Engineering of DVFS-PowerManagement for Cloud Data Center Clusters

Paul J. Kühn

[1] Institute of Communication Networks and Computer Engineering (IKR)
University of Stuttgart, Germany

paul.j.kuehn@ikr.uni-stuttgart.de

Abstract –>***Dynamic Voltage and Frequency Scaling (DVFS) is a method to adapt the energy consumption of electronic devices to the actual processing load at the expense of service delay. This is achieved by a dynamic adaptation of the supply voltage V and the clock frequency f such that Service Level Objectives (SLO) are still maintained. In this paper a novel analysis of a model for the control of operating DVFS is solved for Server Clusters of Cloud Data Centers (CDC) under prescribed bounds of SLOs which are defined by Service Level Agreements (SLA) either on the average or on the quantiles of service delays. The method is based on the theory of queuing models of the type GI/G/n for the server cluster and performed for the Intel Pentium M Processor with Enhanced SpeedStep Power Management as an example. As result of this method precise bounds are provided for the load range λ of arriving service request rate for each power mode which guarantee minimum power consumption dependent on given SLA values. As the instantaneous load in a CDC can be highly volatile the current load level is usually monitored by the cluster operating system which results in a rather high frequency of DVFS range changes and corresponding overhead. An automated smoothing method is suggested based on a Finite State Machine (FSM) with hysteresis levels which reduces the frequency of DVFS range changes significantly. This model of a Finite State Machine with hysteresis levels can be easily implemented by monitoring two parametrs of the cluster state (X,Z), where X is the number of busy servers and Z the number of waiting jobs.***

1 Introduction

Cloud Data Centers (CDC) have become an enabling system component for virtualized network operation and Cyber-Physical Systems. Their energy consumption for computing and cooling is growing fast and amounts already to about 1.5 % of the global energy consumption. The main energy is consumed through computations by servers for processing, data storage, and internal communications. Cooling is required to protect the microelectronic parts against overheating and outage. According to experience, the energy consumption of the electronic parts of a data center amounts typically to about one third of the total energy consumption, expressed by the so-called "Power Usage Effectiveness" (PUE), defined as the fraction of the power consumptions of the whole data center and of the electronic devices only [1]. By more efficient usage of the electronic parts both, electronic and cooling power consumptions can be reduced simultaneously. During the last two decades much efforts have been undertaken on the "greening" of CDCs by dynamic activation/deactivations and sleep mode operations of CDC servers (Server Consolidation), by dynamic load assignments among different server clusters or DCs (Load Balancing) through Virtual Machine (VM) Migrations to under-loaded server groups or CDCs, or by Dynamic Voltage and Frequency Scaling (DVFS) through theoretical studies on operational strategies, performance modeling and optimization, or experimentally by simulation tools or by testbed benchmarks. For an overview on main contributions of energy efficiency operation methods we refer to [2-5]. For studies on DVFS specifically we refer to references [6-10].

In this paper an analytical performance of a queuing model of the general type GI/G/n is used for the evaluation of a cloud server cluster operating under the DVFS load-dependent strategy, being operated automatically by the cluster operating system to meet either mean values or quantiles of prescribed threshold values on service delays. As the dynamic load of cloud data centers may be highly volatile, the operating system has to monitor the load level. Load changes could result in frequent changes of V and f and corresponding additional processing and time overhead. To reduce such load range changes significantly an automatic control is proposed based on a Finite-State Machine model for the system state (X,Z), where X indicates the actual number of busy cluster servers and Z the actual number of waiting jobs. The FSM is based on a two-dimensional hysteresis state transition diagram with SLA-dependent thresholds.

The remaining parts of this paper are as follows: In Section 2 the properties of DVFS will be reminded based on the example of the Intel Pentium M Processor and prescribed Service Level Objectives (SLO) between tenants and the CDC operation management. To connect these two totally independent aspects we will model the DVFS problem in Section 3 by methods of Queuing Theory using an n-server cluster queuing model of type GI/G/n with general job arrival processes (GI) and general type of service processes (G) which provides the relation between the tenant's negotiated SLA requirements on response time, and the server cluster performance. Through this method we establish the optimum operating load ranges for minimum power consumption of the data center server cluster. Section 4 addresses the problem how DVFS can be implemented in the Operating System of the Server Cluster for an automated operation with very low frequencies of power state changes in case of highly fluctuating service requests. Section 5 concludes the paper with a summary.

2 DVFS and Service Level Objectives

First, The power consumption P of microelectronic devices as CMOS-transistors follows approximately the law $P \sim C \cdot V^2 \cdot f$, where C denotes the capacitance of the transistor, V the supply voltage, and f the clock frequency. For energy saving purposes this relation suggests to reduce the supply voltage as much as possible; this, however, requires more time for charge exchanges, which results in a lower clock operating frequency f. Below, we will consider the special processor Intel Pentium M as a typical example which will be the base for the studies in this paper.

