Leakage and Temperature Aware Server Control for Improving Energy Efficiency in Data Centers

Marina Zapater*, José L. Ayala†, José M. Moya‡, Kalyan Vaidyanathan§, Kenny Gross§, Ayse K.Coskun¶
*CEI Campus Moncloa UCM-UPM, Madrid 28040, Spain, marina@die.upm.es
†DACYA Universidad Complutense de Madrid, Madrid 28040, Spain, jayala@fdi.ucm.es
‡Electronic Engineering Dept., Universidad Politécnica de Madrid, Madrid 28040, Spain, josem@die.upm.es
§Oracle Physical Sciences Research Center, San Diego, CA 92121, {kalyan.vaidyanathan, kenny.gross}@oracle.com
¶ECE Department, Boston University, Boston, MA 02215, acoskun@bu.edu

***Abstract*—Reducing the energy consumption for computation and cooling in servers is a major challenge considering the data center energy costs today. To ensure energy-efficient operation of servers in data centers, the relationship among computational power, temperature, leakage, and cooling power needs to be analyzed. By means of an innovative setup that enables monitoring and controlling the computing and cooling power consumption separately on a commercial enterprise server, this paper studies temperature-leakage-energy tradeoffs, obtaining an empirical model for the leakage component. Using this model, we design a controller that continuously seeks and settles at the optimal fan speed to minimize the energy consumption for a given workload. We run a customized dynamic load-synthesis tool to stress the system. Our proposed cooling controller achieves up to 9% energy savings and 30W reduction in peak power in comparison to the default cooling control scheme.**

I. Introduction

Energy-related costs are major contributors to the total cost of ownership in data centers today. The increase in energy consumed is due to both rising computational demands and the cooling costs for ensuring reliable operation of computer chips. Data center electricity represents 1.3% of all the electricity use in the world, and 2% in the US, yielding 250 billion kWh consumption per year worldwide [1].

In tandem with the technology scaling to 45nm and beyond, leakage has become an important component of the overall power consumption in computers. Prior work analyzing the effect of leakage on servers [2] highlights the limited usefulness of reducing cooling power by allowing temperature increases in the data center. Leakage power is exponentially dependent on temperature and higher temperatures could potentially lead to higher power consumption. Studies on leakage-temperature tradeoffs need to consider the power dynamics of individual servers under variable fan speeds. As fan power is a cubic function of fan speed, solutions based on over-provisioning of cold air into the servers can easily lead to energy inefficiency. On the other hand, using low fan speeds can increase temperature and leakage power.

In this paper, we present a leakage and temperature-aware server control mechanism that improves energy efficiency of enterprise servers in data centers. Our main contributions are as follows:

- The experimental methodology described in this paper allows to isolate, accurately measure, and control the power consumption associated with the fan speed by separating the power supply of the fans from the server. Such a setup has not been previously used for detailed temperature, leakage and dynamic energy characterization.
- We demonstrate the leakage-temperature tradeoffs in a real server as measured by the Continuous System Telemetry Harness [3]. By splitting the contribution of cooling power from that of leakage and dynamic system power, we derive models for the leakage as a function of the CPU temperature.
- We design a temperature and leakage-aware control policy to dynamically select the fan speed that minimizes the server energy and peak power consumption. Our controller uses a lookup table (LUT), which is generated based on the leakage and fan power analysis. The LUT is addressed by the workload utilization level at runtime and outputs the best fan speed for a given load. Our controller reduces energy consumption by up to 9% for a set of test workloads.

II. Related Work

Related work on fan and cooling control strategies is based on the experimental observation of the over-provisioning of air-flow and over-cooling of servers. Xuefei et al. propose a thermal model-based real-time fan controller to reduce the energy consumption in CPU fans [4]. Other recent methods (e.g., [5]) uses dynamic voltage frequency scaling (DVFS) together with the control of the fan speed in an energy-aware fashion. As opposed to our work, these models have not been tested on a real enterprise server.

A similar technique to ours is proposed by Wang et al. [6]. Their work provides optimal fan speed control for thermal management of servers and tackles the problem of over-cooling. However, they disregard the leakage contributions and its effects on power consumption. To the best of our knowledge, our work is the first to directly experiment with the leakage and temperature tradeoffs on enterprise servers, jointly addressing fan control and leakage power reduction.

III. Experimental Methodology

All the experiments proposed in this paper are performed on a presently shipping enterprise server with two SPARC T3 CPUs in 2 sockets, 32 8GB memory DIMMs and 2 SD hard drives. Each CPU has 16 cores and each core has 8 hardware threads that provide a total of 256 hardware threads. 6 fans, distributed in 3 rows of 2 fans, are located in the front part of the machine, driving air into the server. Airflow first goes through the DIMMs before reaching the CPUs. The power

978-3-9815370-0-0/DATE13/ © 2013 EDAA

supply units (PSUs) and hard disks are located on one side of the server, without interfering in the airflow.

