ISSN:2231-0711

Kavyashree.V et al | IJCSET(www.ijcset.net) | 2017 | Volume 7 Issue 5, 35-40

Comparing VMware's DRS Migration with a Human Expert Migration

Kavyashree.V, Prasad A.Y
Dept of ISE, RRCE
Bangalore, India

***Abstract*— As the technology is growing rapidly, huge amount of data is being generated. The generated data can either be from simple devices like mobiles, weather, banks and many more. These data is loaded to the virtual environment. Virtualization provides a greater flexibility in terms of sharing up of resources. Next challenge that we face is balancing of load on such virtual machines (VM). VMware DRS is a tool that automatically balances the load in VMs. Here we consider large telecommunication application and measure the performance of host before and after the migration in terms of CPU utilization. Then we compare the results of VMware DRs migration with Human Expert. DRS could balance load in 40% of cases and in rest of the cases it could not balance the load as expected. In few cases it did unnecessary migration such as back and forth migration.**

***Keywords*—Cloud Computing, Distributed resource Scheduler (DRS), Virtual Machine Migration, Virtualization, VMware**

I. INTRODUCTION

Virtualization is a technology where we separate the Operating system with underlying hardware. With this we can have n number of Operating System on same work station. This enhances the CPU utilization in terms of memory and resources. Today ITs implementing virtualization as it is cost effective as we don't have to invest more on workstation of different Operating System.

There are number of benefits of virtualization and one such advantage is dynamic migration of virtual machine on physical machine cluster. To balance the load an VMs resources from heavily loaded hosts are migrated to lightly loaded hosts. This helps us to maximize the resource utilization by optimizing the mapping of VM to hosts.

A. Distributed Resource Scheduler (DRS)

VMware DRS is a tool that is provided by Virtualization. It monitors the system resources and migrates the VM to balance the load using VMotion command. All hosts are added to DRS cluster which balance the entire system. VMware VCenter is one which continuously monitors CPU and memory usage of all VMs in the cluster. DRS migrates the VM within the cluster to ensure that there is even distribution of load in hosts. DRS has a fixed threshold upon which it decides load on VMs. If the machine is loaded beyond the threshold, it automatically selects the VM for migration.

DRS takes decision of resources management according to matrices related to VMs hosts, memory and CPU utilization. DRS has different levels of aggressiveness, according to which DRs migrates, namely: conservative (level 1), moderately conservative (level 2), moderate (default) (level 3), moderately aggressive (level 4), and aggressive (level 5).

In this paper, using Virtualization technology we have built testbed with number of VMs. To know the migration of VMs we have triggered the load on each of the VM and we have compared the results of DRS algorithm with human expert migration decision.

II. EXPERIMENTAL SETUP

A. Testbed Setup

The testbed setup is as shown in the Figure 1. It comprises of three physical hosts running VMware ESXi 5.5.0. On top of VMware ESXi 5.5.0, RedHat Enterprise Linux, Version 6.2 has been introduced as a guest OS. Each host is designed with 128 GB of RAM, two 6-center CPUs (2x Intel XEON 2.0 GHz) with hyper threading empowered in each center (i.e., a sum of 24 consistent centers). These hosts make a DRS

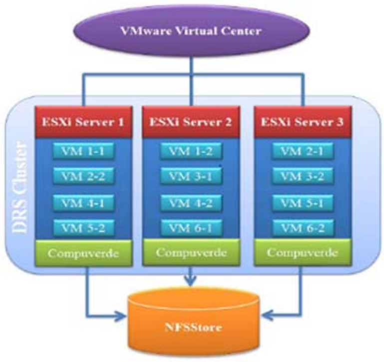

Fig 1. Testbed Setup

cluster. All hosts in our DRS cluster are associated with the Compuverde [17] conveyed capacity framework which gives 2.55 terabyte (TB) vNAS capacity. The Compuverde conveyed capacity framework comprises of two segments, 1) Compuverde programming, that is introduced inside a virtual machine and 2) information stockpiling this comprises of SSD store (7x Intel 330 60 GB (RAID 10+hotspare)) and disk persevering storage capacity (8x Intel 330 60 GB).

