CALIFORNIA ENERGY COMMISSION

CALIFORNIA NATURAL RESOURCES AGENCY

ENERGY RESEARCH AND DEVELOPMENT DIVISION

FINAL PROJECT REPORT

Enabling Energy Efficient Data Center in Smart Power Distribution Systems

May 2024| CEC-500-2024-035

PREPARED BY:

Nanpeng Yu
Daniel Wong
Hyeran Jeon
University of California, Riverside
Primary Authors

Christian Fredericks
Project Manager
California Energy Commission

Agreement Number: EPC-16-030

Cody Taylor
Branch Manager
INDUSTRY & CARBON MANAGEMENT BRANCH

Jonah Steinbuck, Ph.D.
Director
ENERGY RESEARCH AND DEVELOPMENT DIVISION

Drew Bohan
Executive Director

ACKNOWLEDGEMENTS

The authors thank the California Energy Commission for the financial support. The helpful comments and suggestions from technical advisors of the California Energy Commission, Southern California Edison Company, Pacific Gas and Electric Company, and Data Center Solutions Providers greatly improved the quality of the research and development.

PREFACE

The California Energy Commission's (CEC) Energy Research and Development Division supports energy research and development programs to spur innovation in energy efficiency, renewable energy and advanced clean generation, energy-related environmental protection, energy transmission, and distribution and transportation.

In 2012, the Electric Program Investment Charge (EPIC) was established by the California Public Utilities Commission to fund public investments in research to create and advance new energy solutions, foster regional innovation, and bring ideas from the lab to the marketplace. The EPIC Program is funded by California utility customers under the auspices of the California Public Utilities Commission. The CEC and the state's three largest investor-owned utilities—Pacific Gas and Electric Company, San Diego Gas and Electric Company, and Southern California Edison Company—were selected to administer the EPIC funds and advance novel technologies, tools, and strategies that provide benefits to their electric ratepayers.

The CEC is committed to ensuring public participation in its research and development programs that promote greater reliability, lower costs, and increase safety for the California electric ratepayer and include:

- Providing societal benefits.
- Reducing greenhouse gas emission in the electricity sector at the lowest possible cost.
- Supporting California's loading order to meet energy needs first with energy efficiency and demand response, next with renewable energy (distributed generation and utility scale), and finally with clean, conventional electricity supply.
- Supporting low-emission vehicles and transportation.
- Providing economic development.
- Using ratepayer funds efficiently.

For more information about the Energy Research and Development Division, please visit the CEC's research website (www.energy.ca.gov/research/) or contact the Energy Research and Development Division at ERDD@energy.ca.gov.

ABSTRACT

The goal of this project was to improve energy efficiency and enable demand response for data centers in smart power distribution systems. This project achieved two objectives: 1) it developed pre-commercial server, data center, and data center cluster energy efficiency technologies and strategies, and 2) it provided easily accessible software solutions to facilitate the adoption of energy efficient data center technologies.

Technological and scientific advancements were achieved at three levels. At the server level, an innovative low power management system was developed that coordinates deep sleep states and dynamic voltage-frequency scaling and selects the optimal power state configuration for a given workload and traffic pattern. At the rack/data center level, a new workload scheduling algorithm was developed to improve the data center level energy efficiency. This new algorithm collects system statistics of worker servers to predict power levels and trigger load migration to require all servers to run at peak energy efficiency. At the data center level, the project team developed a solution to enable data centers to provide ancillary services to the electricity market by adjusting their energy consumption.

If all data centers in California adopt the three technologies developed in this project, it could result in estimated annual electricity savings of 1,342 gigawatt hours, a corresponding cost reduction of $163 million, and a greenhouse gas emission reduction of 596,114 metric tons. The lessons learned from this project are being incorporated into IEEE Standards P1924.1.

Keywords: data center, energy efficiency, demand response, ancillary services

Please use the following citation for this report:

Yu, Nanpeng, Daniel Wong, and Hyeran Jeon. 2020. *Enabling Energy Efficient Data Center in Smart Power Distribution Systems*. California Energy Commission. Publication Number: CEC-500-2024-035.

TABLE OF CONTENTS

Acknowledgements i

Preface ii

Abstract iii

Executive Summary 1

- Background 1
- Project Purpose 1
- Project Approach 1
- Project Results 2
- Technology/Knowledge Transfer/Market Adoption 3
- Benefits to California 4

CHAPTER 1: Introduction 5

CHAPTER 2: Project Approach 8

- Improve Server Level Energy Efficiency 8
  - Overall Framework 9
  - Technical Methods 10
- Improve Data Center Level Energy Efficiency 13
  - Overall Framework 13
  - Technical Methods 14
- Improve Data Center Cluster Energy Efficiency 17
  - Overall Framework 18
  - Technical Methods 18
- Transmission System Operator 20
  - Performance-based Compensation 20

CHAPTER 3: Project Results 21

- Improvement in Server Level Energy Efficiency 21
  - Evaluation Setup 21
  - Performance of Server Low Power Management 21
- Improvement in Data Center Level Energy Efficiency 23
  - Simulation Setup 23
  - Performance of DNN-accelerated Load Scheduling Algorithm 24
- Improvement in Data Center Cluster Energy Efficiency 27
  - Simulation Setup 27
  - Performance of Frequency Regulation Service Provision by Data Center 27
- Summary 30

