On Impact of Dynamic Virtual Machine Reallocation on Data Center Efficiency

Andrzej Kochut
IBM T.J. Watson Research Center
19 Skyline Drive
Hawthorne, NY, 10532, USA
akochut@us.ibm.com

Abstract

Modern OS virtualization technology allows for "live migration" of virtual servers between physical hosts after the initial consolidation thus providing new avenues for performance optimization. This paper focuses on quantifying by how much data center efficiency can be improved using dynamic (i.e., in response to demand changes) reallocation of virtual machines between physical servers. An analytical performance model of dynamic virtual machine reallocation is presented. It allows to estimate the reduction (as compared to the static consolidation) in the number of physical servers required to host the workload (under the same requirements of the maximum number of capacity overloads). Model is validated using simulations and several insights based on the analysis are provided.

1. Introduction

Traditional data centers consist of a large number of physical machines, each executing a single instance of an operating system. For example, a certain group of servers supports email function and executes Linux with an appropriate set of programs for email handling. Another group of servers may run Windows providing access for remote users to office applications. Installing new application usually means purchasing a new physical server and installing a new instance of an operating system and the application. Capacity of the servers needs to be planned to handle peak loads of applications resulting in a relatively low utilization leading to wasteful power consumption and maintenance costs.

A popular way of addressing this problem is server consolidation. It is an optimization approach that leverages technique called operating system virtualization to reduce the number of physical servers in a data center. Virtualization was first proposed by IBM in the 1960's [9, 13, 15] but with rapid increases in computing capacity of the low-end machines, similar capabilities are now available for both x86 and RISC processing platforms [8, 32, 34]. A good background on the current state of virtualization technology can be found in [26]. Typically, a thin software or firmware layer called a hypervisor executes on a physical machine (PM) and presents an abstraction of the underlying hardware to host multiple virtual machines (VMs).

Server consolidation increases average server utilization and thus decreases running costs of the data center. Instead of having a large number of highly over-provisioned servers, consolidated data center consists of a smaller number of servers each executing many operating system instances. Our prior work [3, 20] as well as of other authors [18, 19] has concluded that after the initial static consolidation of multiple operating systems onto smaller number of physical machines, further optimization can be achieved by dynamic reallocation of VMs based on the workload changes. The main reason for this is that even after the consolidation the resulting aggregate workload on each of the PMs has still significant variation on a longer timescales (such as a day). This optimization can be facilitated by a process called "live migration" [6, 22]. In live migration an executing VM can be moved from one physical host to another with only very short (on the order of 100ms [6]) interruption in the execution of the operating system contained in the VM. Nevertheless, the live migration process requires devoting part of the computing power to the transfer of the VM state between two physical machines while still maintaining the execution of a contained OS. Both source and destination server need to devote CPU power to network transfer as well as network bandwidth is consumed by transfer of large chunks of data representing memory state of VM. Moreover, even though the period when the migrating VM is not executing its OS is very brief, the total migration time is significant (typically on the order of minutes). Those costs make the migration decision quite important from the system performance perspective. In our prior work [20] we have proposed an analytical model of the

978-1-4244-2818-2/08/$25.00 ©2008 IEEE

single migration process that allows to quantify potential gain from a migration decision. Moreover, there has been significant amount of research [10, 12, 16, 23, 29] on utility of process migration and, in particular, how the expected process lifetime can be used to perform optimal decisions.

This paper focuses on the following question: to what extent the utilization of a consolidated data center can be increased by employing dynamic reallocation of VMs? As discussed above since the reallocation has inherent costs and degrades the overall performance of the system it can not be performed too often. Thus the amount of gain that can be realized from reallocating the VMs is not easily quantifiable. As a simple example consider the case of resource demand for which most of the variability is contained in frequencies higher than the VM migration frequency. In this case moving does not make sense because the system will not be able to keep up with the changing load. However, if the majority of signal's power is located at lower frequencies then dynamic migration can be a viable option potentially reducing the number of required physical servers.

We present an analytical model of the virtualized data center and use it to derive relationship between important parameters that influence the amount of gain expected from the dynamic reallocation of virtual machines. The parameters include: resource demands of virtual machines, time between reallocations and desired level of capacity overflows. Resource demands of virtual machines are represented using stochastic processes (with autoregressive properties of a process used to quantify the variability on the migration timescales).

Even though the actual gain from the dynamic reallocation approach depends on the details of resource demand distributions, we manage to provide closed-form solution that depends solely on the first two moments of those distributions and the values of their autocorrelation functions. It is possible by use of central limit theorem which provides very good approximation for larger data center sizes. We verify the model using extensive data center simulations and use it to discover relationships between the parameters mentioned above and the amount of gain in average data center utilization. The gain is defined as the reduction in the number of physical machines required to host the workload as compared to the number required by static server consolidation while assuming the same requirement on the maximal rate of capacity overflows.

One more aspect of the study reported in this paper is analysis of large number of server CPU utilization traces focused on estimation of typical values of parameters that we have identified as important for the model. Statistics include the variability of server workload and level of autocorrelation in the traces.