A large real time telecommunication application is introduced in all the VMs. The application handles billing related demands in at telecommunication application systems. The application fills in as a active server is clustered with one passive server. In the Experimental Setup (i.e. in Figure 1, VM 1-1 and VM 1-2 are one application cluster, VM 1-1acts as a active server and VM 1-2 acts as a passive server, and same example utilized for

whatever is left of the VMs). Both the server can get demands. In any case, all loads gotten by the passive server is sent to the relating active server. The active server at that point sends the response back to the passive server.
Movement going specifically to the active server is taken care of without including the passive server. A different host runs a test system that produces stack towards the servers running in the clusters. The test system is situated in a similar LAN(it is not shown in Figure 1). All correspondences are done utilizing Ethernet (Intel X520 SFP+ 10 GB). All hosts in the DRS cluster are monitored with VMware Virtual Center 5.5.0.

We have setup 12 VMs in cluster. Inside these VMs we have introduced our application, i.e., in complete we have 6 active-passive application clusters. We have distributed 10 vCPU and 14 GB
RAM(14 GB RAM is the base that is prescribed for the application) to each of these VMs. Inside each host 24 vCPUs are accessible. With a specific end goal to stay away from any obstruction between the VM that contains the Compuverde programming and different VMs, we have made two distinctive resource pools, one with 4 vCPUs and the other one with 20 vCPUs. The VM containing the Compuverde programming has been bound to the resource pool with 4 vCPUs and whatever is left of the VMs are bound to the resource pool that contains 20 vCPUs.

B. Test Scenarios

Test scenarios are always done in such a way that it always gives us better solution in terms of load balancing. Test scenarios are as follows:

1. There are three hosts and each host contains 4 VMs. All VMs get a similar number of demands every second (700, 500, and 300). All of a sudden we increment the heap on one VM (up to 2100, 1500, and 900 req/s).
2. There are three hosts and each host contains 4 VMs. All VMs get a similar number of demands every second (600, 500, and 300). All of a sudden we increment the load on two VMs on two distinct hosts (up to 2500, 2000, and 900 req/s).
3. There are two hosts and each host contains 6 VMs. All VMs get a similar number of demands every second (600, 500 and 300). All of a sudden we add another vacant host to the DRS cluster.
4. There are three hosts and each host contains 4 VMs. All VMs get a similar number of demands every second (700, 500 and 300). Abruptly we reduce the load
on one VM and increment the load on another VM, on two unique hosts. This test situation recreates the circumstance when one VM quits working. As we have said before the application functions as a active-passive cluster, so in the event that the dynamic server quits working the passive server will end up plainly active and requests are handle.
5. There are three hosts and each host contains VMs. All VMs get a similar number of demands every second (700, 500 and 300). All of a sudden we put one of the hosts in maintenance mode.

We will be carrying out all the test cases in three different levels for each of the test scenario. i.e. DRS aggressiveness for level 1 (conservative), level 3 (moderate/default), level 5 (most aggressive).

CPU utilization in each of the test scenario is measured in four different states, namely:

1) Distribute same amount bof load on all VMs when DRS is in off state.
2) When still the DRS is in off state, shift the load on one of the VMs.
3) Measurement the CPU utilization after turning on the DRS.
4) Comparing the CPU utilization after human expert has done migration manually.

In this paper we have only measured performance in state 1, 2 and 4 as there is no second state in 3, 5.

