Decision support for virtual machine reassignments in enterprise data centers

Thomas Setzer and Alexander Stage
Technische Universität München (TUM)
Department of Informatics (I18)
Boltzmannstr. 3, 85748 Garching, Germany
{setzer,stage}@in.tum.de

***Abstract*—We introduce a novel method to discover beneficial time frames for adapting virtual machine (VM) assignments in consolidated enterprise data centers. Our key insight lies in learning an optimal orthonormal transform from the workload data of a set of enterprise applications hosted in VMs. The transform allows us to extract only a few indicators from long, time-varying and complex workload time series. The indicators represent the initially high-dimensional data set in a reduced form which allows for a concise identification of periods of relatively stable resource demands and turning points in the behavior of a set of VM workloads that require VM reassignments. In this work, we address one of the most pressing problems for data center operators, namely the reduction of managerial complexity of resource and workload management in data centers hosting thousands of applications with complex and varying workload behaviors. We demonstrate the decision support model using workload traces from a professional data center.** [1]

I. Introduction and Motivation

In today's data centers, typically thousands of enterprise applications with time varying and complex resource demand behaviors are concurrently hosted. Virtualization based server consolidation is increasingly used to run multiple applications plus underlying operation systems in virtual machines (VM), hosted jointly on physical servers, sharing and multiplexing server resources over time. Virtualization technology provides for required performance and security isolation amongst col-located VMs.

Energy consumption is one of the key cost drivers in data centers, and already accounts for up to 50% of the total operating costs [1]. As energy consumption increases almost linearly with the amount of running servers, IT service managers basically need to decide which VMs should be assigned to which physical servers to raise average server utilization in order to reduce the total number of required, energy consuming physical servers while avoiding overload situations. In order to benefit from multiplexing the resources of a single server in an efficient way, complementary workloads need to be combined on a server subject to the constraint that the aggregated resource demands of the VMs assigned to a server must not exceed its capacity at any point in time within a given planning period.

Besides the technical means for server consolidation virtualization allows for agile adaptation of virtual machines capacities and server assignments by live migration. To cope with daily, weekly, or short term workload fluctuations, shifts, trends and uncertainties that you typically find in enterprise application workload [2] [3]) dynamic VM reassignment methods have been proposed recently [4], [5], [6] to avoid potential server underutilization or overload.

Short, predetermined planning cycles or reactive control techniques for VM placement decisions are typically applied by these approaches. However, frequent alignment of VM assignments (re-consolidation) in order to further reduce the required overall capacity as done for example in [4] leads to higher control action, planning, and migration overheads. Control actions such as VM live migrations [7] incur significant overheads on the source and destination servers as well as on the network infrastructure and have negative effects on the responsiveness of the applications running in VMs. Furthermore, short planning cycles also imply the usage of heuristics instead of exact methods for repeatedly determining VM to server assignments.

Longer planning periods such as weeks and months [8] on the other hand are more sensible towards changes in the resource demand of VMs. However, longer planning periods lead to less control overhead and allow for exact optimization methods for VM assignments that potentially lead to higher efficiency.

Consequently, accurate methods for determining time frames for VM reassignments that help to minimize costly migrations while maximizing the possible benefits of re-consolidation is an important aspects in dynamic infrastructure management. Existing work as for example presented in [9], [10] proposes rather arbitrarily defined, static control intervals of several hours without analyzing current and expected demand patterns of the set of managed workloads.

In this paper we present a novel method for defining variable time intervals for reassigning VMs based on the identification of significant changes in the behavior of the overall set of workloads. First, we project workload data into a low dimensional space using multivariate statistics to reduce

[1]Acknowledgement: This work is supported by Siemens IT Solutions and Services (SIS). As one of the largest European IT service providers, SIS provisioned the data for our numerical experiments as well as domain knowledge and various valuable insights.

978-1-4244-6039-7/10/$26.00

the dimensionality of the data while preserving its relevant structure as far as possible. We interpret the coordinates of the workload time-series projections as indicators for dominant relationships between workload levels of different VMs (principal workload mixes) as well as overall workload levels. This approach allows for determining significant turning points in workload height and/or workload composition where re-migrations are recommended.