2. Problem Formulation

A system that we are interested in modeling consists of a set of physical servers (PMs) each equipped with the virtualization software. The virtualization software allows for concurrent execution of several virtual machines (VMs) on a single physical server. Each of those virtual machines executes a standard operating system. Moreover, an executing VM can be moved from one physical host to another.

In *static allocation* virtual machines are assigned to physical servers once and the mapping does not change. The mapping is decided by statistical properties of workloads so the likelihood of a group of virtual machines hosted together on a server of overloading the server's capacity is no greater than the desired value.

In *dynamic allocation* the virtual machines can keep moving between the physical servers by means of described above "live migration". The decisions to migrate or not are based on the historical knowledge of workloads and the allowed migration frequency. We assume that the reallocation can happen at intervals of length T. Time T is chosen based on the costs of the migration process so the migration overheads do not reduce the system's utility.

Given a set of virtual machines (as specified by the statistical properties of their workloads), the length of a reallocation time interval T, and the overload probability p we want to compute the difference between the number of physical machines required to host the workload with the static allocation and the number required by the dynamic allocation policy. Precisely, for a given number of m physical machines static allocation approach computes the required capacity C of each of the m identical physical servers for which the required degree of capacity overflows can be achieved (i.e., an aggregate load of $\frac{n}{m}$ VMs exceeds C with probability not greater than p). Next, we compute the number m_D of servers (each having the same capacity C) that is required to host the same set of virtual machines while using the dynamic reallocation approach. The difference between the two is the gain from dynamic reallocation.

The subsequent section derives analytical model for this and presents the closed-form approximate solution based on the central limit theorem. The approximation becomes very good as the number of virtual machines and the density of VMs per server increase. The model assumes that stochastic processes representing virtual machine demands are stationary, and i.i.d. While the assumption about the processes being identically distributed can be relatively easily relaxed, the stationarity and pair-ways independence assumptions for stochastic processes representing demands are crucial for the analysis. While we certainly acknowledge that real demand processes are often correlated we still believe that the model provides many useful insights into the benefits of the dynamic allocation of virtual machines.

3. Analytical Model of Gain from Dynamic Reallocation of Virtual Machines

Resource demand of virtual machine i is represented by a discrete time stochastic process $X_i(t)$, for $i = 1, \ldots, n$ and $t > 0$. Assume that processes $X_i(t)$ are stationary, independent, and identically distributed. Thus denote:

$$F(x) = P(X_i(t) \le x) \tag{1}$$

Since $X_i(t)$ are identically distributed for any k define distribution of sum of k demand processes as

$$F_k(x) = P\left(\sum_{i=1}^{k} X_{j_i}(t) \le x\right). \tag{2}$$

for $1 \le j_1, j_2, \ldots, j_k \le n$ and $j_l \ne j_m$ for $l \ne m$. Now assume that we are to statically allocate those n virtual machines to m identical physical machines (with $1 \le m \le n$). The optimal allocation consists of assigning $\lceil \frac{n}{m} \rceil$ virtual machines to each physical machine (with last physical machine containing smaller number of VMs if $n \bmod m \ne 0$). If we want a physical machine to meet demand of virtual machines allocated to it with probability at least $1-p$, with p $(0 < p < 1)$ representing overload probability, the capacity of each of the m physical machines has to be

$$C = F^{-1}_{\lceil \frac{n}{m} \rceil}(p) \tag{3}$$

(where $F^{-1}(p)$ denotes inverse of function $F(x)$) resulting in the total required system capacity of

$$C_{total} = m * F^{-1}_{\lceil \frac{n}{m} \rceil}(p) \tag{4}$$

In order to investigate the amount of resources required to support the same workload with ability to change assignment of virtual to physical machines let's define the second order distribution of the stochastic process $X_i(t)$ as

$$G(x, \tau) = P(X_i(t+\tau) - X_i(t) \le x). \tag{5}$$

for $1 \le i \le n$ and $t, \tau > 0$. Also define the same probability but for an aggregate of k demand processes:

$$G_k(x, \tau) = P\left(\sum_{i=1}^{k} \left[X_{j_i}(t+\tau) - X_{j_i}(t)\right] \le x\right). \tag{6}$$

for $1 \le j_1, j_2, \ldots, j_k \le n$, $j_l \ne j_m$ for $l \ne m$, and $t, \tau > 0$. We are interested in obtaining the number of physical machines (each with the same capacity C) required to support the workload when it is possible to reallocate virtual machines between physical machines. Denote this value by m_D. The aggregate demand of all virtual machines exceeds $F_n^{-1}(p_1)$ with probability p_1. Note that p_1 is a control parameter $(0 < p_1 < p)$ that should be chosen to minimize m_D. Thus each of the m_D physical machines has to support $\frac{F_n^{-1}(p_1)}{m_D}$ of workload. However, if the reallocation of virtual machines can happen only every T time units each physical machine has to have additional free capacity to handle variation of the workload allocated to it within the time interval of length T. Assuming that there is $\lceil \frac{n}{m_D} \rceil$ VMs assigned to one PM this extra capacity is $G^{-1}_{\lceil \frac{n}{m} \rceil}(p - p_1, T)$. This assures that the demand will be met with probability $p - p_1$ bringing total overflow probability to desired level p. Moreover, the above discussion assumes "fluid"-like virtual machines that can be divided between physical machines thus hiding inefficiency of the bin-packing algorithm. We account for that by reducing the available capacity of a PM to its fraction $0 < b < 1$ representing this additional inefficiency. Thus the required condition is

