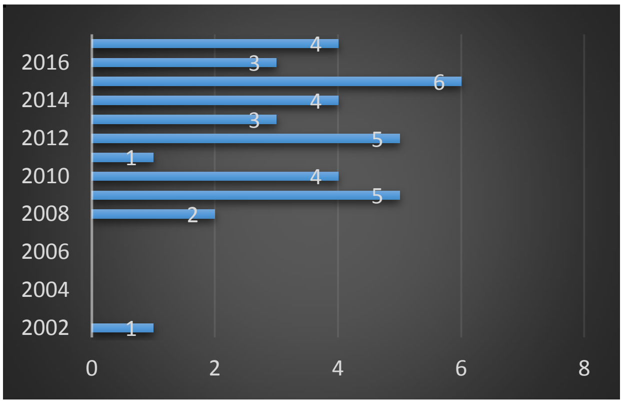

Fig 6: Chronological Review of publications in area of migration time

We attempted to conduct this systematic review as rigorously as possible. However, it might have still endured from several validity threats. Hence, future efforts, to interpret or directly utilizing the reviewed or conclusions in this systematic review should bear some limitations like. (i) Research questions (ii) Research scope

4.2 Discussion

In this section we presented a summarized performance from comprehensive analysis of various famous dynamic load balancing schemes through table 1, figure 5 and figure 6. This can be seen after systematic revive of migration reduction algorithms that a lot of work has been done to minimize the migration time (task transfer time) to improve the system performance which a very important metrics of task scheduling. Through table 1 we tried to show the algorithms used and disadvantages of the existing migration reduction techniques which shows that majority of researchers did not focus to improve the downtime and scalability of load balancing algorithms. The chief drawback of existing techniques is trade-off between migration time and scalability so in future we need to work in this direction. In addition different tools are used to simulate and compare the results but this literature showed that CLOUDSIM is very popular and best tool to create real cloud environment since it is open source also that is why we can use it very easy manner. As shown in figure 6, the rapid amelioration has been occurred immediate after year 2008 in the field of cloud computing.

5. OPEN ISSUES AND FUTURE SCOPE

This section proposes primary issues of load balancing techniques with reduced migration overhead that have not been analytically investigated yet. Here we have emphasized on basic performance metrics and area of application regarding popular dynamic load balancing algorithms. After in depth observation of collected information we observed that there is no autonomous technique which can addresses all issues involved in task migration like several load balancing technique with optimal migration time insures QOS (quality of service), scalability and reliability but some schemes ignored completely. Also some revived techniques used simulation tools but many have either not used any tool or not mentioned its name after using it. Since most of the techniques are using open source modelling and simulation tool to validate the results hence an attractive future study point would be to investigate the effects the size would have on server efficiency on a big scale by real cloud environment or utilizing some simulator such as Cloud Analyst or CloudSim. In all the migration techniques data centers are treated as a single pool of resources, in case of any undesirable failure no scheme is holding water from the point of view scalability. An adaptive task scheduling algorithm should be robust, and resourceful but in this literature review through a table we have shown that schemes are able to reduce the task migration time in dynamic environment but mostly are not scalable so in future we need a highly scalable mechanism that can transfer the jobs from one node to another in short span. Another area that get a low attention that dynamic migration techniques are not included failure management and downtime.

6. CONCLUSION

This literature presents a systematic review of dynamic task migration schemes in the field of cloud computing, we revived relevant dynamic task migration & load balancing articles, discussing & illuminating open queries by deeply investigated about 30 articles from 465 papers. In this study the ideas of cloud computing, task migration, DVFT based migration, storage migration, Precopy, post copy mechanism and bandwidth optimization techniques are highlighted, while challenges and future directions are highlighted in open issue section in order to design optimal task migration strategy. Here we found the evidences clarifying that healthy task migration is great mechanism that ability to keep minimum resource consumption, maximize throughput and high scalability in order to maintain the server heterogeneity. In this paper we systematically surveyed the past mechanisms of task migration by classified them in many aspects, we also highlighted the disadvantages under each and every category. Proper task migration polices in cloud environment still need a scalable algorithm that can basically take minimum time to migrate the requests from one host to other. The collected statistics from various sources help researchers to introduce the current challenges, new directions, future trends, defined queries and open research issues to heighten a good level to laod balancing system in the are of cloud computing.