The remainder of this paper is structured as follows. In section II we review VM consolidation and dynamic resource management with regard to cost-savings and associated overheads. In section III we review Singular Value Decomposition (SVD) and show how it can be applied to extract indicators that allow us to detect turning points in the behavior of a set of workloads. Subsequently we demonstrate the applicability of our approach using VM workload data from a professional data center. Related work is discussed in section IV. In section V conclusions are drawn and future work is discussed.

II. Server Consolidation

In this section we review different techniques for static server consolidation and associated cost-savings. Subsequently we present a case study that demonstates the benefits of dynamic workload management by using differentiated VM consolidations for day- and nighttimes, and show that the migrations required for dynamic workload management entail significant overheads that need to be taken into account to avoid overload situations and corresponding performance problems.

A. Static Peak Workload Based Consolidation

The major goal of server consolidation is to reduce the number of physical servers in a data center. However, it needs to be ensured that the aggregated workloads of VMs assigned to a server must never exceed a server's capacity regarding server resources such as CPU or memory in order to avoid performance problems and associated SLA violations.

First Fit / First Fit Decreasing Heuristics are very commonly used for server consolidation by IT service organizations, as they provide a simple and relatively efficient technique [11] for server consolidation, as shown in Figure 1. For each VM, resource demand measurements are collected over time that can be used to compute the maximum (or a certain percentile, e.g. 99%) of CPU usage, memory requirements, disk I/O and network bandwidth usage over a period of time (e.g., several months) as described in [2]. Then the first VM is mapped to the first server. The second VM is mapped to the same server if it can accommodate the resource requirements. Otherwise, a new server is introduced and the VM is mapped to this new server. This step is repeated until each of the VMs has been mapped to a server. The set of resulting hosts then comprises the consolidated server farm. When applying a first fit decreasing heuristic, the VMs are first ordered by there capacity usage level (e.g. according to their CPU-usage if CPU is assumed to be the primary bottleneck resource or a weighted value derived by an ranking of multiple server

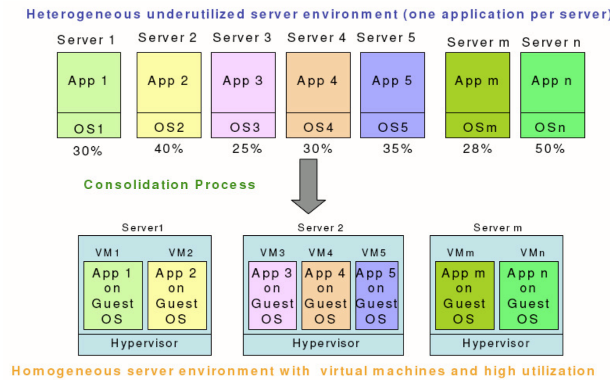

Fig. 1. Typical Server Consolidation Scenario. Source: [5]

resources). While this heuristic is simple and can be computed in $O(n \ log \ n)$, where n denotes the amount of VMs to be consolidated, in general it yields a sub-optimal mapping of VMs to servers. For example, if VMs with high CPU-usage are assigned to the first server, and VMs with low CPU usage but high memory usage are assigned to the second server, lots of resources are unused on both servers.

Obviously this problem can be solved with better solution quality when modelling it as the well-known *static bin packing problem*, with the capacity limits of servers as bin size, and the resource demand level of a VMs as an item's size. However, solving the bin-packing problem exactly is proven to be NP-hard and cannot be solved for larger data centers [12].

B. Improved Consolidation based on Workload Complementarities and Time Multiplexing

Usually, workloads show recurring patterns on a daily or weekly basis. For example, payroll accounting is performed at the end of the week, while workload of an OLAP application has a daily peak in the morning when managers access their reports. More advanced consolidation models leverage these cycles by first determining representative e.g. daily VM workload profiles (component days) describing the workloads expected in each time-slot (e.g. maximum over a five-minute interval) for different resource types such as CPU and memory over a planning period such as a day. Consecutively an integer program attempts to assign those VMs together on servers whose workloads are complementary, i.e. expose peak demands at different times of the day to smoothen and further increase overall server utilization which leads to a reduction of the total number of required servers. One constraint per resource and time-slot ensures that the aggregated workload of VMs assigned to a server must not exceed the servers capacity within a planning period.