$$\frac{F_n^{-1}(p_1)}{m_D} + G^{-1}_{\lceil \frac{n}{m_D} \rceil}(p - p_1, T) \le b F^{-1}_{\lceil \frac{n}{m} \rceil}(p) \tag{7}$$

Equation 7 represents fundamental relationship between the number of physical machines required by static allocation algorithm (m) and the dynamic allocation algorithm (m_D). It is worthwhile to discuss boundary properties of this solution. If the variability of the demand process is negligibly small on the timescales of the order of T and the inefficiency of the bin-packing algorithm is ignored then $m_d \ge \frac{F_n^{-1}(p_1)}{F^{-1}_{\lceil \frac{n}{m} \rceil}(p)}$ which is simply the perfect division of aggregate load among physical machines. As $G^{-1}_{\lceil \frac{n}{m_D} \rceil}(p - p_1, T)$ increases m_D increases eventually reaching and exceeding m. When that happens dynamic reallocation is no longer a viable option. Specific conditions under which these occur depend on shapes of the probability distribution of aggregate demand as well as probability distribution of change of demand on time scales equal to T.

The Equation 7 can be evaluated numerically (for example, using fixed point computation since G^{-1} depends on m_D) or in some cases symbolically for specific distributions. However, we take a different approach: use Central Limit Theorem to approximate the quantities in Inequality 7. Note that as long as the number of virtual machines n is large distribution $F_n(x)$ approaches normal distribution. Similarly, as long as the densities of VMs per PM remain high (as represented by $\lceil \frac{n}{m_D} \rceil$ and $\lceil \frac{n}{m} \rceil$) the distributions $G_{\lceil \frac{n}{m_D} \rceil}(x, T)$ and $F_{\lceil \frac{n}{m} \rceil}(x)$ approach normal distribution.

Recall that p percentile of a normally distributed random variable with mean μ and standard deviation σ is given by $N_p(\mu, \sigma) = \mu + \sqrt{2}\sigma \; erf^{-1}(2p - 1)$ where

$erf(x) = \frac{2}{\sqrt{\pi}} \int_{t=0}^{x} e^{-t^2} dt$ is the "error function". Denote $E[X_i(t)] = \mu$, $VAR[X_i(t)] = \sigma^2$, $E[(X_i(t+T) - X_i(t))^2] = \sigma_T^2$ for $1 \leq i \leq n$ and $t, T > 0$. Based on the stationarity and i.i.d. assumptions for $\{X_i(t)\}$ we have: $E\left[\sum_{i=1}^{k} X_{j_i}(t)\right] = k\mu$, $VAR\left[\sum_{i=1}^{k} X_{j_i}(t)\right] = k\sigma^2$, $E\left[(\sum_{i=1}^{k} X_{j_i}(t+T) - \sum_{i=1}^{k} X_{j_i}(t))^2\right] = k\sigma_T^2$. for $1 \leq j_1, j_2, \ldots, j_k \leq n$ and $j_l \neq j_m$ for $l \neq m$. Thus

$$\begin{aligned} F_k^{-1}(p) &= k\mu + a(p)\sqrt{k}\sigma, \\ G_k^{-1}(p,T) &= a(p)\sqrt{k}\sigma_T \end{aligned} \tag{8}$$

for $a(p) = \sqrt{2}\,\mathrm{erf}^{-1}(1-2p)$. Assuming $n \bmod m = 0$ and $n \bmod m_D = 0$ (without loss of generality since the equations can be refined to account for extra PM not fully populated with VMs) and using Equations 8 in Inequality 7 we obtain:

$$\begin{aligned} &\frac{n\mu + a(p_1)\sigma\sqrt{n}}{m_D} + a(p-p_1)\sigma_T\sqrt{\frac{n}{m_D}} \\ &\leq b\left(\frac{n}{m}\mu + a(p)\sigma\sqrt{\frac{n}{m}}\right) \end{aligned} \tag{9}$$

Inequality 9 is quadratic in $\sqrt{m_D}$ for which closed form solution can be readily obtained. The roots of the equation (in $\sqrt{m_D}$) are

$$x_{1,2} = \frac{a(p-p_1)\sigma_T\sqrt{n}}{2b\left(\frac{n}{m}\mu + a(p)\sigma\sqrt{\frac{n}{m}}\right)} \pm \frac{\sqrt{(a(p-p_1)\sigma_T\sqrt{n})^2 - 4(n\mu + a(p_1)\sigma\sqrt{n})(b\left(\frac{n}{m}\mu + a(p)\sigma\sqrt{\frac{n}{m}}\right))}}{2b\left(\frac{n}{m}\mu + a(p)\sigma\sqrt{\frac{n}{m}}\right)}. \tag{10}$$