7. REFERENCES

[1] Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A., & Buyya, R. (2011). CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Software: Practice and experience, 41(1), 23-50.

[2] Wu, T. Y., Guizani, N., & Huang, J. S. (2017). Live migration improvements by related dirty memory prediction in cloud computing. Journal of Network and Computer Applications, 90, 83-89.

[3] Sreenivas, V., Prathap, M., & Kemal, M. (2014, February). Load balancing techniques: Major challenge in Cloud Computing-a systematic review. In Electronics and Communication Systems (ICECS), 2014 International Conference on (pp. 1-6). IEEE.

[4] Gupta, A., & Garg, R. (2017, September). Load Balancing Based Task Scheduling with ACO in Cloud Computing. In Computer and Applications (ICCA), 2017 International Conference on (pp. 174-179). IEEE.

[5] Verma, P., Shrivastava, S., & Pateriya, R. K. (2017). Enhancing Load Balancing in Cloud Computing by Ant Colony Optimization Method.

[6] Sarma, P., Chana, I. G., & Bala, A. G. (2017). Optimized Hybrid Task Scheduling Algorithm in Cloud (Doctoral dissertation).

[7] Arya, P. S. S. K., & Tripathi, P. (2016). Various Issues & Challenges of Load Balancing Over Cloud: A Survey. International Journal of Engineering and Computer Science, 5(8).

[8] Ghumman, N. S., & Sachdeva, R. (2016). An Efficient Approach for Load Balancing in Cloud Computing using Composite Techniques. International Journal of Research in Engineering and Applied Sciences, 6(2), 145-149.

[9] Xu, M., Tian, W., & Buyya, R. (2017). A survey on load balancing algorithms for virtual machines placement in cloud computing. Concurrency and Computation: Practice and Experience, 29(12).

[10] Milani, A. S., & Navimipour, N. J. (2016). Load balancing mechanisms and techniques in the cloud environments: Systematic literature review and future trends. Journal of Network and Computer Applications, 71, 86-98.

[11] Ramezani, F., Lu, J., & Hussain, F. K. (2014). Task-based system load balancing in cloud computing using particle swarm optimization. International journal of parallel programming, 42(5), 739-754.

[12] Harraz, A., Cherkaoui, R., Bissiriou, C., & Zbakh, M. (2016, May). Study of an adaptive approach for a Cloud system implementation. In Cloud Computing Technologies and Applications (CloudTech), 2016 2nd International Conference on (pp. 230-236). IEEE.

[13] Babu, K. R., & Samuel, P. (2016). Enhanced bee colony algorithm for efficient load balancing and scheduling in cloud. In Innovations in bio-inspired computing and applications (pp. 67-78). Springer, Cham.

[14] Gutierrez-Garcia, J. O., & Ramirez-Nafarrate, A. (2015). Agent-based load balancing in cloud data centers. Cluster Computing, 18(3), 1041-1062.

[15] Wang, T., Lin, Z., Yang, B., Gao, J., Huang, A., Yang, D., & Niu, J. (2012). MBA: A market-based approach to data allocation and dynamic migration for cloud database. Science China Information Sciences, 55(9), 1935-1948.

[16] De Falco, Ivanoe, et al. "Extremal Optimization applied to load balancing in execution of distributed programs." Applied Soft Computing (2015): 501-513.

[17] Kalra, M., & Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. Egyptian informatics journal, 16(3), 275-295.

[18] Ahmad, R. W., Gani, A., Hamid, S. H. A., Shiraz, M., Yousafzai, A., & Xia, F. (2015). A survey on virtual machine migration and server consolidation frameworks for cloud data centers. Journal of Network and Computer Applications, 52, 11-25.

[19] Beloglazov, A., & Buyya, R. (2010, May). Energy efficient resource management in virtualized cloud data centers. In Proceedings of the 2010 10th IEEE/ACM international conference on cluster, cloud and grid computing (pp. 826-831). IEEE Computer Society.

[20] Osman, S., Subhraveti, D., Su, G., & Nieh, J. (2002). The design and implementation of Zap: A system for migrating computing environments. ACM SIGOPS Operating Systems Review, 36(SI), 361-376.