To enable customized dynamic cooling control on the server and for quantifying the leakage power component, we first characterize the fans by verifying their speed with highly accurate vibration sensors and obtaining their power consumption values at each RPM setting. We then use independent power supplies (Agilent E3644A) to directly control each pair of fans. The power supplies are connected via RS-232 to a Data Logging and Control PC (DLC-PC), so that the fan speed can be adjusted by software scripts. The DLC-PC is also responsible for collecting runtime dynamics through the CSTH [3] running on the service processor of the enterprise server. For our experiments, the data collected through CSTH are: (i) 4 CPU temperature values (2 thermal sensors per die); (ii) 32 memory temperature values (1 per DIMM); (iii) per-core voltage and current values; and, (iv) power consumed by the whole system. These data are polled every 10 seconds, providing sufficient visibility into the runtime power and thermal behavior. CSTH runs as a part of the existing system software stack; therefore, the sensor data processing does not introduce additional overhead.

We explore all ranges of utilization scenarios by means of *LoadGen*, a customized dynamic load-synthesis tool that: (i) uses a core algorithm that maximally stuffs the instruction pipes of the multi-threaded CPUs so that the highest theoretically possible gate switching occurs in the chips; and (ii) allows customized dynamic profiles that can meet any desired utilization level by duty-cycling between 100% and idle at a fine granularity. When running *LoadGen*, the system is guaranteed to maintain the given CPU utilization and the workload is evenly spread among the cores. Using deterministic load profiles enables easy characterization of the system behavior.

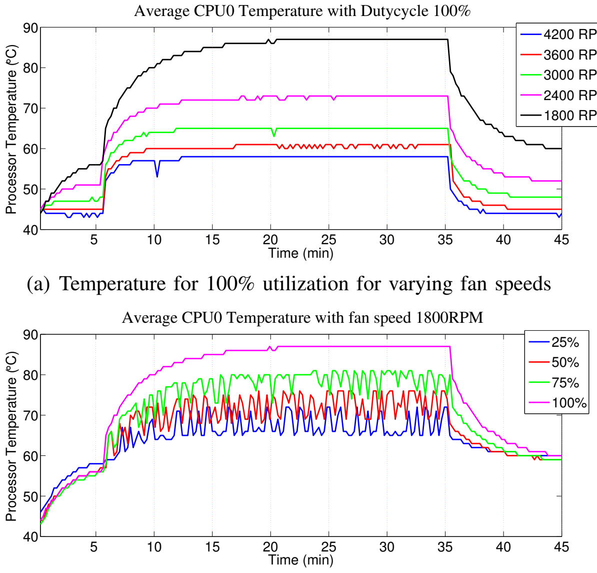
IV. Leakage and Temperature Tradeoffs

Leakage-temperature tradeoffs can have a high impact on the energy efficiency of enterprise servers. To evaluate them, we first perform a series of experiments. All experiments take place under the same conditions as follows: (i) the server is in an isolated environment at an ambient temperature of 24°C ; (ii) the machine always starts execution from a cold state that has been previously forced by at least 10 minutes of idle execution with fans rotating at 3600RPM; (iii) at the beginning of the execution (i.e., $t = 0$), fan speed is set to the appropriate value, and the machine is idle for another 5 minutes to allow temperature stabilization; (iv) the last 10 minutes of the experiments are always conducted with the CPUs idle, to let temperature drop to a steady state. These conditions are selected so that experiments reflect realistic working conditions and isolate the thermal-energy issues that we want to study.

We first perform experiments to gather data at varying utilization levels and fan speeds. Data are used to derive the various contributors to the server power consumption. Once all contributions are isolated, we can evaluate the leakage-temperature tradeoffs in the server.

Exploring System Dynamics:

In order to explore the system dynamics, we run *LoadGen* for 30 minutes at various utilization levels (10%, 25%, 40%, 50%, 60%, 75%, 90% and 100%) with different fan speeds (1800RPM, 2400RPM, 3000RPM, 3600RPM and 4200RPM). In this case, we set the same fan speed for all three pairs of fans. Figure 1(a) shows the CPU0 temperature under 100% utilization and all the fan speeds. These experiments show interesting results for both the transient and the steady state. We observe significantly different time constants depending on the fan speed. For 1800RPM the steady state is reached after 15 minutes of execution, whereas for the 4200RPM case, steady state is achieved after only 5 minutes. The magnitude of the thermal time constant is important for designing control mechanisms. The lower the fan speed, the slower the temperature reaction, which leaves more time for control decisions. However, temperature reacts much faster for the 4200RPM case, so a controller should be able to adapt faster. In addition, the considerable change in thermal time constants indicate that thermal models/predictors based on chip thermal modeling would need to take fan speeds into account to ensure accuracy in real-life settings.