Thus

$$m_D = \begin{cases} x_1^2 & x_1 > 0 \\ x_2^2 & \text{otherwise.} \end{cases} \tag{11}$$

Denote average system utilization for both static allocation and dynamic allocation of VMs to PMs by U_S and U_D, respectively. We have $U_S = \frac{n * \mu}{m * \left(\frac{n}{m}\mu + a(p)\sigma\sqrt{\frac{n}{m}}\right)}$ and $U_D = \frac{n * \mu}{m_D * \left(\frac{n}{m}\mu + a(p)\sigma\sqrt{\frac{n}{m}}\right)}$. Also, the relative improvement in system utilization between system utilization with static allocation and dynamic allocation is $I = \frac{U_S - U_D}{U_S} = 1 - \frac{m}{m_D}$.

4. Empirical Studies

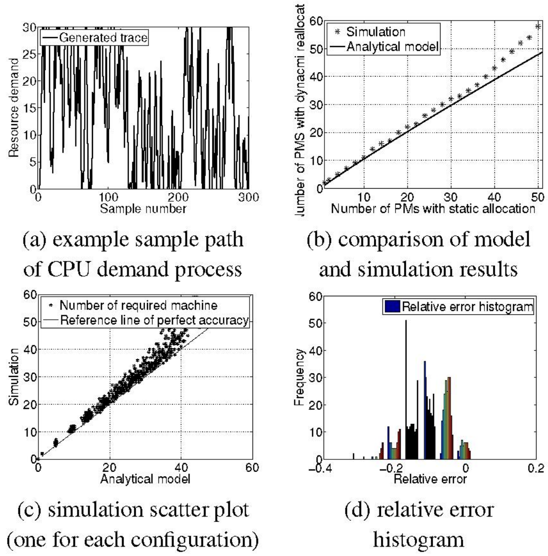
We now present results of empirical studies of the model. First, we focus on verifying the correctness of the model by comparing its results with those of the simulation. Validation is followed by statistical analysis of CPU usage traces from actual servers that reveals the typical levels of variability and autocorrelation. Next, we present numerical results for various sets of parameters that uncover important relationships between the features of workload and length of the VM reallocation period and the amount of gain from dynamic reallocation of VMs. Finally, we discuss our broader conclusions from the empirical study.

Model Validation

In order to validate the correctness of the analytical model presented in Section 3 we have performed a simulation study. We have implemented a simulator that for a given set of time-series representing resource demands of virtual machines computes the number of physical machines required to host this workload under both static allocation and dynamic allocation. The simulator is implemented in Matlab [21] and works in steps of length equal to the reallocation interval. At each step it performs a first-fit bin packing of current values of the demand processes into the bins representing PMs. Capacity of each PM is set to the capacity required by the static allocation of VMs to PMs. The criteria whether one more VM can be added to a PM is based on whether the total load on the PM plus $G_c^{-1}(p,T)$ is smaller than the PM's capacity, where G is defined in Equation 6, c is an average density of VMs per PM ($\frac{n}{m_D}$), p is desired rate of capacity overflows, and T is the reallocation period. At the end of each simulation run the maximum number of used PMs is computed (note that in each step dynamic algorithm uses potentially different number of PMs). The number is compared with the result produced by our analytical model (Equation 11 with the preceding derivation).

Sample paths of stochastic processes required for simulation are generated using standard techniques of stochastic process generation. For each simulation experiment new set of demand traces is generated and fed to the simulator. We have simulated 90 parameter configurations covering large spectrum of possible values of workload variability and of the ratio of $\frac{\sigma_T}{\sigma}$ for 100 VMs over 1000 time units. The parameter p ranged between 0.01 and 0.5 and m between 1 and n. The ratio of $\frac{\sigma_T}{\sigma}$ ranged between 0.1 and 1.

Figure 1 presents an example comparison between the model and simulation results for one simulation run as well as aggregate results. Figures 1a and 1b show an example experiment with 100 VMs. Figure 1a presents a generated evolution of CPU demand of a VM and 1b the comparison of simulation results with those predicted by our model. The capacity of a physical machine ranged from sufficient to host (on average) only 2 VMs to sufficient to host all 100



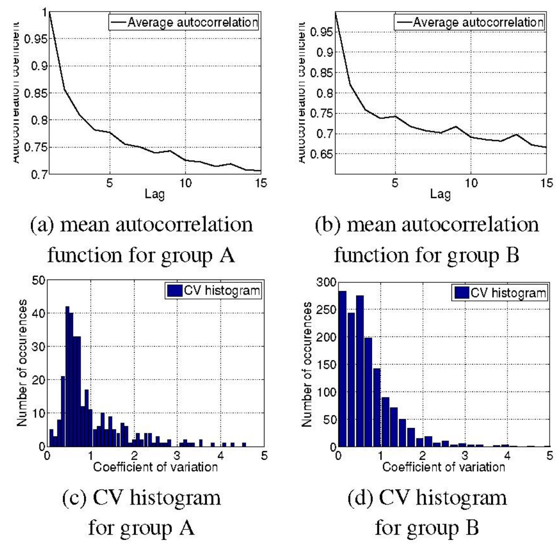
(a) example sample path of CPU demand process

(b) comparison of model and simulation results

(c) simulation scatter plot (one for each configuration)

(d) relative error histogram

Figure 1. Validation of the analytical model.