In [13] it is demonstrated that in larger data centers, approximately 30-35% of servers can be saved by exploiting complementarities instead of using the peak-load oriented bin-packing model. However, as the problem is strongly NP-hard, it cannot be solved up to optimality for larger instances, in particular as the number of constraints grows with the time granularity multiplied by the number of server resources under consideration. Therefore, usually time-slots are coarsened to reduce the number of constraints, e.g., hourly workload intervals are used by taking maxima over 12 consecutive five-minute intervals in order to reduce the problem size. Unfortunately, coarsening intervals reduces the ability to exploit workload complementarities and therefore impacts the solution quality by requiring more servers compared to using smaller time-slots.

C. Additional Savings through Time-Varying Consolidation

In static server consolidation, VM to server assignments are calculated for extended periods of time such as weeks and months. This approach lead to improvements in average server utilization but still leads to temporary servers underutilization during e.g. night hours due to typical resource demand patterns of enterprise applications (as workload is typically lower during night-hours).

More dynamic approaches that reassign VM for periods of underutilization promise to further reduce operational costs. We demonstrate the potential for additional cost savings, that can be achieved by adapting VM assignments by the following consolidation case study. In this study, we apply the decision model for static server consolidation [13] exploiting workload complementarities to determine optimal VM assignments for day (6 a.m. to 10 p.m.) and night (10 p.m. to 6 a.m.) shifts for each day of the week for a set of 500 real world workload traces provided by our industry partner. This procedure results in a series of 14 individual assignments for the seven days of the week. Solving the resulting 14 consolidation problems using exact optimization methods leads to an upper bound to the savings achievable within a resource-constrained setting for VM migrations. In figure 2 we summarize the results of the

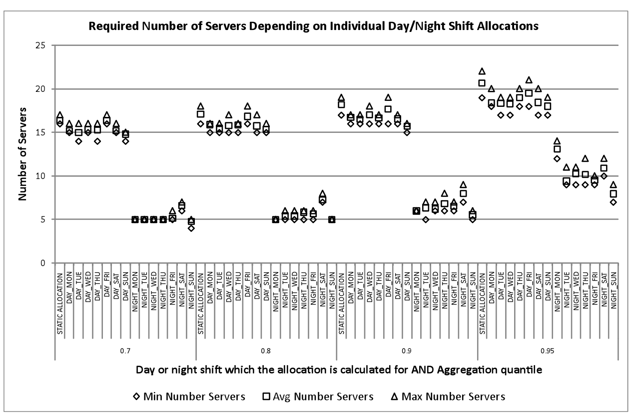

Fig. 2. Required physical servers for day and night shifts

consolidation study for various levels of buffer capacities to account for uncertainties in workload prediction (i.e., overload-avoidance even in the case when real workload exceeds the forecast). We show the minimum, maximum and average amount of servers for each of the 14 time periods over a set of 20 consolidation scenarios. Each scenario consisted of 500 workload traces. A scenario was derived by sampling workload traces from the set of 500 original traces with replacement.

In summary it is possible to save 58 % on average over all scenarios of servers during the night shifts by applying day and night shift assignments at each day of the week, which results about 32 % savings of operational hours of servers for an assumed planning period of three months. The significant potential for cost savings emphasizes the need to determine suitable time frames for VM reassignments whenever significant changes in the overall set of workloads can be detected and reliably forecasted based on historical workload behavior.

However, the benefits of migrations must outweight their costs in terms of overhead, which are significant as illustrated in the next subsection.

D. Migration Overhead

In order to quantify live migration overheads we conducted some experiments using different server benchmarks. Here we present the result of experiments with the Support, E-commerce and Banking workloads of the SPECweb2005 [14] web server benchmark and virtual machines running on the Xen 3.3.2 hypervisor. We instrumented this benchmark since web servers are the first tier of modern multi-tier architecture transaction processing enterprise applications. In all scenarios we run the web server application in a VM with a fixed resource entitlement of one core (of a four core Intel Xeon 2.66 GHz processor) and 2 GB of main memory. Especially the amount of memory has been chosen rather low in comparison to actual memory requirements of application or database servers. In figure 3 we report the CPU overhead for live migrations of VMs running the E-commerce workload for six scenarios that exercised the web server application with different amounts of concurrently running virtual users (ranging from 100 to 600). We executed these experiments multiple times and report average values (the standard deviation was found to be very low).