(a) Temperature for 100% utilization for varying fan speeds

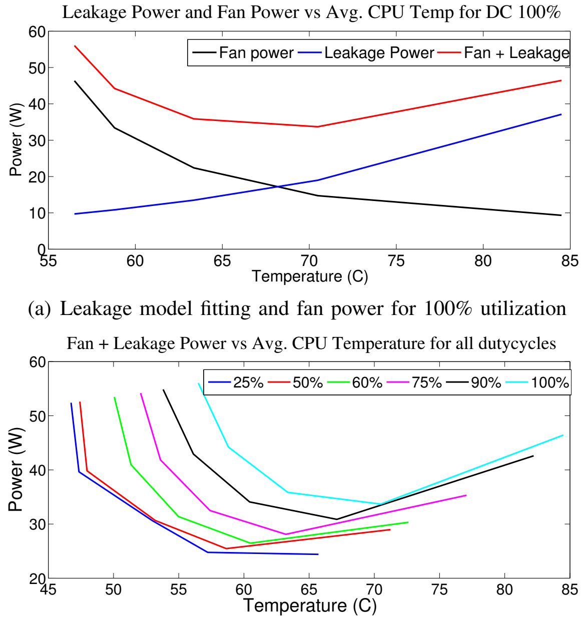
(b) Temperature for varying utilization levels at 1800RPM

Fig. 1. Processor temperature with different fan speed and utilization

Figure 1(b) shows the temperature at different workload utilization levels using a fan speed of 1800RPM. Thermal oscillations occur as *LoadGen* uses PWM to achieve a desired level of utilization. This plot shows the two transient temperature trends: a fast trend that raises the CPU temperature by 5°C to 8° in less than 30 seconds due to workload changes (from idle to high utilization), and the slow temperature increase taking up to 15 minutes due to the time constants.

Leakage Model Fitting:

The results discussed above are key to analyze the different contributors to the server power consumption and to derive an empirical leakage model as in Eqn.(1). In our setup, we can separately measure and control the fan power P_{fan}. By monitoring the power sensors, we are also able to measure the sum of leakage power P_{leak} and active power P_{active} of the sever. As *LoadGen* stresses the system at different levels of utilization (U) using the same workload, we can describe active power as a function of U. Leakage has an exponential dependence on temperature T as shown in Eqn.(2), where



(a) Leakage model fitting and fan power for 100% utilization

(b) Fan and leakage power for various utilization values

Fig. 2. Leakage power model fitting results

C is a constant. Using all the measurements of power and temperature at a set of utilization values, we apply model fitting techniques to derive the constant values k_1, k_2, k_3.

$$P_{total} = P_{active} + P_{leak} + P_{fan} \tag{1}$$

$$P_{active} = k_1 \cdot U \quad and \quad P_{leak} = C + k_2 \cdot e^{k_3 \cdot T} \tag{2}$$

For the equations above, we obtain the following parameters from the fitting: $k1 = 0.4452, k2 = 0.3231, k3 = 0.04749$, with a fitting error of only $2.243W$ and an accuracy of 98%. This fitting gives an analytical model that is valid across all utilization values. Figure 2(a) shows the fan power, the leakage power, and their sum for 100% utilization. The sum of leakage and fan power is a convex-like curve that reaches a minimum around 70°C, which corresponds to a fan speed of 2400RPM. In Figure 2(b) we show that a similar trend is observed for other utilization levels. Thus, for each U, there is an optimum fan speed that can be used to minimize the energy consumption. Note that for all the optimum points, average temperature is never higher than 70°C. Even though the server critical temperature threshold is set at 90°C, for reliability purposes [7] we target a maximum operational temperature of 75°C. Power savings achieved only by setting the appropriate fan speed can reach 30W for our server.

V. Temperature and Leakage-aware Cooling Control

Based on the model fitting results we generate a Lookup Table (LUT) that holds the optimum fan speed values for each utilization level. The goal is to save energy by setting the optimum fan speed at runtime as workload utilization varies. This section describes the LUT-based controller to automatically adjust the fan speeds of the server and compares the controller's efficiency against several baseline cases for a set of test workload profiles.