(a) mean autocorrelation function for group A

(b) mean autocorrelation function for group B

(c) CV histogram for group A

(d) CV histogram for group B

Figure 2. Analysis of CPU utilization for two groups (total of 1791 traces).

VMs (this is why the number of physical machines on the x-axis ranges between 1 and 50) while keeping the remaining parameters constant. Each dot in 1b represents a simulation for a single value of capacity. In aggregate we have found the results computed using the analytical model to be within 20% of the values obtained using the simulation. The scatter plot of results is presented on Figure 1c and the histogram of relative error in Figure 1d. The model tends to underestimate the required number of PMs. The reason is that we have set parameter b (which represents the inefficiency of bin-packing algorithm) to 0. The parameter can be fine tuned either based on empirical results or on analytical relationship between the number of required bins when packing discrete-size indivisible objects versus "fluid-like" objects that can be divided between bins.

Resource Demand Dynamics of Data Center Traces

An important aspect of using the model we have developed is a good understanding of realistic parameters of the real workload. To ensure that our discussion of model results considers parameter ranges observed in reality we have performed a simple statistical analysis of large group of CPU demand traces of production servers. We are interested in two important features of workload: its coefficient of variation and autocorrelation function. These are the two parameters of the model that affect the gain from dynamic reallocation of VMs.

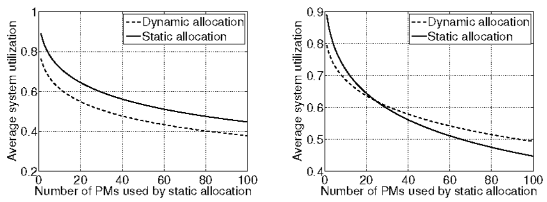
The trace collection contains 1791 traces each containing 15-minute averages of CPU utilization taken using standard methods (*sar* program on Unix and Linux servers and Windows Management Interface on Windows servers). The set consisted of two groups: group A containing 1473 5-day traces and group B containing 318 50-day traces. The traces represent diverse population of machines serving wide variety of uses ranging from web and email servers to the application and database servers. Even though the trace values represent utilization (thus can not be compared between the servers which can have different CPUs) it does not pose the problem because two metrics we are interested in (coefficient of variation and autocorrelation function) do not depend on actual values (have no unit).

Figure 2 presents our findings. Plots 2a and 2b show results of averaging of autocorrelation functions within each of the groups. Next, we consider the variability of workloads as measured by coefficient of variation. Histograms for both groups A and B are presented on plots 2c and 2d, respectively. Mean coefficient of variation for A is 0.74 and for B is 0.98.

Empirical Results

The model that we have formulated in Section 3 can be used to gain important insights how the crucial parameters of the workload (long term and short term variability components) and the time constant of migration interval affect the gain from dynamic reallocation of virtual machines.

Figure 3 presents two examples illustrating the differences between average system utilization achieved by a dynamic allocation and the static allocation for configuration $\mu = 20$, $\sigma = 15$, $n = 100$, $p = 0.05$ with $\sigma_T = 15$ in (a) and $\sigma_T = 10$ in (b). Recall that in our model the static allocation (when given m machines) uses servers with capacity



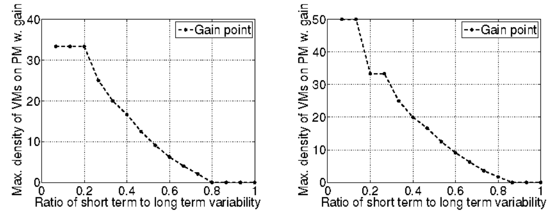
(a) $\sigma_T = 15$ (b) $\sigma_T = 10$

Figure 3. Examples of the gain (or lack of thereof) from dynamic reallocation of virtual machines for $\mu = 20$, $\sigma = 15$, $n = 100$, $p = 0.05$.

which is optimal for a given workload (computed based on the p percentile of the demand distribution of aggregate of $\frac{n}{m}$ VMs). In configuration in plot 3a the variability is fully contained in frequencies higher than $\frac{1}{T}$ thus dynamic reallocation of VMs is not bringing any improvement. In configuration in plot 3b the significant fraction of variability is in frequencies lower than $\frac{1}{T}$ thus dynamic reallocation of VMs can bring improvement for densities lower than 5 VMs per PM.