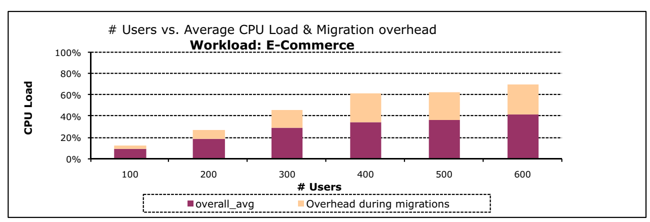

Fig. 3. CPU overhead for VM live migrations

We observe that with increasing service demand caused by larger numbers of concurrent virtual users, the additional CPU cycles required during a migration increase from about 20% to almost 80% of the actual CPU demand of a VM. The overhead is mainly due to additional I/O operations for main memory transfer from the source to the target servers and the shadow page table management of the hypervisor.

Figure 4 shows the time required to finish a migration for the respective runs, which ranges between 20 seconds for a lightly loaded VM with 100 concurrent virtual users up to 40 seconds for heavily loaded VMs. Besides the additional CPU demand, we observed an extensive utilization of network bandwidth for all migration scenarios as all migrations saturated the interconnecting 1 Gbit network link for the time of a migration almost completely.

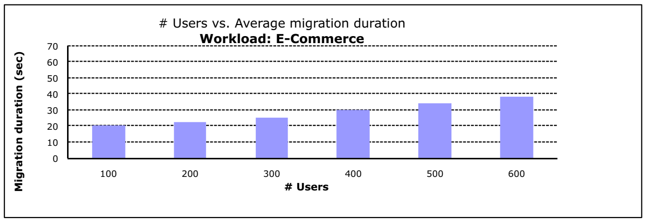

Fig. 4. Time required for VM live migrations

During our research efforts, we found comparable overheads for a variety of other application performance benchmarks.

Hence, VM reassignments, entailing costly migrations and high computational costs to derive new VM assignments, should be done only when the workload behavior changes significantly in terms of its level and/or mix so that the benefits overweight their associated costs. In the remainder of this paper we propose a decision support model that is based on a few indicators that allow us to identify points in time where re-consolidation is advisable as the overall workload structure of a set of VMs (rather than observing single VMs) changes significantly.

III. Dimensionality Reduction of Workload Data

Let u_{jt} describe the CPU requirements of a VM j in timeslot t, $t \in 1, \ldots, \tau$, over a component period such as a day of the week, a week or any other periodicity of a workload time series. The τ-dimensional tuples describing a VM's workload time series can be represented as points in a τ-dimensional space, where a VM workload level in t is indicated as a coordinate along a dimension (t). To reduce dimensionality, these points need to be projected into an E-dimensional space so that $E << \tau$. We apply truncated singular value decomposition (tSVD) for that purpose as it is applicable to non-square and not full-ranked workload matrices.

Let R be the original $|J|$ by τ matrix of $|J|$ VMs (J denotes the set of VMs), with time series of length τ as row-vectors (elements of R are u_{jt}). Let $U\ \Sigma\ V^T$ be R's factorization using standard SVD, where R's singular values σ_e in Σ are ordered in non-increasing fashion, U contains the left singular vectors, and V^T contains the right singular vectors [15].

The intuition of this factorization is that the left singular vectors in U are the axis of a new space (forming R's row space), the associated singular values are scaling factors for these axis, and the right singular-vectors in V^T represent the coordinates of timeslots in the new space. One can think of the left singular vectors as major proportions of VM workload, or in other words the typical mix of the workload amongst VMs over all timeslots. As an illustration, consider a simple scenario with only two VMs A and B as shown in Figure 5. The data points indicate the workloads levels of both VMs in different time-slots (point labels). The orientation of the first singular vector u_1 points in a direction closer to A's axis, which means that A's workload is typically higher than B's workload.