We use four different benchmarks of 80 minutes of total duration to test our controllers: (i) Test-1 ramps up and down from 0% to 100% utilization to test how controller reacts to gradual changes in utilization; (ii) Test-2 generates different periods (5, 10 and 15 minutes) between high and low utilization values to test controller reaction against sudden changes; (iii) Test-3 changes utilization values every 5 minutes to test reaction against sudden and frequent changes in utilization; and (iv) in Test-4 utilization value follows a statistical distribution of Poisson arrival times and exponential service times that emulates a shell workload as described in prior work [8].

The LUT-Based Controller:

To determine the optimum fan speed setting that achieves the minimum energy consumption at runtime, we propose a LUT-based controller. This controller is installed in the DLC-PC, which periodically monitors the load utilization through *sar* and *mpstat* utilities. Based on the LUT output, DLC-PC then sets the fan speed to the appropriate value by increasing or decreasing the current of the power supplies. Utilization is polled every second to be able to respond to sudden utilization spikes. Polling the utilization does not introduce any noticeable overhead on the CPUs. The controller makes decisions based on changes in the load utilization rather than reacting to temperature changes, which allows the system to proactively set the optimum fan speed before a thermal event occurs.

In order to ensure the stability of the controller and to prevent fan reliability issues in the case of unstable workloads, we set a maximum frequency for the fan speed changes. We allow the controller to react fast (i.e., change fan speed as soon as a spike is detected); however, we do not allow RPM changes for 1 minute after each RPM update. This 1-minute value is a tradeoff between the maximum number of fan changes allowed during the execution of a highly variable workload and the maximum temperature overshoot we want to tolerate in our system. Note that 1-minute is a safe choice for our system considering the large thermal time constants.

The Bang-Bang Controller:

The bang-bang controller only tracks the temperature using the CSTH and tries to maintain the temperature in between two desirable temperature values, 65°C-75°C, without using load utilization information. Our bang-bang controller has 5 different actions: (i) if maximum temperature T_{max} goes below 60°C, fan speed is set to 1800RPM (lowest); (ii) if T_{max} is in between 60°C to 65°C, fan speed is lowered by 600RPM; (iii) if T_{max} is between 65 to 75 degrees, no action is taken; (iv) if T_{max} is above 75°C, fan speed is increased by 600RPM; and, (v) if T_{max} is above 80°C, fan speed is increased to 4200RPM. Smaller target temperature ranges (e.g., 70°C-75°C) increase fan speed change frequency whereas larger ranges (e.g., 60°C-75°C) create higher temperature overshoots and undershoots, which can lead to higher fan speeds and larger thermal cycles. The threshold values are experimentally chosen to optimize this tradeoff, ensuring the stability of the controller while keeping temperature in a range that ensures high reliability. As the time between two consecutive actions of the controller is longer than the time it takes for the temperature values to cross thresholds, no additional maximum frequency change control (as in the LUT controller) is required.

Discussion of Controller Properties:

Table I summarizes the results for all the controllers for

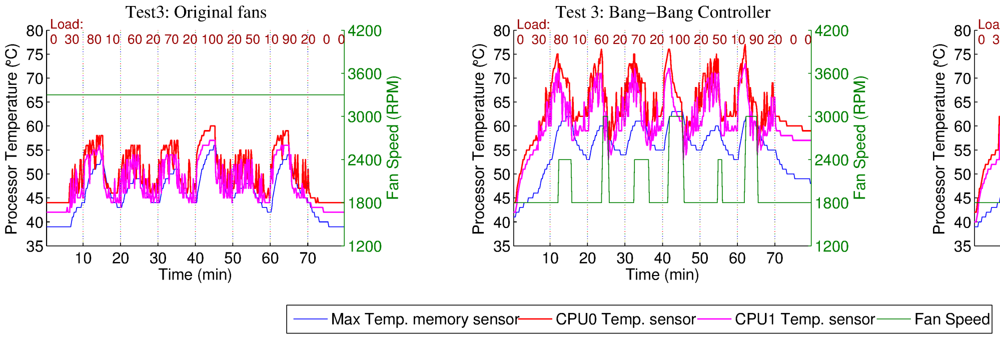


Fig. 3. Temperature sensor readings in Test-3 for the three different controllers.