The second interesting relationship is the dependence between the ratio $\frac{\sigma_T}{\sigma}$ and the maximum density of VMs per PM under which there is a gain from dynamic reallocation of virtual machines. The ratio represents what fraction of variability occurs in frequencies higher than $\frac{1}{T}$. The density relates to the average number of VMs per PM thus to the degree of averaging (and thus variability reduction) that happens on each PM. Figure 4 presents the results for two different sets of parameters (plot 4a for $\mu = 30$, $\sigma = 15$, $n = 100$, $p = 0.05$ and plot 4b for $\mu = 10$, $\sigma = 15$, $n = 100$, $p = 0.01$). The x-axis represents the ratio $\frac{\sigma_T}{\sigma}$ and the y-axis the maximum density ($\frac{n}{m}$) under which there is a gain from dynamic reallocation of VMs. As expected the maximum density allowed decreases as the variability shifts to higher frequency ranges. The model allows for precise quantification of this effect.

The final round of experiments quantifies the effect of coefficient of variation of workload on the amount of gain from dynamic reallocations. We have evaluated the model for several values of cv (ranging from almost deterministic value of 0.03 to highly variable 2.0), two overflow target rates ($p = 0.01$ and $p = 0.05$) as well as two levels of ratio $\frac{\sigma_T}{\sigma} = 0.4$ and $\frac{\sigma_T}{\sigma} = 0.8$. The results are shown in Figure 5. Each plot shows the number of PMs required by static allocation on x-axis and the one required by dynamic reallocation on y-axis. Static allocation line ($m = m$) is provided as a reference. In each case the amount of gain reduces with smaller values of coefficient of variation. Thus the less variability in the workload the less potential for optimization. Also, looking at plots 5a, 5b, and 5c it is apparent



(a) $\mu = 30$, $\sigma = 15$, $n = 100$, $p = 0.05$ (b) $\mu = 10$, $\sigma = 15$, $n = 100$, $p = 0.01$

Figure 4. Quantification of effect of the ratio of $\frac{\sigma_T}{\sigma}$ (x-axis) on the maximum density of VMs per PM (y-axis) resulting in the gain from dynamic reallocation of VMs.

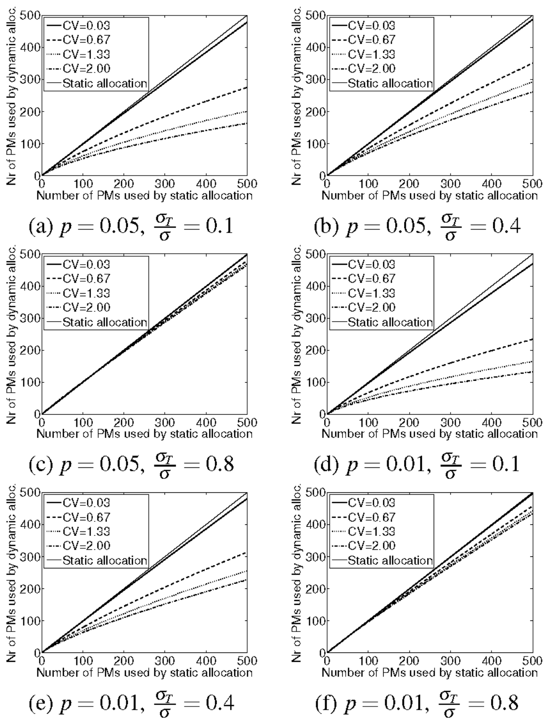
that for a fixed overflow target $p = 0.05$ the gain decreases with increasing ratio of $\frac{\sigma_T}{\sigma}$ which corresponds to the fraction of variability in frequencies higher than $\frac{1}{T}$. Similar effect can be observed on plots 5d, 5e, and 5f for $p = 0.01$. Another interesting dependency is relatively little effect of changes in p keeping other parameters constant. Comparing plots 5a and 5d is an example. Same holds true for pairs 5b and 5e and 5c and 5f. This last behavior can be explained by the fact that lowering p has symmetrical effect on both long term and short term demand distributions since for larger densities of VMs per PM both converge to normal distribution thus have the same percentile behavior.

Conclusions from Empirical Studies

Empirical results presented above verify that the model is adequate in representing the amount of gain due to dynamic reallocation of VMs. It can be used to rapidly discover relationships among important parameters affecting this gain. Results of empirical experiments lead to conclusion that the gain from dynamic management is significant when: (1) the density of VMs per PM after the consolidation is not very high, i.e., the aggregate workload retains significant variability, and (2) fraction of workload variability that resides in frequencies higher than the inverse of reallocation interval is smaller than 0.6. Also, the effect of the overflow target p on the gain is small when others parameters are kept constant.