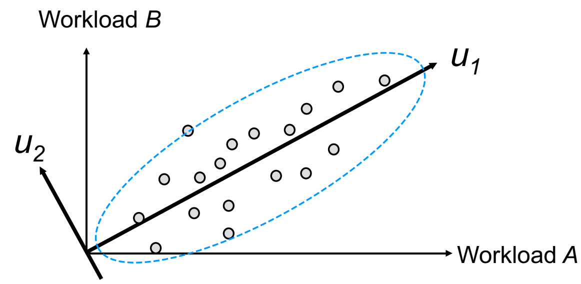

Fig. 5. Time Slot Projection into the VM Workload Space

For each t, coordinates v_1^t are calculated by perpendicular projection of the points onto u_1, the first left singular vector. These coordinates show the best 1-dimensional approximation of the data because u_1 captures as much of data variation between timeslots as possible by one direction. Time coordinates v_2^t regarding the second left singular vector u_2 (note that u_2 is orthogonal to u_1) capture the maximum variance after removing the projection of the data along u_1, i.e., the second most dominant workload mix between VMs over all timeslots (in this 2-dimensional example, u_2 captures all of the remaining variance; in general the number of singular vectors equals Rs rank).

A. Decision Support for Detecting Workload Changes

The practical appeal of the tSVD is that variation below a particular threshold E can be ignored as the singular values associated with the left singular vectors sort them in goodness order with explained variation from most to least. This is the main idea of tSVD where only the first E column vectors of U and the first E row vectors of V^T are considered.

The first singular vectors in U describe the most relevant workload mixes (proportions) between VMs over all time-slots. The singular values in Σ associated with these left singuar values indicate the relevance (the scale) of a certain workload mix for describing the behavior of the set of VM workloads from a global point of view. As the right singular vectors contain coordinates of time-slots t along the new axis, one can directly derive the relevance of a particular workload mix is in a given time-slot.

Decision support for determining turning points in the behavior of a set of workloads that indicate the beginning of a new period of relative stability of resource demands is given by analysing the positions of coordinates of subsequent time-slots along the first axis. The more the coordinates along the first or the second axis (or both) change from one time-slot to another, the more advisable it is to consider VM re-assignmemts. If for example, the coordinates along the first axis decrease significantly, the server park can most likely be shrunk. If the coordinates along the first axis decrease, but increases along the second axis, it might be required to migrate a VM from one server to another as the workload mix changes and VMs that share a large portion of the second workload mix might cause an overload situation.

In the next subsection we demonstrate the decision support tool using industry data. For reasons of brevity, we will now focus on coordinates along the first two new axis only, as typically over 80% of the total variance in workload time series can be captured by coordinates along the first two new axis only. Please note that we could also use and visualize coordinates along the first three axis in order to increase the accuracy of our approach. In our experiments the first three axis covered about 90% of the total workload variance which is found to be a reasonable degree of description detail allowing for detecting significant changes in a workload mix.

B. Demonstration

From a professional data center we obtained data describing one-hour maxima for CPU demand of hundreds of VMs over four months. Most of these workloads exhibit rather deterministic daily patterns without a significant trend. Demand anomalies could be observed seldom, but exist as an inspection revealed. According to our industry partner, CPU is the primary bottleneck resource for the applications under study. Thus, we consider daily CPU workload profiles.

In our experiments, we utilized four months of data for each workload to derive component working day profiles (estimated maximum workload levels per time interval). The resulting time series profiles of this preprocessing step are illustrated in figure 6. Furthermore, we assume physical servers to be identical with unit capacity. To illustrate the decision support tool introduced in the last section we describe a small scenario of 16 VMs to be consolidated in detail.