According to the original Intel White Paper on the Enhanced SpeedStep Technology for the Pentium M Processor [11] 6 Power States P0,...,P5 are distinguished together with their clock frequencies f_i and supply voltages V_i, c.f. Table I. Assuming an average processing time of $h_0 = 1$ s as time unit for one CDC service request during state P0 we have completed Tables and Charts of [11] for this request by the required average processing times h_i in seconds, the power consumptions P_i of this job in Watts (W), and its energy consumption E_i in Ws. The implementation of DVFS is accomplished by 2 processor registers named IA32_PERF_CTL and IA32_PERF_STATUS through Get State and Change State commands. For the decision on state

TABLE 1 POWER STATES OF THE INTEL PENTIUM M PROCESSOR

P-State	Frequency f_i	Voltage V_i	Proc.Time h_i	Power P_i	Energy E_i
P0	1.6 GHz	1.484 V	1.00 s	25 W	25 Ws
P1	1.4 GHz	1.420 V	1.25 s	15 W	17 Ws
P2	1.2 GHz	1.276 V	1.50 s	10 W	13 Ws
P3	1.0 GHz	1.164 V	1.75 s	8 W	10 Ws
P4	0.8 GHz	1.036 V	2.00 s	7 W	8 Ws
P5	0.6 GHz	0.956 V	2.25 s	6 W	6 Ws

changes a power manager and a processor driver are responsible. Decisions can be based on various inputs as user power policy, processor utilization, battery level (in portable devices), thermal conditions, or events. In our application of CDC server clusters we will assume that the decision is based on the current processor load or utilization **and** a negotiated SLA criterion on the mean **or** quantile of service delays.

Typical Service Level Agreements (SLAs) in a CDC environment refer to the response times of the CDC to service requests. In interactive applications, as in case of Cyber - Physical System (CPS), Smart Grid, Software-Defined Networking (SDN), production automation, traffic or health control, real-time performance criteria are of prime interest. In this paper we will, therefore, define the SLA by the mean of the random waiting time T_W of an arriving service request when it has to wait, $t_W = E[T_W | T_W > 0]$, or the quantile probability $p = P\{T_W > t_{Th} | T_W > 0\}$ by which service requests would have to wait longer than a threshold time $t_{Th.}$ As every state change requires some overhead time (acc. to [11] the Enhanced Intel SpeedStep Technology hardware unavailability is 10 μs instead of 250 μs before), we should try to further reduce the frequency of changes between Power States, c.f. Section 4.

The problem will be solved by establishing a functional relationship between the SLAs, prescribed either by the mean delay t_W or by the quantile p for exceeding a threshold delay t_{Th}, and the job arrival rate λ. These relationships are provided in principle by the performance analysis of the queuing model of type GI/G/n. The problem is, however, that theoretically exact results are only known for some special cases of the model GI/G/n and, secondly, in forward direction only from a given load figure to performance, where we

need the inverse direction from the performance to the load figure. We will solve this problem by a typical traffic engineering approach, c.f. Section 3.

3 Modeling of the DVFS Operation

The best way to model a server cluster is a queuing model with n servers and an unlimited buffer space for arriving service requests which cannot be served immediately at the instant of arrival when all n servers are occupied. Fig. 1 shows this standard queuing model of the type GI/G/n acc. to Kendall's notation, where "GI" denotes the type of arrival process (**G**enerally distributed and **I**ndependent interarrival times), "G" denotes the type of service time process (**G**enerally distributed service times), and n denotes the number of servers.

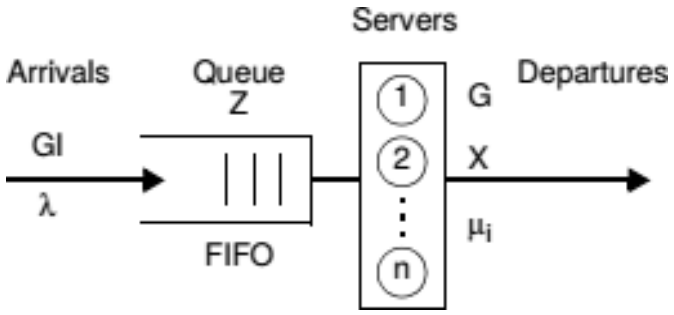

Fig.1. Queuing Model GI/G/n for a CDC Server Cluster

The queuing model is further specified by:

- the arrival rate λ of service requests (mean number of arrivals per second)
- the service rate μ_i of the servers in Power Mode i, where $h_i = 1/\mu_i$ denotes the mean service time of a job in Power Mode i, i = 0,1,...,5
- the queuing service discipline FIFO (first-in, first-out)
- the system state random variables (X,Z), where X denotes the number of busy servers and Z the number of buffered service requests.