III. COMPARING VMWARE'S DRS MIGRATIONS WITH HUMAN EXPERT MIGRATION

A.Test Case 1

In the test case 1, we consider all three hosts. It is taken care that all VMs on the host receive the same number of request per second. Then we increase the load on any one of the VMs say VM 1-1(Figure 1) on host 1(H1 ESXi Server 1). So this VM suffers twice the load. We will be conducting the test for three different load scenarios and three different DRS migration threshold levels.
As found in Table I, after performing the load shift, DRs had performed migration in two of the cases (500 and 300 req/s) with migration threshold set to level 5 (aggressive). In both the cases VMs were migrated from H1 to H3. Table II demonstrates that for the most minimal load (300 req/s), the CPU usage is around 27% before moving each of the three hosts. After incrementing the load on VM 1-1 to 600 req/s, the CPU utilization on the Host 1(ESXi Server 1) is incremented to 34%. Since the passive server (VM 1-2) is running on Host 2 (ESXi Server 2), Table II demonstrates that the CPU usage on Host 2 has increment by 28%. For level 5, and 500 and 300 req/s, a human expert would do likewise as DRS (move one VM from Host 1 to Host 3) (see Table I). So for this situation DRS made a good decision.

Load (req/s)		Level of Aggressiveness			Human Expert
All VMs	Two VMs	Level 1	Level 3	Level 5	
700	1400	-	-	-	VM 2-2 from H1 to H3
500	1000	-	-	VM 2-2 from H1 to H3	
300	600	-	-	VM 5-2 from H1 to H3	

TABLE I. SUMMARY OF THE RESULTS OF THE TEST CASE 1

	CPU Utilization (%)											
	a. Low Load (300req/s)				b. Medium Load (500req/s)				c. High Load (700req/s)			
	Same Load	Load Shift	Migra-tion	Human Expert	Same Load	Load Shift	Migra-tion	Human Expert	Same Load	Load Shift	Migra-tion	Human Expert
H1	27	34	31	31	38	49	44	44	49	67	67	58
H2	27	28	28	28	38	40	40	40	49	51	51	52
H3	26	26	30	30	37	38	44	44	48	49	49	57

TABLE II. TEST CASE 1 RESULT, CPU UTILIZATION

For the instance of medium load (500 req/s), Table II shows that when the load was the same on all VMs, the CPU utilization was around 38% on each of the three hosts. On increasing the load on VM 1-1 to 1000 req/s, the CPU utilization on Host 1 and Host 2 expanded to 49% and 40% respectively. Later we have set the threshold of DRS to level 1, and then we have changed to level 3(direct/default), and to level 5 (aggressive). As we see in Table I, DRS starts to migrate when threshold is set to level 5.in this case DRS migrates.VM from Host 1 to Host 3. Result is tabulated as shown in Table II. On comparing the CPU utilization on all hosts with DRS migration, Host 1 is decreased to 44% and on Host 3 it is been increased to 44%, while Host 2 stayed unaltered. In this manner, the CPU utilization on all hosts after human expert movement is indistinguishable with the CPU utilization after DRS migration (see Table II).

As we see DRS did not migrate any of the VMs when load was high (700 req/s). as it can be seen from Table II, CPU usage on the Host 1 was high, 67%, while on Host 3 the CPU utilization has down to 49%, so clearly one VM ought to have been relocated from Host 1 to Host 3. In this come situation, a human expert would relocate one VM from Host 1 to Host 3 (see Table I), to adjust the CPU utilization over all hosts; so CPU use progressed toward becoming around 57% on Host 3, 58% on Host 1 also, 52% on Host 2 (see Table II). DRS did not migrate as human expert as it could not find better host for receiving.

B.Test Case 2

In Test case 2, we do a more intricate load shifts. In this case, we have expanded the load on two VMs running on two distinctive has in the meantime, while all different VMs are accepting a similar measure of load. For this situation, all hosts could not bolster our past high load which was 700 req/s, in this manner we expected to decrease the high load to 600 req/s, yet the other load situations are stayed unaltered.