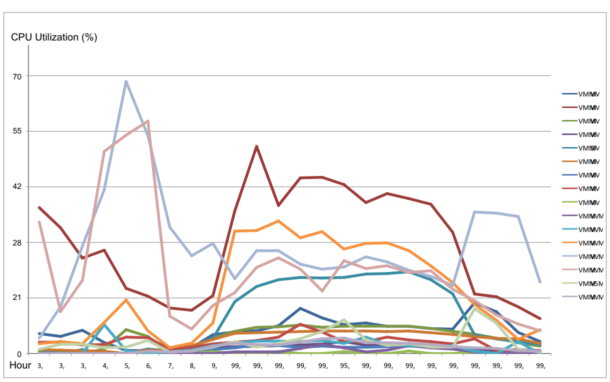

Fig. 6. VM Daily Workload Profiles

The scatterplot of the u_1 - u_2 coordinates of all 24 timeslots are shown in figure 7. By examining the plot, we identified five distinct periods of time for which the coordinates of their constituent, continuously sequenced time-slots are in close distance to each other: period 1 {hour 1-3}, period 2 {hour 4-6}, period 3 {hour 7-9}, period 4 {hour 10-20}, and period 5 {hour 21-24}. The five periods, or time clusters expose a significant inter-cluster distance in euclidean space. We claim that VM migrations are required between the clusters. We now analyze the real workload behavior of the set of VMs to validate the mentioned claim.

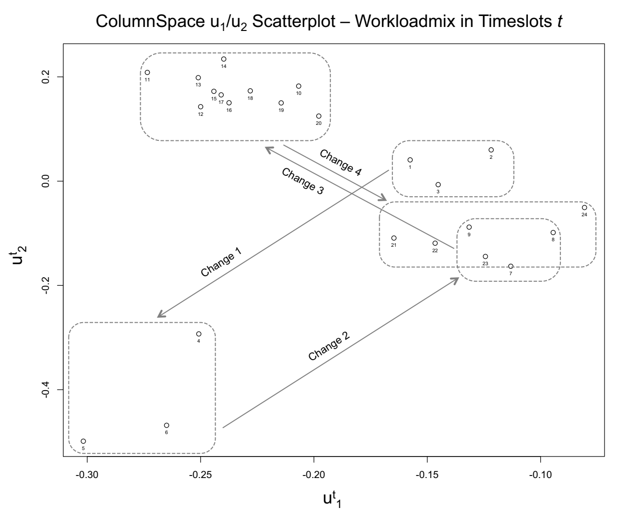

Fig. 7. Timeslot Coordinates in the $\{u_1, u_2\}$ Space

The table in figure 8 shows the aggregated workload of all VMs in each time-slot (hour). In period 1, the total workload never exceeds 0.87 and one server suffices to handle all VM workloads. In period 2, the highest absolute coordinates along u_1 and u_2 can be found. Here, the total workload level is approximately 1.5 (or 150%), and a second server is required. In period 3, the total workload decreases again below 1, thus, from 7 to 9 a.m. one server suffices and the second server can be shut-down. Then, in period 4 from 10 a.m. to 8 p.m. the total workload level is again between 100% and 200%, and two host servers are required to handle the workload. In Period 5, again one server suffices as the aggregated workload drops below 1.

Hour	1	2	3	4	5	6	7	8
Load	0.87	0.67	0.83	1.41	1.65	1.47	0.58	0.54
Hour	9	10	11	12	13	14	15	16
Load	0.88	1.43	1.78	1.73	1.80	1.75	1.76	1.67
Hour	17	18	19	20	21	22	23	24
Load	1.67	1.62	1.48	1.23	0.96	0.93	0.77	0.50

Fig. 8. Aggregated Workload in a Timeslot

Please note that although in period 2 and in period 4 a second server is required, the different location of both periods (clusters) in the scatterplot indicate that the workload mix in both periods differs significantly so that optimal VM assignments should differ in both periods. Indeed, if we take VM 13 and VM14 with their strong peaks in period 2 into consideration, both exceeding 50% usage level, both servers must be assigned to different servers in order to avoid overload. In period 4 the two VMs can be hosted on the same server without exceeding a single server's capacity. The other way arround, hosting together VM 2, VM 5, and VM 12 would lead to overload in period 4, but not in period 2.

We conclude that all identified periods require VM reassignments to minimize the total operational hours of servers and to avoid overload situations. Existing distance-based clustering methods [16] can be applied to automatically discover stable workload periods considering that cluster members forming an uninterrupted sequence of time-slots, finding the optimal balance of inter- and intra-cluster distances.

IV. Related Work

While there has been a lot of work on capacity planning and management for multi-tier enterprise applications, little work has focused on efficient and especially overhead aware dynamic resource allocation in virtualized data centers.