Queuing-theoretic analyses will not be presented in this paper; for details we refer to a forthcoming paper [20].

3.1 DVFS for the SLA on Mean Delays of Delayed Jobs

In Fig.2 the mean waiting times t_{Wi} of the 6 Power Modes are plotted dependent on the request arrival rate λ for n = 100 servers for the queuing model M/M/100, where "M" denotes negative-exponentially distributed inter-arrival and service times, respectively. Note, that the operation range for Power Mode P0 is up to λ = 100 1/s; at λ = 100 arriving jobs/s the system is saturated and the mean waiting time t_W approaches infinity asymptotically; this means that $\lambda_{0,max} = n/h_0 = 100$ 1/s. We will use the same model for all Power Modes; as the mean service times increase with slower servers (named "processing times" in Table 1) the system saturates for smaller arrival rates λ accordingly, i.e. $\lambda_{i,max} = n/h_i$. At the intersection between the curves for the mean waiting times and the SLA threshold value, e.g., $t_{WT} = 0.05$, we find the upper bounds for the service request arrival rates λ_i for Power Mode i, i = 0, 1,..., 5. The Power Mode operating ranges for the SLA criterion of mean waiting times of delayed service requests t_{WT} are given in Table 2 for the 4 queuing model examples M/M/n, M/D/n, GI/M/n and GI/D/n for n = 100 where D denotes constant service time and GI hyper-

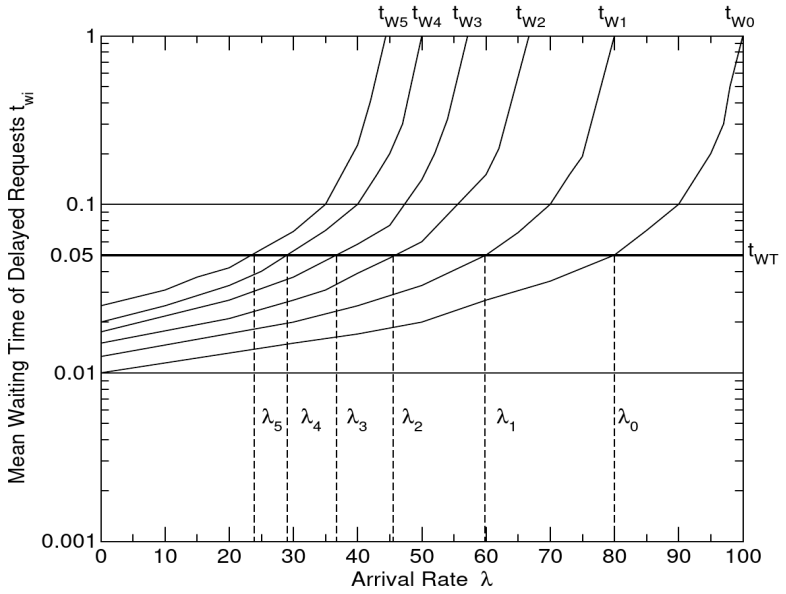

Fig. 2. Mean Waiting Times of Delayed Requests versus the Request Arrival Rate λ for 6 Cluster Power Modes of the Pentium M Processors.

Example: Model M/M/100 for Mean Waiting Time $t_{WT} = 0.05$ s.

exponential service times with coefficient of variation $c_A = 2$). For the last case we applied approximation results from Tables on Delay Systems of type GI/G/n which are based on phase-type process approximations [16]. Example: The optimum operating range for Power Mode i is $\lambda_{i+1} < \lambda < \lambda_i$, i= 0,1,...,5, $\lambda_6 = 0$.

TABLE 2 OPERATING RANGES λ_i FOR MEAN DELAYS FOR QUEUING MODELS M/M/n, M/D/n, GI/M/n AND GI/D/n FOR MEAN DEALAYS $t_{Wi} = t_{WT} = 0.05$ s AS SLA FOR MEAN WAITING TIME

P-States	P0	P1	P2	P3	P4	P5
Type	Upper Thresholds of Arrival Rates λ_i in Power State P_i					
M/M/100	80	60	46	37	29	24 [1/s]
M/D/100	86	66	51	40	32	25 [1/s]
GI/M/100	64	46	35	26	21	16 [1/s]
GI/D/100	66	48	37	27	22	17 [1/s]

From Table II it can be conjectured that the operating ranges decrease with increasing coefficient of variation of the arrival and service processes.