Test	Control scheme	Energy (kWh)	Net Savings (kWh)	Peak Pwr (W)	Max. Temp (°C)	#fan change	Avg RPM
1	Default	0.6695	–	710	61	0	3300
	Bang	0.6570	6.8%	715	75	6	2089
	LUT	0.6556	7.7%	705	73	6	2117
2	Default	0.6857	–	720	61	0	3300
	Bang	0.6856	0.05%	722	76	10	2173
	LUT	0.6685	8.7%	705	75	8	2181
3	Default	0.6284	–	720	60	0	3300
	Bang	0.6253	2.0%	722	77	14	2042
	LUT	0.6226	3.9%	710	69	12	2161
4	Default	0.6160	–	720	62	0	3300
	Bang	0.6101	4.7%	722	76	10	1936
	LUT	0.6071	6.9%	710	74	12	1968

TABLE I. SUMMARY OF CONTROLLER PROPERTIES

all the tests. We use the default behavior of the server as the baseline. For all the tests, the baseline setting keeps the fans rotating close to a fixed speed of 3300RPM, which leads to very low temperatures and to overcooling of the system. Note that setting a high minimum RPM is common in commercial servers to ensure reliable operation under a wider range of ambient and altitude settings. Both bang-bang and LUT controller provide energy savings in comparison to the original fan control scheme. However, in some cases such as Test-2, the improvement of bang-bang controller is small. This is because the controller reacts after a thermal event occurs, leading to high average temperatures for the case of spiky loads, increasing leakage power. LUT-based controller reacts rapidly to workload changes and keeps average temperature lower, resulting in the lowest energy across the tests. Net energy savings are computed by subtracting the total server idle energy from the energy values (3^{rd} column) and comparing each of our controllers against the baseline. We discard the idle server power as that part of the consumption is dependent on the server hardware configuration and cannot be influenced by the fan control. The LUT-based controller achieves up to 8.7% energy savings and 25W peak power reduction compared to the baseline. It also keeps temperature under 75°C using a low number of fan speed changes.

Figure 3 compares the runtime behavior of the three controllers for Test-3. As expected, the default fan controller keeps temperature very low with a fan speed of 3300RPM. The bang-bang controller addresses the over-cooling in the baseline case by letting the temperature rise but keeping it in between the 55°-75°range. The bang-bang controller is similar to existing fan controllers in commercial servers but it allows higher temperatures. As a result, bang-bang controller generates temperature spikes and higher oscillations. The LUT controller changes fan speed according to utilization to minimize power. Even though it does not monitor temperature, the runtime temperature values are lower and more steady, so leakage is always kept low. In this test, LUT controller only needs to change the RPM between two different fan speeds because the machine is in a colder environment compared to the ambient of a data center.

VI. Conclusions

Reducing the energy consumption of enterprise servers in data centers continues to be a major challenge. As technology nodes shrink, leakage becomes an important contributor to the overall power consumption. This paper has presented an experimental methodology to explore the effect of leakage-temperature tradeoffs on the energy efficiency of enterprise servers. By utilizing a stress test, we derived an analytical model for leakage power and computed the optimum fan speeds for different utilization values. Based on our analysis, we implemented a LUT-based cooling controller that adjusts the fan speed of the system to the optimum value during runtime. Our controller provides energy savings of up to 9% and decreases the peak power by 30W. Our technique can be extended to real-life workloads by using a larger set of performance counters to characterize runtime dynamics and applying statistical analysis to derive energy-performance models for different classes of applications.

Acknowledgment

This research has been partially supported by a PICATA predoctoral fellowship of the Moncloa Campus of International Excellence (UCM-UPM), the Spanish Ministry of Economy and Competitivity under research grant TEC2012-33892, and by Oracle, Inc.

References

[1] J. Koomey, "Growth in data center electricity use 2005 to 2010," Analytics Press, Oakland, CA, Tech. Rep., 2011.

[2] M. Patterson, "The effect of data center temperature on energy efficiency," in *ITHERM*, 2008, pp. 1167 –1174.

[3] K. Gross, K. Whisnant, and A. Urmanov, "Electronic prognostics through continuous system telemetry," in *MFPT*, April 2006, pp. 53–62.

[4] X. Han and Y. Joshi, "Energy reduction in server cooling via real time thermal control," in *SEMI-THERM*, 2012, pp. 20 –27.

[5] D. Shin, J. Kim, N. Chang, J. Choi, S. W. Chung, and E.-Y. Chung, "Energy-optimal dynamic thermal management for green computing," in *ICCAD*, 2009, pp. 652–657.

[6] Z. Wang, C. Bash, N. Tolia, M. Marwah, X. Zhu, and P. Ranganathan, "Optimal fan speed control for thermal management of servers," in *InterPACK*, 2009.

[7] D. Atienza *et al.*, "Reliability-aware design for nanometer-scale devices," in *ASPDAC*, 2008, pp. 549–554.

[8] D. Meisner and T. F. Wenisch, "Stochastic queuing simulation for data center workloads," in *EXERT*, 2010.