As found in Table III, VMware's DRS migrated only in level 5. On increasing the load upto 1800 req/s and upon turning DRS on, one can see that DRS starts to migrate one VM from Host 1 to Host 2 and in next moment it has begun to migrate other VM from Host 2 back to Host 1(Table III), this is said to be “ping pong” effect. In such situation human expert would have migrated one Vm from Host 1 to Host 2 (e.g., VM 3-2) and one VM from Host 3 to Host 2 (e.g., VM 6-2) (see Table III). As we can see from Table IV that the CPU utilization after DRS migration did not change, while after human expert relocation the CPU utilization progressed toward becoming around 62% on all hosts. Here too DRS suffered from “ping pong” effect (see Table III). As can be found in Table IV, the CPU use stayed unaltered after DRS movement while after human master movement the CPU usage moved toward becoming around 53%. In these cases DRS made better job.

Load (req/s)		Level of Aggressiveness			Human Expert
All VMs	Two VMs	Level 1	Level 3	Level 5	
600	1800	-	-	VM 3-2 from H1 to H2 VM 5-2 from H2 to H1 (ping-pong)	VM 3-2 from H1 to H2 VM 6-2 from H3 to H2
500	1500	-	-	VM 3-2 from H1 to H2 VM 5-2 from H2 to H1 (ping-pong)	
300	900	-	-	VM 2-2 from H1 to H2	

TABLE III. SUMMARY OF THE RESULTS OF THE TEST CASE 2

	CPU Utilization (%)											
	a. Low Load (300req/s)				b. Medium Load (500req/s)				c. High Load (700req/s)			
	Same Load	Load Shift	Migra-tion	Human Expert	Same Load	Load Shift	Migra-tion	Human Expert	Same Load	Load Shift	Migra-tion	Human Expert
H1	28	38	34	34	40	60	60	54	46	72	72	63
H2	28	29	34	36	40	41	41	53	47	47	47	62
H3	27	38	38	34	39	59	59	51	46	72	72	62

TABLE IV. TEST CASE 2 RESULTS, CPU UTILIZATION

C.Test Case 3

In Test Case 3, each of the two hosts consist of 6VMs. we put on the same amount of load on all VMs. we add a vacant server as there is no load shifts.

In this case for the highest load (600 req/s), when the migration threshold for DRS was level 3 (moderate/default) it started to migrate two VMs from Host 1 to Host 3 and two VMs from Host 2 to Host 3, and at the end it migrated back one of the VMs from Host 3 to Host 2, i.e., the “ping-pong” effect (see Table V). As can be seen in Table VI, the CPU utilization of Host 1, Host 2, and Host 3 after DRS migration became 36%, 53%, and 48% when the threshold for migration was set to moderate (level 3). The human expert migrations were two VMs from Host 1 to Host 3 and two VMs are migrated from Host 2 to Host 3 (see Table V). At the end we can observe that the CPU utilization has been evenly distributed to all hosts after human expert migration (see Table VI). For the same load (600 req/s), when the DRS migration threshold has been set to level 5(aggressive), DRS migrated two VMs from Host 1 to Host 3 and two VMs from Host 2 to Host 3, but after sometime migrated back one ofthe VMs from Host 3 to Host 2, i.e., the “ping-pong” effect (see Table V). For the situation with medium load (500 req/s), when the DRS relocation limit was set to level 3 (direct/default), DRS relocated two VMs from Host 1 to Host 3 and one VM from Have 2 to Host 3 (see Table V). Table VI demonstrates that after DRS relocation the CPU usage on Host 1 diminished to 31% while on Host 2 it ended up noticeably 47and on Host 3 it ended up noticeably 42%. On the off chance that we contrast this and what a human master would do, we can watch that after human master movement the CPU use will progress toward becoming around 40% on all hosts. For the same stack (500 req/s) and the DRS relocation limit, level 5 (forceful), DRS relocated two VMs from Host 1 to Host 3 furthermore, one VM from Host 2 to Host 3. After some time DRS moved one more VM from Host 2 to Host 3 (see Table V).