Closest in spirit to our work is the work by Bichler et al. [2] and by Rolia et al. [17], both use integer programs to exploit workload complementarities and statistically multiplex resources over time to minimize the amount of targets while ensuring sufficient capacity in each time interval. They apply approximations such as time-slot coarsening and meta-heuristics such as Genetic Algorithms (GA) to make their solutions computationally tractable but do not consider VM migration overheads.

Rolia et al. [18] and Cherkasova et al. [19] describe an approach based on statistical multiplexing using GA that penalize low target utilizations and target overload to minimize the number of target servers. Seltzam et al. [20] propose forecast workload profiles to multiplex server resources and utilizes reactive, fuzzy logic based rules to initial VM migrations. While this approach seems to be reasonable, it exposes the risk of system instabilities and does not asses the gains of a VM migration against its overheads. In fact, the time required for a VM migration is even elongated in an overload situation and increases the resource contention on a server. Urgaonkar et al. analyse best-fit and worst-fit heuristics to bundle complementary services on common servers [21] .

Khanna et al. propose a VM migration decision method based on SLA-observation to track the performance of the key metrics and to move virtual machines if for example response-time exceed a critical threshold such that the SLA is restored, while trying to minimize the number of required host servers [5].

Zhu et al. [9] and Gmach et al. [4] present similar approaches to resource allocation in virtualized data centers. Both papers present a hierarchy of controllers that operate at different time scales. Reactive rules for discovering resource shortages or underutilization are utilized to adjust resource allocations and heuristics are used to take virtual machine placement decisions at arbitrarily defined time intervals. While both approaches take overheads for migrations into account, without determining or quantifying them exactly, they do not aim at minimizing the overhead-related overheads or determining beneficial time frames for virtual machine reassignments.

Fukunaga [22] addresses the issue of minimizing the amount of migrations required to transfer a data center from one VM assignment to another by modeling it as a constraint satisfaction problem with optimization goals and propose several search algorithms. While the algorithmic approach is an interesting problem of its own, Fukunaga does not address the issue of determining when to trigger reassignments, nor is the approach overhead-aware. Instead of minimizing the overhead weighted amount of migrations, the amount of migrations is minimized which may lead to costly reassignments.

V. Summary and Outlook

We introduced a method to discover beneficial time frames for adapting VM assignments based on only two extracted indicators per time interval of a set of complex VM workload behaviors. These indicators propose points in time when reassignments of VMs are recommended. Using these indicators, principal workload behavior in time-slots can be easily analyzed either be automateted clustering mechanisms or by simple inspection of a resulting visualization such as a scatterplot.

The method aims at the reduction of managerial complexity of resource and workload management in data centers hosting thousands of applications with varying workload behaviors. Although complexity is a major issue for practioneers as well as for scientists, to the best of our knowledge there is no previous work on determining points in time where migrations are likely to be beneficial for a given set of workloads. Since our approach is computational efficient and tractable through the existence of proven approximation algorithms for the tSVD computation, scalability is not expected to become an issue for extending our approach to large problem sizes containing several hundreds of workload traces.

In our future research we plan to evaluate larger sets of workload traces and we will automate the process of identifying periods of stable demand. Furthermore, we plan to extend our approach by taking into account migration-related CPU and network overheads to allow for dynamic resource management in a controlled and service level agreements aware fashion.

References

[1] D. Filani, J. He, S. Gao, M. Rajappa, A. Kumar, P. Shah, and R. Nagappan, "Dynamic data center power management: Trends, issues, and solutions," *Intel Technology Journal*, February 2008.

[2] M. Bichler, T. Setzer, and B. Speitkamp, "Provisioning of resources in a data processing system for services requested," 2006.

[3] D. Gmach, S. Krompass, A. Scholz, M. Wimmer, and A. Kemper, "Adaptive quality of service management for enterprise services," *ACM Transactions on the Web (TWEB)*, vol. 2, 2008.

[4] D. Gmach, J. Rolia, L. Cherkasova, and A. Kemper, "Resource pool management: Reactive versus proactive or let's be friends," *Comput. Netw.*, vol. 53, no. 17, pp. 2905–2922, 2009.