5. Related Literature

Operating system level virtualization is regaining popularity since its inception by IBM in the 1960s [9, 15]. Examples of virtualized server environments are IBM's Power 5 [8], Xen [2, 11, 34], and VMWare [30, 32, 33]. Another



(a) $p = 0.05$, $\frac{\sigma_T}{\sigma} = 0.1$
(b) $p = 0.05$, $\frac{\sigma_T}{\sigma} = 0.4$
(c) $p = 0.05$, $\frac{\sigma_T}{\sigma} = 0.8$
(d) $p = 0.01$, $\frac{\sigma_T}{\sigma} = 0.1$
(e) $p = 0.01$, $\frac{\sigma_T}{\sigma} = 0.4$
(f) $p = 0.01$, $\frac{\sigma_T}{\sigma} = 0.8$

Figure 5. Effect of coefficient of variation of workload on the amount of gain from dynamic reallocation of VMs for several levels of overflow target (p) and ratios of $\frac{\sigma_T}{\sigma}$.

example is *Cellular Disco* [14] which leverages an existing operating system to provide an abstraction of a virtual machine.

There is a large body of literature devoted to dynamic resource management in virtualized environments. However, the major focus is on developing algorithms rather than on quantifying the possible gain from dynamic control. The authors of [4] employ prediction techniques and queuing theory results to allocate resources efficiently within a single server serving a web workload. Static allocation approach is used in [27] where authors propose a simple heuristic for vector bin-packing problem and apply it to minimize the number of servers required to host a given web traffic. In [1] control theory is applied to design a system for performance control of web server. The arrival rate of requests to the server is throttled based on the feedback system. [5] introduces a concept of resource economy in the web server hosting center. The authors propose an optimization algorithm that allocates resources depending on the expected financial gain. Similar work is presented in [24], which in addition to the economic model applies the feedback control loop to drive resource allocation. Resource overbooking is advocated in [31] as a means of increasing the revenue generated by available resources in a shared hosting platforms. However, only static allocation is considered. Work of [28] presents an integrated resource management framework for quality of service in network services. The notion of quality-aware service-yield is introduced to trade-off the resource requirements with the cost of supporting the workload. The authors of [19] propose an allocation algorithm that attempts to minimize the number of migrations while reallocating resources. A similar objective is also pursued in [17]. The presented algorithm attempts to minimize the number of migrations of virtual machines while minimizing the number of physical machines used. A resource management algorithm for grids is presented in [25]. It provides statistical guarantees for the amount of resources needed to satisfy the combined requirements of multiple applications. A theoretical algorithm for scheduling of tasks is presented in [18]. The authors focus on providing a fair schedule while minimizing job migrations. Computing node reallocation is analyzed in [7]. The authors focus on data access aspects and develop a caching scheme and migration strategy that optimizes decisions based on the local availability of the data. However, we are not aware of analytical models quantifying the gain from dynamic reallocation of virtual machines.

Our prior work [3] focused on proposing an approach to quantify which workloads have a potential to gain from dynamic reallocation. We have also proposed [20] an analytical model of a single reallocation step and its expected effect on system's responsiveness.

Finally, there has been significant amount of research on utility of process migration for performance optimization [10, 12, 16, 23, 29]. The conclusion is that the migration can present substantial benefits if executed properly, i.e., based on good understanding of lifetime distribution of a process. This work is related to our analysis, however significant difference stems from the fact that the processes in aforementioned research are treated as independent and thus scheduling them depends solely on the lifetime distribution. In virtual machine migration, however, persistent nature of a VM warrants different approach leveraging statistics of autocorrelation function of resource demand.

6. Conclusions and Future Work

An analytical model relating the gain from dynamic reallocation of virtual machines to the features of the workload (its variability and autocorrelation) and the length of reallocation interval has been presented. The model is applicable to independent and identically distributed virtual machine demand processes. We have validated the accuracy of the model using extensive simulations and found it to be accurate (with maximum relative error smaller than 20%). Moreover, analysis of the real workload traces has been presented shedding light on the typical values of variability and autocorrelation found in server CPU usage traces. Numerical analysis of the model suggests that the gain from dynamic management is significant when: (1) the density of VMs per PM after the consolidation is not very high, i.e., the aggregate workload retains significant variability, and (2) fraction of workload variability that resides in frequencies higher than the inverse of reallocation interval is smaller than 0.6.

Future research plans include extending the model to account for correlation structure between demand processes. Another option is examination and simulation of gain from dynamic management based on the real workload traces.

References

[1] T. Abdelzaher, K. Shin, and N. Bhatti. Performance Guarantees for Web Server End-Systems: A Control-Theoretical Approach. *in Proceedings of the IEEE Transactions on Parallel and Distributed Systems*, 2002.

[2] P. R. Barham, B. Dragovic, and F. et. al. Xen 2002. Technical Report UCAM-CL-TR-553, University of Cambridge, Computer Laboratory, 2003.

[3] N. Bobroff, A. Kochut, and K. Beaty. Dynamic Placement of Virtual Machines for Managing SLA Violations. *in Proc. of the IEEE Symp. on Integr. Management*, 2007.

[4] A. Chandra, W. Gong, and P. Shenoy. Dynamic Resource Allocation for Shared Data Centers using Online Measurements. *in Proceedings of the IEEE International Workshop on Quality of Service*, 2003.