3.2 DVFS for the SLA on Quantile of Delayed Jobs

In this section we want to show how a more real-time-centric SLA can be guaranteed. For this we choose the quantile p of the waiting time of waiting requests $\{T_W | T_W > 0\}$ as the SLA criterion, defined by the probability

$$p = P\{T_W > t_{Th} | T_W > 0\}$$

which means that waiting times for delayed requests at their arrival instant exceed a threshold time t_{Th} only with probability p ("quantile"). From the exact analyses of the queuing model GI/M/n (which includes M/M/n) it is known that the complementary distribution function of delays is also exponentially, i.e., we have an explicit formula as in case of the model M/M/ n (but with different mean t_{Wi}):

$$W_i^c(t)/W_i = exp(-t/t_{Wi}) = p$$

From this formula we find directly the relation between the mean waiting time t_{Wi} and the SLA tuple (p, t_{Th}):

$$t_{Th} = - t_{Wi} \cdot ln\, p$$

For prescribed SLA tuples (t_{Th},p) we find immediately the corresponding mean waiting time t_{Wi} of a delayed request. With the value of t_{Wi} we find from Fig. 2 $t_{Wi} = f(\lambda,i)$ the upper threshold values (bounds) λ^i for each power mode i.

Fig, 3 illustrates the relationship between the complementary delay distribution function $W_i^c(t)/W_i$, the delay quantile p, and the time t. The delay threshold time $t_{Th,i}$ for delayed requests for Power Mode i follows from the intersection point between the two straight lines: The smaller the threshold time t_{Th}, the smaller must be the average delay time t_{Wi} and , thus, the load level λ .

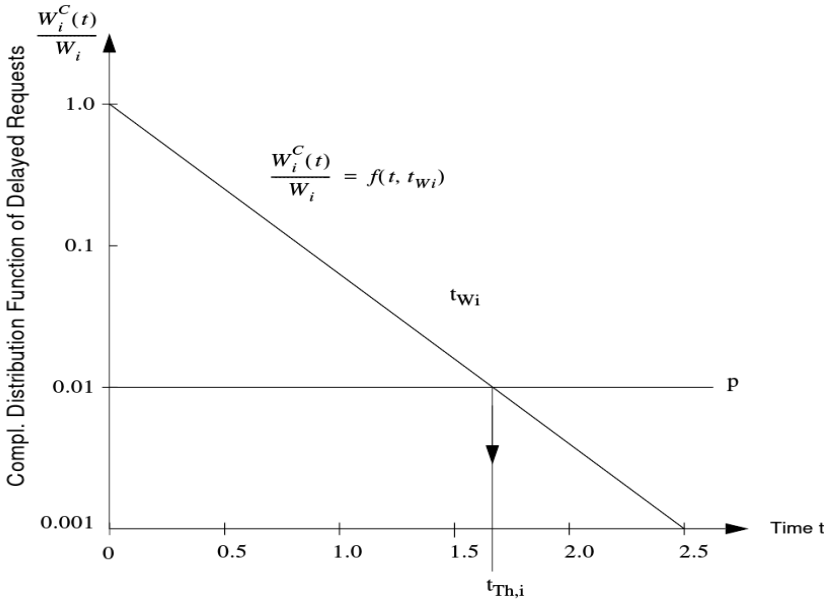

Fig. 3. Complementary Distribution Function of delayed Requests versus Time t for the Queuing Model GI/M/n and Delay Quantile p for the Determination of Delay T Time $t_{Th,}$Threshold (principal diagram).

The Power Mode operating ranges for the SLA criterion of delay quantiles p of delayed service requests are given in Table 3 for the three queuing models M/M/100, M/D/100 and GI/M/100 with exact solutions and for GI/D/100 from the tail approximation using results tabled in [16]. The method for non-exponential delay distributions is based on a theorem on the tail behavior of queuing systems of the type Ph/Ph/n under the FIFO queuing discipline, c.f. Section 3.4.

TABLE 3 OPERATING RANGES λ_i FOR QUANTILES OF DELAY MODELS M/M/n, M/D/n, GI/M/n AND GI/D/nm FOR QUANTILES OF DELAY p = 0.05,THRESHOLD TIME t_{Th} = 0.10 s AS SLA

P-State	P0	P1	P2	P3	P4	P5
Type	Upper Thresholds λ_i/λ^i of Arrival Rates in Power State Pi					
M/M/100	70	50	42	27	19	12 [1/s]
M/D/100	72	56	42	30	20	15 [1/s]
GI/M/100	53	35	25	18	12	8 [1/s]
GI/D/100	54	37	28	16	13	8 [1/s]

3.3 DVFS for SLAs of General Queuing Models GI/G/n

From Queuing Theory we know, that an arbitrary probability distribution function of a random variable can be approximated by a "Phase-type" distribution function (DF), especially by a graph of exponentially distributed phases [12]. This allows to analyze a more complex model by a multi-dimensional Markov Chain. The problem is, however, to find the parameters for the exponential phases by solving a corresponding approximation problem. From experimental measurements we often settle to meet the first two ordinary moments of the DF of a random variable T, $E[T^i]$, i = 1,2, or, equivalently, the mean E[T] and the coefficient of variation (CV) c, where $c^2 = VAR[T]/E[T]^2 - 1$. Simple Phase-type models are just a series of phases or a probabilistic choice between a few (especially 2) exponential phases, allowing for the range of coefficients of variation $0 \leq c \leq \infty$. The corresponding queuing system is a special case of a Ph/Ph/n model, where the phase parameters are derived from given values for the mean E[T] and CV c. Queuing models of type Ph/Ph/n have been tabled in [16].