[5] G. Khanna, K. Beaty, G. Kar, and A. Kochut, "Application performance management in virtualized server environments," in *10th IEEE/IFIP Network Operations and Management Symposium (NOMS)*. Vancouver, BC, Ca: IEEE Computer Society, April 2006, pp. 373 – 381.

[6] T. Wood, P. Shenoy, A. Venkataramani, and M. Yousif, "Sandpiper: Black-box and gray-box resource management for virtual machines," *Computer Networks*, vol. 53, no. 17, pp. 2923 – 2938, 2009, virtualized Data Centers. [Online]. Available: http://www.sciencedirect.com/science/article/B6VRG-4WPTXWG-1/2/e52906805acc139e082e8b3f9ccedb5e

[7] C. Clark, K. Fraser, S. H, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in *In Proceedings of the 2nd ACM/USENIX Symposium on Networked Systems Design and Implementation (NSDI*, 2005, pp. 273–286.

[8] J. Rolia, L. Cherkasova, M. Arlitt, and A. Andrzejak, "A capacity management service for resource pools," in *WOSP '05: Proceedings of the 5th international workshop on Software and performance*. New York, NY, USA: ACM, 2005, pp. 229–237.

[9] X. Zhu, D. Young, B. Watson, Z. Wang, J. Rolia, S. Singhal, B. Mckee, C. Hyser, D. Gmach, R. Gardner, T. Christian, and L. Cherkasova, "1000 islands: an integrated approach to resource management for virtualized data centers," *Cluster Computing*, 2009. [Online]. Available: http://dx.doi.org/10.1007/s10586-008-0067-6

[10] D. Gmach, J. Rolia, L. Cherkasova, G. Belrose, T. Turicchi, and A. Kemper, "An integrated approach to resource pool management: Policies, efficiency and quality metrics," in *Dependable Systems and Networks With FTCS and DCC, 2008. DSN 2008. IEEE International Conference on*, June 2008, pp. 326–335.

[11] M. Garey and D. Johnson, "A 71/60 theorem for bin packing," *Journal of Complexity*, no. 1, pp. 65–106, February 1985.

[12] A. Abdelbar and S. Hedetniemi, "Approximating maps for belief networks in np-hard and other theorems," *Artificial Intelligence*, vol. 102, pp. 21–38, 1998.

[13] M. Bichler, T. Setzer, and B. Speitkamp, "Capacity management for virtualized servers," in *Workshop on Information Technologies and Systems (WITS)*, Milwaukee, USA, 2006.

[14] Standard Performance Evaluation Corporation. (2005) Specweb2005 web server benchmark, version 1.20. [Online]. Available: http://www.spec.org/web2005/

[15] G. H. GOLUB and C. F. VAN LOAN, *Matrix Computations*. The Johns Hopkins University Press, 1990.

[16] P. Berkhin, "A survey of clustering data mining techniques," *Grouping Multidimensional Data*, pp. 25–71, 2006.

[17] J. Rolia, A. Andrzejak, and M. Arlitt, "Automating enterprise application placement in resource utilities," in *Workshop on Distributed Systems*, Heidelberg, Germany, 2003.

[18] J. Rolia, L. Cherkasova, and M. Arlitt, "A capacity management service for resource pools," in *Workshop on Software and Performance*, Palma de Mallorca, Spain, 2005.

[19] L. Cherkasova and J. Rolia, "R-opus: A composite framework for application performability and qos in shared resource pools," Tech. Rep., 2006.

[20] S. Seltzam, D. Gmach, S. Krompass, and A. Kemper, "Autoglobe: An automatic administration concept for service-oriented database applications," in *Int. Conference on Data Engineering (ICDE 2006)*, Atlanta, USA, 2006.

[21] B. Urgaonkar, P. Shenoy, A. Chandra, P. Goyal, and T. Wood, "Agile dynamic provisioning of multi-tier internet applications," *ACM Trans. Autonom. Adapt. Syst.*, vol. 3, no. 1, p. 39, 2008.

[22] A. S. Fukunaga, "Search spaces for min-perturbation repair," in *15th Int. Conf. on Principles and Practice of Constraint Programming*, 2009, pp. 383–390.