[21] Deshpande, Umesh, Unmesh Kulkarni, and Kartik Gopalan. "Inter-rack live migration of multiple virtual machines." Proceedings of the 6th international workshop on Virtualization Technologies in Distributed

Computing Date. ACM, 2012.

[22] Gkatzikis, L., & Koutsopoulos, I. (2013). Migrate or not? Exploiting dynamic task migration in mobile cloud computing systems. IEEE Wireless Communications, 20(3), 24-32.

[23] Koto, A., Yamada, H., Ohmura, K., & Kono, K. (2012, July). Towards unobtrusive VM live migration for cloud computing platforms. In Proceedings of the Asia-Pacific Workshop on Systems (p. 7). ACM.

[24] Svärd, P., Hudzia, B., Tordsson, J., & Elmroth, E. (2011). Evaluation of delta compression techniques for efficient live migration of large virtual machines. ACM Sigplan Notices, 46(7), 111-120.

[25] Hines, M. R., Deshpande, U., & Gopalan, K. (2009). Post-copy live migration of virtual machines. ACM SIGOPS operating systems review, 43(3), 14-26.

[26] Yin, F., Liu, W., & Song, J. (2014). Live virtual machine migration with optimized three-stage memory copy. In Future Information Technology (pp. 69-75). Springer, Berlin, Heidelberg.

[27] Sahni, S., & Varma, V. (2012, October). A hybrid approach to live migration of virtual machines. In Cloud Computing in Emerging Markets (CCEM), 2012 IEEE International Conference on (pp. 1-5). IEEE.

[28] Verma, A., Ahuja, P., & Neogi, A. (2008, December). pMapper: power and migration cost aware application placement in virtualized systems. In Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware (pp. 243-264). Springer-Verlag New York, Inc.

[29] Zhang, X., Huo, Z., Ma, J., & Meng, D. (2010, September). Exploiting data deduplication to accelerate live virtual machine migration. In Cluster Computing (CLUSTER), 2010 IEEE International Conference on (pp. 88-96). IEEE.

[30] Wang, Z., Zhu, X., McCarthy, C., Ranganathan, P., & Talwar, V. (2008, June). Feedback control algorithms for power management of servers. In Third International Workshop on Feedback Control Implementation and Design in Computing Systems and Networks.

[31] Von Laszewski, G., Wang, L., Younge, A. J., & He, X. (2009, August). Power-aware scheduling of virtual machines in dvfs-enabled clusters. In Cluster Computing and Workshops, 2009. CLUSTER'09. IEEE International Conference on (pp. 1-10). IEEE.

[32] Jeong, J., Kim, S. H., Kim, H., Lee, J., & Seo, E. (2013). Analysis of virtual machine live-migration as a method for power-capping. The Journal of Supercomputing, 66(3), 1629-1655.

[33] Morrison, D. G., & Schmittlein, D. C. (1988). Generalizing the NBD model for customer purchases: What are the implications and is it worth the effort?. Journal of Business & Economic Statistics, 6(2), 145-159.

[34] Hirofuchi, T., Ogawa, H., Nakada, H., Itoh, S., & Sekiguchi, S. (2009, May). A live storage migration mechanism over wan for relocatable virtual machine services on clouds. In Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (pp. 460-465). IEEE Computer Society.

[35] Rani, A., & Kaur, P. (2015). Migration Jobs in Cloud Computing. International Journal of Grid and Distributed Computing, 8(6), 151-160.

[36] Choudhary, A., Govil, M. C., Singh, G., Awasthi, L. K., Pilli, E. S., & Kapil, D. (2017). A critical survey of live virtual machine migration techniques. Journal of Cloud Computing, 6(1), 23.

[37] Osanaiye, O., Chen, S., Yan, Z., Lu, R., Choo, K. K. R., & Dlodlo, M. (2017). From cloud to fog computing: A review and a conceptual live VM migration framework. IEEE Access, 5, 8284-8300.

[38] Aznoli, F., & Navimipour, N. J. (2017). Cloud services recommendation: Reviewing the recent advances and suggesting the future research directions. Journal of Network and Computer Applications, 77, 73-86.