For queuing models of the Type Ph/Ph/n with queue discipline FIFO Yutaka Takahashi has shown in [17] that the tails of the delay distribution function behave asymptotically exponential:

$$W_i^c(t)/W_i \sim a \cdot exp(-bt) \qquad (8)$$

i.e., they are asymptotically linear on the log/linear coordinate plane for $W_i^c(t)/W_i$. This approximation has been proved as of high accuracy. This is exactly that range of the complementary DF which is relevant for our delay threshold as SLA. In the log/linear plane it can be determined easily as a straight line by two points (0,a) and (t_{Th},p). This allows us to find the load region thresholds the same way as explained before.

Table 4 provides results for the operating load ranges λ_i (for t_{Wi}) and λ^i (for t_{Th},p) for an optimistic/pessimistic case of hypo-/hyper-exponential arrival/service processes of queuing model types $E_3/E_3/25$ and $H_2/H_2/25$ with $c_A^2 = c_s^2 = 1/3$ and 1.5, respectively. For comparison we have placed the results for the basic queuing model M/M/25 between them ($c_A^2 = c_S^2 = 1$). For the quantile study we have assumed, that the waiting time should exceed the doubled mean waiting time value only with probability of 1%. Table 4 emphasizes the striking influence of the stochastic arrival and service processes on the optimum operation load ranges: The higher the variability of the arrival/service processes the higher must be the voltage/frequency mode.

TABLE IV. OPERATING RANGES λ_i/λ^i FOR MEAN QUANTILE DELAYS FOR QUEUING MODELS E3/E3/25, M/M/25 AND H2/H2/25 FOR MEAN DELAYS t_{Wi} AND FOR QUANTILE p OF DELAYS, MEAN WAITING TIME THRESHOLD t_{WT} = 0.4 s, THRESHOLD TIME $t_{Th} = 2t_{WT}$ = 0.8 s, QUANTILE p = 0.01

P-States	P0	P1	P2	P3	P4	P5
Type	Upper Thresholds λ_i/λ^i of Arrival Rates in Power State Pi					
E3/E3/25 λ_i	24.2	19.2	15.8	13.2	11.5	10.2 [1/s]
E3/E3/25 λ^i	23.2	18.2	14.8	12.2	10.5	9.0 [1/s]
M/M/25 λ_i	22.5	17.5	14.0	12.5	10.0	9.0 [1/s]
M/M/25 λ^i	19.5	14.5	11.2	9.3	7.5	6.5 [1/s]
H2/H2/25 λ_i	19.0	11.0	8.3	6.2	5.0	4.0 [1/s]
H2/H2/25 λ^i	17.0	12.2	9.2	7.0	5.5	3.8 [1/s]

4 Implementation Aspects of DVFS

DVFS aims at the adaptation of the power level to the instantaneous traffic load by lowering the power consumption accordingly. This can be accomplished by a permanent monitoring of the current load. As the load is generated by many independent tenants the instantaneous load changes statistically, an effect well-known from massive service systems as the telephone network, or from interactively used computer systems. As already stated above, each change of the power level causes a short time-out which reduces the system capacity and causes extra energy consumption. Besides this effect, load monitoring always lacks behind as statistics are only available over the recent past. In the literature many studies on DVFS were reported using system parameter optimization algorithms where SLO aspects are not considered.

For that reason we advocate for another method of a Finite State Machine (FSM)- based monitoring of the current system state using a staggered hysteresis model for server activations/deactivations which had originally been suggested by the authors for power-saving by server consolidation [18,19]. This principle is based on a strategy to retard new server activation buffering of arriving service requests as long as possible while still guaranteeing a delay-based SLA. This had the effect that the server activation rate could be lowered considerably which is in particular of interest when each server activation requires additional time and energy [6]. For these reasons we would like to suggest applying this method also in connection with DVFS.

The server cluster is modeled by a queuing system with an FSM-controlled operation strategy, c.f. Fig. 4. The system state (X,Z) indicates the current random number of occupied servers X and the random number Z of queued requests which are served according to the FIFO queuing discipline. The function of the control is indicated by a State Transition Diagram (STD), see Fig. 5. At each number x of busy servers new arrivals are queued until state $(x,w^{(x)}-1)$ which guarantees that the SLA on the mean (or on the quantile) of delay will be reached. An arrival at state $(x,w^{(x)})$ will immediately ignite that an idle server has to be activated and starts service, $x = 1,2,...,n-1$, where $w^{(x)} = x \cdot w$, for $x = 1,2, \ldots,n-1$. The value w can be determined easily for the mean waiting time $t_{WT} = w \cdot h$ as SLA in case of exponential service times from the mean residual time until the next server terminates service and x. Similarly, w can also be determined for the case of an SLA for the quantile p and the

delay threshold t_{Th} of delays. Server deactivations take only place when the waiting line is empty.