Load(req/s)	**Level of Aggressiveness**			**Human Expert**
All VMs	**Level 1**	**Level 3**	**Level 5**	
600	-	VM 2-1 from H2 to H3 VM 1-1 from H1 to H3 VM 4-1 from H2 to H3 VM 3-1 from H1 to H3 VM 4-1 from H3 to H2 (ping-pong)	VM 2-2 from H1 to H3 VM 6-1 from H2 to H3 VM 1-1 from H1 to H3 VM 2-1 from H2 to H3 VM 2-2 from H3 to H2 (ping-pong)	VM 1-2 from H2 to H3 VM 2-1 from H2 to H3 VM 5-1 from H1 to H3 VM 6-2 from H1 to H3
500	-	VM 5-1 from H1 to H3 VM 2-1 from H2 to H3 VM 1-1 from H1 to H3	VM 1-1 from H1 to H3 VM 3-1 from H1 to H3 VM 2-1 from H2 to H3 VM 3-2 from H2 to H3	
300	-	VM 3-1 from H1 to H3 VM 2-1 from H2 to H3	VM 6-1 from H2 to H3 VM 3-1 from H1 to H3 VM 2-1 from H2 to H3 VM 6-2 from H1 to H3	

TABLE V. SUMMARY OF THE RESULTS OF THE TEST CASE 3

	CPU Utilization (%)											
	a. Low Load (300req/s)				**b. Medium Load (500req/s)**				**c. High Load (700req/s)**			
	Same Load	Load Shift	Migra-tion	Human Expert	Same Load	Load Shift	Migra-tion	Human Expert	Same Load	Load Shift	Migra-tion	Human Expert
H1	43	33	29	28	64	31	37	41	73	36	46	46
H2	44	34	24	29	63	47	41	40	73	53	42	45
H3	3	20	34	28	4	42	41	39	4	48	49	46

TABLE VI. TEST CASE 3 RESULTS, CPU UTILIZATION

D.Test Case

In Test Case 4, here we simulate the scenario on one VM. i.e. if one of the active-passive server stops working in an application environment, it is then handled by passive server.

For the case with high load, 700 req/s, and when we set the limit level to the most aggressive (level 5), DRS moved one VM from Host 2 to Host 1 (as see Table VII). Table VIII demonstrates that after we have killed VM 1-1 on Host 1, the CPU use wound up plainly 40% on Host 1 and 63% on Host 2. After the DRS movement, the CPU usage wound up noticeably 48% on Have 1 and 55% on Host 2. A human expert would move one dynamic server from Host 2 to Host 1 and one detached server from Host 1 to Host 2 (see Table VII). In the event that we think about the consequences of human expert relocations and DRS, we see that after human expert relocations the CPU use on all hosts progressed toward becoming around 52% while after DRS movements, the CPU use on Host 2 was as yet higher than the two different hosts (see Table VIII). In spite of the fact that DRS was not ready to adjust the stack totally, we see that DRS could adjust the heap to some degree. For the case with medium load (500 req/s), when the DRS relocation limit was set to level 5 (forceful), DRS relocated one dynamic server from Host 2 to Host 1 and one aloof server from Host 1 to Host 2, which was the same as what human master would do in this circumstance (see Table VII). Table VIII demonstrates that the CPU usage on all hosts after both the human master and the DRS movement progressed toward becoming around 40%. For the case with low load, 300 req/s, DRS did not move any VM for each of the three distinctive relocation edge levels (see Table IV). For this situation, as it can be seen from the Table VIII, that CPU usage on all hosts after the heap move is stayed unaltered. While after human master relocation we can see from the Table VIII that the CPU usage progressed toward becoming around 29% on all hosts. Here we ought to specify that DRS was not ready to settle on legitimate choices contrasting with human master choices in the vast majority of the cases with diverse relocation limit levels. Just in one of the cases it worked when the heap was medium (500 req/s) and the movement edge level was 5 (forceful).