[5] J. S. Chase, D. C. Anderson, P. N. Thakar, A. Vahdat, and R. P. Doyle. Managing Energy and Server Resources in Hosting Centres. *in Proceedings of OSDI*, 2001.

[6] C. Clark, K. Fraser, S. Hand, J. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield. Live Migration of Virtual Machines. *in Proceedings of NSDI*, 2005.

[7] A. Coles and A. Edwards. Rapid node reallocation between virtual clusters for data intensive utility computing. In *CLUSTER*, 2006.

[8] I. Corporation. Advanced POWER Virtualization on IBM System p5. http://www.redbooks.ibm.com/abstracts/sg247940.html.

[9] R. Creasy. The Origin of the VM/370 Time-Sharing System. *IBM Journal of Research and Development*, 1981.

[10] A. Downey. A note on "The Limited Performance Benefits of Migrating Active Processes for Load Sharing", 1995.

[11] B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, I. Pratt, A. Warfield, P. Barham, and R. Neugebauer. Xen and the Art of Virtualization. *in Proc. of OSDI*, 2003.

[12] D. Eager, E. Lazowska, and J. Zahorjan. The limited performance benefits of migrating active processes for load sharing. *ACM SIGMETRICS*, 1988.

[13] R. Goldberg. Survey of Virtual Machine Research. *in IEEE Computer Magazine*, 1974.

[14] K. Govil, D. Teodosiu, Y. Huang, and M. Rosenblum. Cellular Disco: Resource Management using Virtual Machines on Shared Memory Multiprocessors. *in Proceedings of Symposium on Operating Systems Principles*, 1999.

[15] P. Gum. System/370 Extended Architecture: Facilities for Virtual Machines. *IBM Journal of Research and Development*, 1983.

[16] M. Harchol-Balter and A. B. Downey. Exploiting process lifetime distributions for dynamic load balancing. *ACM Transactions on Computer Systems*, 15(3):253–285, 1997.

[17] G. Khanna, K. Beaty, A. Kochut, and G. Kar. Dynamic Application Management to Address SLAs in a Virtualized Server Environemnt. *in Proceedings of IEEE/IFIP Network Operations and Management Symposium*, 2006.

[18] T. Kimbrel, B. Schieber, and M. Svirdenko. Minimizing Migrations in Fair Multiprocessor Scheduling of Persistent Tasks. *in Proc. of ACM-SIAM Symp. on Discrete Algorithms*, 2004.

[19] T. Kimbrel, M. Steinder, M. Svirdenko, and A. Tantawi. Dynamic Application Placement under Service and Memory Constraints. *in Proceedings of International Workshop on Efficient and Experimental Algorithms*, 2005.

[20] A. Kochut and K. Beaty. On Strategies for Dynamic Resource Management in Virtualized Server Environments. *in Proc. of the IEEE MASCOTS*, 2007.

[21] MathWorks. Matlab. http://www.mathworks.com/.

[22] M. Nelson, B. Lim, and G. Hutchins. Fast Transparent Migration for Virtual Machines. *in Proc. of Usenix*, 2005.

[23] M. Nuttall and M. Sloman. Workload characteristics for process migration and load balancing. In *International Conference on Distributed Computing Systems*, pages 0–, 1997.

[24] G. Pacifici, M. Spreitzer, A. Tantawi, and A. Youssef. Performance Management for Cluster Based Web Services. *in Proc. of the IEEE Integrated Mgmt.*, 2003.

[25] J. Rolia, X. Zhu, M. Arlitt, and A. Andrzejak. Statistical service assurances for applications in utility grid environments. *Performance Evaluation Journal*, 58, 2004.

[26] M. Rosenblum and T. Garfinkel. Virtual machine monitors: Current technology and future trends. In *IEEE Computer*, 2005.

[27] J. Shahabuddin, A. Chrungoo, V. Gupta, S. Juneja, S. Kapoor, and A. Kumar. Stream-Packing: Resource Allocation in Web Server Farms with a QoS Guarantee. *Lecture Notes in Computer Science*, 2001.

[28] K. Shen, H. Tang, T. Yang, and L. And. Integrated Resource Management for Cluster-based Internet Services. *in Proceedings of the OSDI*, 2002.

[29] C. Steketee. Process migration and load balancing in amoeba. *Twenty Second Asutralian Computer Science Conference*, 1999.

[30] J. Sugerman, G. Venkitachalam, and B. Lim. Virtualizing I/O Devices on VMWare Workstation's Hosted Virtual Machine Monitor. *in Proceedings of Usenix*, 2001.

[31] B. Urgaonkar, P. Shenoy, and T. Roscoe. Resource Overbooking and Application Profiling in Shared Hosting Platforms. *in Proc. of OSDI*, 2002.

[32] VMware. http://www.vmware.com.

[33] C. Waldspurger. Memory Resource Management in VMware ESX Server. *In Proc. of OSDI*, 2002.

[34] Xen. http://www.xensource.com.