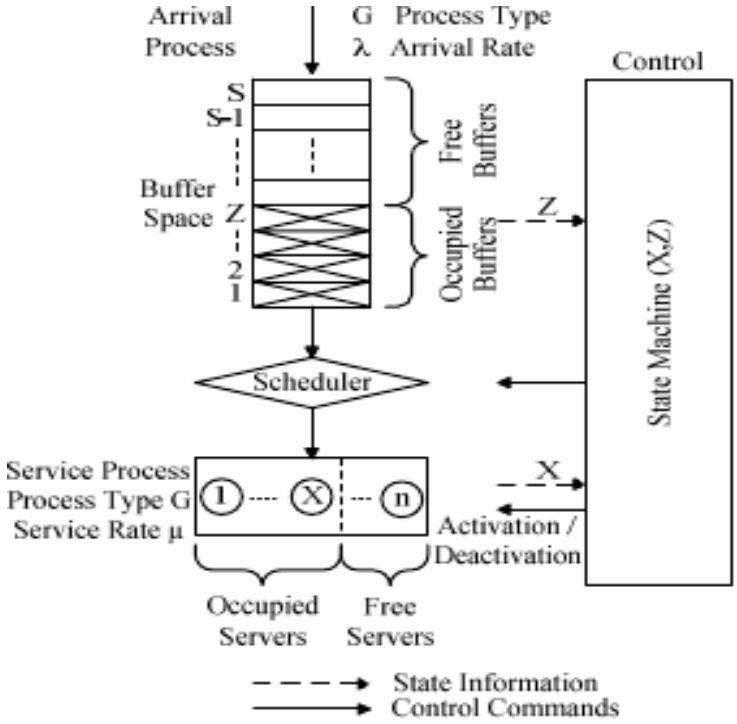

Fig. 4. Model of a Server Cluster with FSM-control

By this strategy the dynamics of server activations/deactivations is greatly reduced while the SLA can still be guaranteed and the system resides for a much longer time with x servers being busy. When this FSM is applied under a DVFS control regime, the temporal and energy overhead involved with DVFS can be reduced. Note, that server activations and deactivations are indicated in Fig. 5 only at states with bold-faced transition arrows. The implementation for an automatic DVFS control is easy, as the state (X,Z) is known by the operating system at any time. without any system state sampling. The transition rates in Fig.4 have been used for the analytic performance evaluation without DVFS and are valid only for exponentially distributed interarrival and service times to determine the server activation rate R_A [18]. In the context of this paper only the **FSM logic** is of importance to decide **when** a Power State has to be changed up or down, respectively.

For the performance analysis within a certain Power State the simpler queuing model of the type GI/G/n without the hysteresis can be applied as outlined in Section 3.