Load(req/s)	**Level of Aggressiveness**			**Human Expert**
All VMs	**Level 1**	**Level 3**	**Level 5**	
700	-	-	VM 4-2 from H2 to H1	VM 6-1 from H2 to H1 VM 2-2 from H1 to H2
500	-	-	VM 3-1 from H2 to H1 VM 1-1 from H1 to H2	
300	-	-	-	

TABLE VII. SUMMARY OF THE RESULTS OF THE TEST CASE 4

	CPU Utilization (%)											
	a. Low Load (300req/s)				**b. Medium Load (500req/s)**				**c. High Load (700req/s)**			
	Same Load	Load Shift	Migra-tion	Human Expert	Same Load	Load Shift	Migra-tion	Human Expert	Same Load	Load Shift	Migra-tion	Human Expert
H1	28	22	22	29	40	31	40	40	53	40	48	52
H2	28	34	34	29	39	49	41	41	51	63	55	52
H3	27	27	27	28	38	38	38	38	51	50	50	51

TABLE VIII. TEST CASE 4 RESULTS, CPU UTILIZATION

E.Test Case 5

In Test case 5, we needed to put Host 3 in upkeep mode. In this situation all VMs on Host 3 ought to be relocated to different hosts. For the situation with high load (700 req/s), after we have turned on the DRS and set the movement limit to level 1 (preservationist), it begun to move two VMs from Host 3 to Have 1 and two VMs from Host 3 to Host 2. DRS moved the same VMs to similar hosts for different levels of forcefulness (level 3 and level 5) (see Table IX). For the case with medium stack (500 req/s), DRS relocated again the same VMs to the same hosts for every one of the three different levels of forcefulness (see Table IX). In both of these cases we can state that DRS did a great job and settled on an indistinguishable choice from a human expert. As found in Table X, after DRS relocation the CPU use on both Host 1 and turned into the same. In any case, when the heap was low (300 req/s), after we have turned on the DRS and set the movement edge to level 1 (preservationist) and level 3 (direct), DRS relocated three VMs from Host 3 to Host 2 and one VM from Host 3 to Have 1 (see Table IX). Table X demonstrates that after DRS movement the CPU usage on Host 2 is 47% and the CPU use on Host 1 is 38%. This implies DRS proved unable circulate the heap between the hosts similarly and DRS movement was not effective for this situation. Nonetheless, when the movement edge was set to both after DRS movement and after human master relocation progressed toward becoming around 40%. In spite of the fact that DRS did not work in two of the cases, we see that in alternate, DRS could make great choices.

Load (req/s)	**Level of Aggressiveness**			**Human Expert**
VM	**Level 1**	**Level 3**	**Level 5**	
700	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H1 VM 5-1 from H3-H2	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H1 VM 5-1 from H3-H2	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H1 VM 5-1 from H3-H2	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H1 VM 5-1 from H3-H2
500	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H1 VM 5-1 from H3-H2	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H1 VM 5-1 from H3-H2	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H1 VM 5-1 from H3-H2	
300	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H2 VM 5-1 from H3-H2	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H2 VM 5-1 from H3-H2	VM 3-2 from H3-H2 VM 6-2 from H3-H1 VM 2-1 from H3-H1 VM 5-1 from H3-H2	

TABLE IX. SUMMARY OF THE RESULTS OF THE TEST CASE 5

	CPU Utilization (%)									
	a. Low Load (300req/s)				b. Medium Load (500req/s)			c. High Load (700req/s)		
	Same Load	Load Shift	DRS Migration	Human Expert	Same Load	DRS Migration	Human Expert	Same Load	DRS Migration	Human Expert
H1	28	38	42	42	40	63	63	52	85	85
H2	28	47	38	38	40	63	63	51	85	85
H3	27	4	4	4	39	4	4	50	4	4

TABLE X. TEST CASE 5 RESULTS, CPU UTILIZATION

IV. CONCLUSION

The main aim of this study is to compare the VMware's DRS migration with human expert migration. Here a number of test scenarios is considered. Test scenario is designed in such a way that there is always a good solution for balancing the load on VMs. Here 5 test cases are considered for 3 different loads on each VM. DRS with 3 levels of aggressiveness i.e. conservative (level 1), moderate (default) (level 3) and aggressive (level 5) is tested. Here in total 45 test cases is performed. Out of 45 test cases performed, VMware DRS did not migrate in 23 of the test cases. In 11 test cases VMware

DRS made decision same as that of human expert. In 4 test cases it made unnecessary migrations. i.e. load suffered from "ping-pong" effect from one VM to other. Results show that tests that were conducted by using VMware DRS could balance the load close to that of Human expert migration decision. In few cases it suffered from "ping-pong" effect that can be overcome by designing the robust VMware DRS.

REFERENCES

[1] C. Clark, et al., "Live migration of virtual machines," 2nd symposium on Networked Systems Design and Implementation, 2005.

[2] S. Sotiriadis, N. Bessis, P. Gepner, N. Markatos, "Analysis of Requirements for virtual machine migrations in dynamic clouds," 12th IEEE International conference on Parallel and Distributed Computing, 2013.

[3] C. A. Waldspurger, "Memory resource management in vmware esx server," SIGOPS Oper.Sys.Rev., vol36, 2002.

[4] H. Nagamani and P. Jayarekha, "Load balancing with optimal cost scheduling algorithm," International conference on Computation of Power, Energy, Information and Communication, 2014.

[5] E. Arzuaga, and D. Kaeli, "Quantifying load imbalance on virtualized enterprise servers," proceedings of the first international conference on Performance engineering, 2010.

[6] VMware Inc. Resource Management with VMware DRS.http://www.vmware.com/pdf/vmware_drs_wp.pdf

[7] VMware Inc. DRS Performance and Best Practices. https://www.vmware.com/files/pdf/drs_performance_best_practices_wp.pdf

[8] K. Lazri, S. Laniepce, J. Ben-Othman, "When Dynamic Vm Migration Falls Under the Control of VM Users," IEEE international conference on Cloud Computing Technology and Science, 2013.

[9] Lei Lu, et al., "Application-driven dynamic vertical scaling of virtual machines in resource pools," IEEE Network Operations and Management, 2014.

[10] H. W. Choi, H. Kwak, A. Sohn, K. Chung, "Autonomous learning for efficient resource utilization of dynamic vm migration," 22nd annual international conference on Supercomputing, 2008

[11] J. Park, J. Kim, H. Choi, Y. Woo, "Virtual machine migration in self-managing virtualized server environments," 11th International conference on Advanced Communication Technology, 2009.

[12] Kochut, K. Beaty, "On strategies for dynamic resource management in virtualized server environments," 15th International symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2007.

[13] G. Khanna, K. Beaty, G. Kar, A. Kochut, "Application performance management in virtualized server environments," 10th IEEE conference on Network Operations and Management, 2006.

[14] F. Hermenier, et al. "Entropy: a Consolidation Manager for Clusters," Proceedings of the ACM international conference on virtual execution environments, 2009.

[15] T. Wood, P. Shenoy, A. Venkataramani, M. Yousif, "Black-box and Gray-box Strategies for Virtual Machine Migration," Proceedings of the 4th USENIX conference on Networked systems design and implementation, 2007.

[16] L. Xu, W. Chen, Z. Wang, Sh. Yang, "Smart-DRS:Astrategy of Dynamic Resource Schedulingin Cloud Data Center," Proceedings of the IEEE international conference on Cluster Computing Workshops, 2012.

[17] S. Shirinbab, L. Lundberg, D. Erman,"Performance evaluation of distributed storage systems for cloud computing," Published in International Journal of Computers and their applications, 2013.