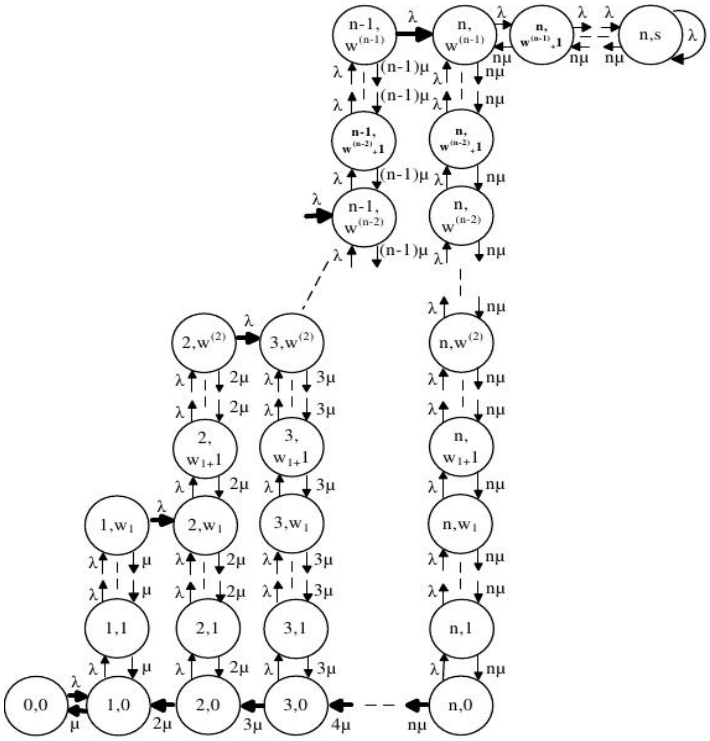

Fig. 5. State-Transition Diagram for a server cluster by FSM control

In Fig. 6 a typical result of the hysteresis model is presented taken from [18], showing the server activation rate R_A versus the job arrival rate λ of events per second, and a constant incremental queue threshold step size $w^{(i)} - w^{(i-1)} = w$, $i = 1,...,n-1$, and $t_{WT} = w \cdot h$ as SLA for the case of the 2-dimensional Markov-Chain Fig.5. The server cluster operating system activates a new server each time when a state $(x,w^{(x)})$ is entered acc. to the bold-faced transitions indicated in Fig. 5, for $x = 1,...,n-1$. The method is also applicable in the case where an SLA for the quantile p has to be met: In that case the condition for the threshold values $w^{(x-1)}$ have to be derived from $W_i^c(t)/W_i \leq p$ in state $(x,w^{(x-1)})$, where $W_i^c(t)/W$ is Erlang-k distributed with $k = x-1$ and service rate x/h.

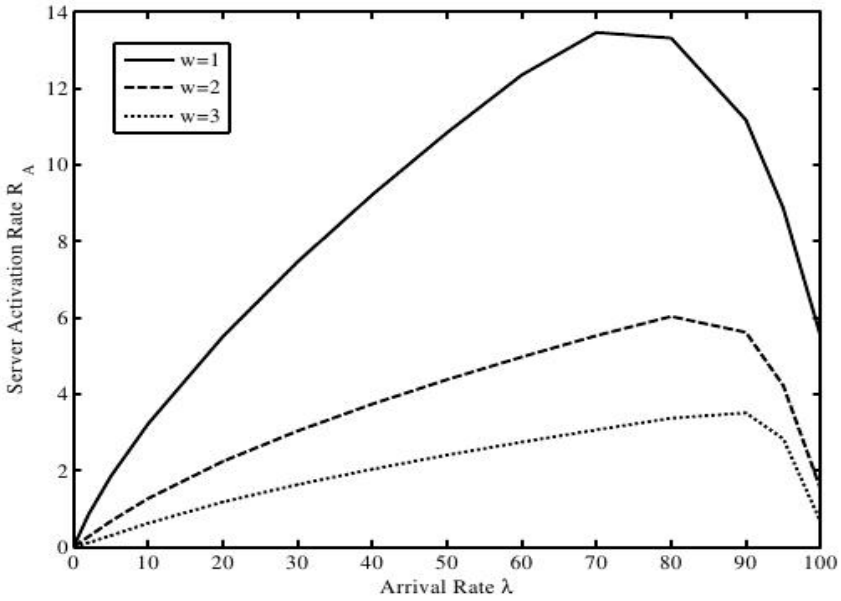

Fig. 6. Server activation rate R_A as function of the job arrival rate.λ for 3 different step parameters w, n = 100 servers, Server Activation Overhead Time h = 1under Markovian assumptions

Notes: After finishing our work we realized two related papers which have appeared quite recently [9,10] which will be discussed shortly.

1. In [9] the method of the CPUFreq Governor for clock scaling is described, where 6 different policies are distinguished:

(1) "Performance": the CPU frequency is set to the highest value within a minimum and a maximum frequency

(2) "Powersave": The CPU frequency is set to the lowest value

(3) "Userspace": The CPU frequency is user-defined

(4). "Ondemand": The CPU frequency depends on the current system load, based on the recent CPU usage statistics

(5) "Conservative":similar as Ondemand, but through graceful changes instead of jumping between the maximua of speed levels

(6) "Schedutil" The CPU frequency follows from a per-entity load tracking mechanism.

The policy "Ondemand" and "Conservative" are the closest ones to our load-depending method in this paper. The differences are that the Linux Governor depends on periodic load statistics without consideration of any SLA conditions. Both policies allow, however, for options as flexibility of the state sampling rate, for options on powersave bias and frequency steps.

2. A recent paper appeared recently on DVFS [10] where a single processor is operated under DVFS and where an M/M/1 queuing model is applied under periodic server utilization checks which requires more overhead when the load is highly variable.

5 Conclusions

DVFS has proved as a powerful method to adapt the power consumption of servers to the application conditions and, in particular, to the actual load. In this contribution a queuing theory-based approach is suggested to adapt the power consumption under Service Level Agreement restrictions with respect to the mean or to the quantiles of service delays. The method allows to fix the lower and upper load level boundaries exactly which are defined by actual service request rates λ_i or λ^i in case of mean or of quantiles of service delays as SLAs to the operative Power Mode levels Pi, respectively. The method applies to general application statistics of the request interarrival and service times. The method is exemplified for the special case of the Enhanced Intel SpeedStep Technology for the Intel Pentium M processor. There exist various different ways how load control can be applied automatically: fine-grained on the individual processor level independently, on the server level, or coarse-grained on the server cluster level of n servers. In this application a server or a whole server cluster are considered where the cluster load situation is controlled by the Operating System. For the latter case a Finite State Machine control method is suggested which is based only on two actual load indicators: the number of busy servers x and the number of delayed service requests z. A staggered hysteresis FSM model is suggested by which the frequency of Power Mode changes can be reduced drastically which contributes to time and energy savings and an easy implementation without specific load level sampling. A simplified application exists also for a single-processor or single-server-system.

REFERENCES

[1] P. Niles, P. Donovan, "Virtualization and Cloud Computing OptimizedPower, Cooling, and Management Maximizes Benefits", White Paper 118, (Rev. 4), Schneider Electric UK, http://news.angelbc-mail.com

[2] A. Ghandi, M.Harchol-Balter, R. Das, C.Lefurgy, "Optimal Power Allocation in Server Farms", ACM SIGMETRICS/Performance '09, June 15 - 19, 2009.

[3] H.H. Kramer, V. Petrucci, A. Subramanian, E. Ochoa, " A Column Generation Approach for Power-aware Optimization of Virtualized Heterogeneous Server Clusters, Publicado em 06/06/2011.

[4] A. Beloglazow, R. Buyya, Y.C. Lee, A. Zomaya, "A Taxomony and Survey of Energy-Efficient Data Centers and Cloud Computing Systems",J. Advances of Computers, Vol. 82, pp. 47 - 99, 2011.

[5] P.J. Kuehn, "Energy-Efficiency and Performance of Cloud Data Centers -Which Role Can Modeling Play ? Inv. Contribution, ACM E²DC Workshop. Waterloo, Canada, 2016, ACM Digital Library.

[6] A. Beloglazov, J. Abawajy, R. Buyya, "Energy-aware Resource Allocation Heuristics for Efficient Mater Systems 28, Elsevier, 2012, pp. 755 - 768.

[7] V.J. Patel, H.A. Bheda, " Reducing Energy Consumption with DVFS for Real.-Time Services in Cloud Computing", IOSR J. of Computer Engineering (IOSR-JCE), Vol. 16, Issue 3, Ver. II, 120143, pp. 53 - 57.

[8] P. Arroba, J.M. Moya, J.L. Ayala, R. Buyya, "Dynamic Voltage and Frequency Scaling-aware Dynamic Consolidation of Virtual Machines for Energy Efficient Cloud Data Centers",Concurrency and Computation,: Practice and Experience, Vol. 29, Issue 10, 31 pages. https://doi.org/10.1002/cpe.4067.

[9] D. Brodowski, "CPU Frequency and Voltage Scaling Code in the Linux (TM) Kernel - Information for Users and Developers -" 7/11/18, https://www.kernel.org/doc/Documentation/cpu-fr

[10] R. Basmadjian, H. de Meer, "Modeling and Analysing ConservativeGovernor of DVFS-enabled Processors", ACM E²DC Workshop, Karlsruhe, Germany, 2018, ACM Digital Library.

[11] -Enhanced Intel SpeedStep Technology for the Intel Pentium M Processor", Intel White Paper, Order Number 30117ß-001, March 2004.

[12] H. Kobayashi, B.L. Mark, "System Modeling and Analysis - Foundations of System Performance Evaluation", Pearson International Edition, Prentice-Hall Inc., Upper Saddle River, 2009.

13] C.D. Crommelin, "Delay Probability Formulae when the Holding Tiomes are Constant",POEEJ 25, 1932, pp. 41 - 50.

[14] W. Whitt, "Approximations for the GI/G/m Queue", J. Production and Operations Management, Vol. 2, No. 2, Spring 1993, pp. 114 - 161.

[15] P. Kuehn, " Tables on Delay Systems", Institute of Switching and DataTechnics, University of Stuttgart, Germany, 1976. Online: http://www.ikr.uni-stuttgart.de/Content/Publications.

[16] L.P. Seelen, H.C. Tijms, M.H. van Hoorn, "Tables for Multi-Server Queues", Elsevier Science Publishers B.V., 1985.

[17] Y. Takahashi, "Asymptotic Exponentiality of the Tail of the Waiting Time Distribution in a Ph/Ph/c Queue", J. Adv. Appl. Probability 13, 1981, pp. 619 - 630.

[18] P.J. Kuehn, M. Ezzat Mashaly, "Automatic Energy Efficiency Management of Data Center Resources by Load-dependent Server Activation and Sleep Modes", Elsevier J.- Ad Hoc Networks, 25, 2015, pp. 497 - 504.

[19] M. Ezzat Mashaly, "Performance of Cloud Data Centers with Service Level Agreement Guarantees", PhD-Dissertation, University of Stuttgart, 2017.[20]

[20] P.J.Kuehn, M. Mashaly, "DVFS-Power Management and Performance Engineering of Data Center Server Clusters", (Submittet to IEEE-Conf. UCC 2018, Zurich, Switzerland, 2018.