CHAPTER 4: Technology/Knowledge/Market Transfer Activities 31

- Target Market – Enterprise Data Centers 31
- Overall Strategy and Transfer Activities 32

CHAPTER 5: Conclusions/Recommendations 36
CHAPTER 6: Benefits to Ratepayers 37
Importance and Benefits to Ratepayers 37
Monetary, Energy, and Emission Savings for Ratepayers 37
Potential for Technology Adoption 38
Potential Societal Benefits to Ratepayers 38
Glossary and List of Acronyms 39
References 40

LIST OF FIGURES

Figure 1: Project Framework 7
Figure 2: Existing Dynamic Power Management 9
Figure 3: μDPM 9
Figure 4: μDPM Run-time Illustrative Example 10
Figure 5: Overall Framework of Workload Scheduling Algorithm for Data Center Level Energy Efficiency 13
Figure 6: Server Power Prediction Errors of Conventional Power Models With Application Memory Intensity 15
Figure 7: Actual vs. Predicted Power 17
Figure 8: Overall Framework of Frequency Regulation Service Provision by Data Center 18
Figure 9: Prices for Frequency Regulation Services and Energy in PJM Market 19
Figure 10: Tail Latency Under Varying Traffic Loan 22
Figure 11: Energy Saving Comparisons Among Different Power Management Schemes 23
Figure 12: Mean Power on Two Servers with Respect to Migration Triggering Point 24
Figure 13: Example Migration Result: Media and Graph 25
Figure 14: Example Migration Result: Red Color Line 25
Figure 15: Example Migration Result: RNN-DNN, Graph, Media, Web Search 26
Figure 16: Example Migration Result: Red Color Trend 26
Figure 17: Fitted Power Consumption Curves With Default Sleep Policy 27
Figure 18: Hourly-averaged Request Arrival Rate After Scaling 28
Figure 19: Frequency Regulation Signal Following One Hour for Three Pages 29
Figure 20: Distribution of Requests Response Time 30

LIST OF TABLES

Table 1: System Statistics Considered for Power Prediction 15
Table 2: Frequency Regulation Signal Following Performance Scores 28
Table 3: Electricity Costs of the Data Center Under Two Operating Scenarios 29

Executive Summary

Background

Data centers currently consume approximately 5,580 gigawatt hours (GWh) per year of electricity in California (2 percent of California's 2019 electricity demand). By 2030, data centers are expected to be responsible for 7 percent of global carbon emissions (Andrae & Edler, 2015). Without additional energy efficiency technologies, electricity consumption from data centers is expected to double in the next 10 years.

Technologies developed in this project improve energy efficiency at the server, rack, data center, and data center cluster level of data centers. In addition, they provide demand response potential, helping to mitigate the intermittency of renewable energy resources while giving data center operators additional revenue streams.

Project Purpose

When someone posts a tweet, shares a post on Facebook, sends an email, or searches a business on Google, requests are routed by data centers, consuming electricity. As the U.S. and California economies continue to digitalize, electricity customers need to be concerned about energy efficiency at data centers, which are the engines of the digital life and economy.

The intended audiences of this project's research and development products are data center owners and operators. On-site data centers would gain the highest percentage of energy efficiency improvement. A pre-commercial server and a data center were developed to demonstrate improved energy efficiency and enable demand response. The project has shown that energy efficiency can be improved:

- at the server level, by coordinating deep sleep states and optimizing the power state according to a given workload and traffic pattern by dynamic voltage-frequency scaling.
- at the rack/data center level, by migrating the computational load between servers and between racks to operate servers at peak efficiency and to balance a three-phase load.
- at the system level, by migrating the computational load between data centers according to conditions of the electric distribution network.

In addition, this project quantified the capabilities of data centers to participate in demand response.

Project Approach

This project is the first study that applies deep neural networks (DNN) for current/future power prediction for data center servers. With not many publications on the topic, the team used studies from other fields (such as DNN-based object detection, stock-price prediction, and so on) and similar approaches to develop a unique design that fits well with data center study. Under the guidance of principal investigator, the team of students learned how to use

tools like DNN frameworks, Linux system monitoring commands, and a Docker container with a swarm feature to do research necessary to reach project objectives.

Processor low power modes available in hardware may not be accessible and usable by users until the operating system includes software support to expose the hardware features to the software. Sometimes it takes years before processor manufacturers implement software support in the operating system. To account for this, the team designed server-level low power management policies based on various assumptions of processor feature availability.

To minimize costs, the project team designed a system with open-source software tools and packages that were compatible with existing hardware. A set of performance metrics, including throughput and latency, was used to compare results against the baseline performance and to ensure the server met or exceeded the standards. Power conception was monitored and compared to the performance metrics to make sure the project achieved greater efficiency.

Project Results

The project team achieved all the objectives and goals laid out in the project proposal, providing electrical cost savings of 71 percent at the server level, 25 percent at rack/datacenter level, and about 8 percent at the datacenter cluster level.

- **Server level:** The project team developed POWERMORPH, the server level low power management system, which enables data centers to participate in regulation service by dynamically adjusting the server power consumption. Based on available literature, POWERMORPH is the first proof-of-concept demonstration of frequency regulation service in realistic data center environments.
- **Rack/datacenter level:** The project team evaluated data center load distribution strategies. They observed that the peak-energy-efficiency-aware load balancing algorithm that forces all servers to stay at their peak energy efficiency levels saves more energy than the conventional packing or uniform distribution algorithm. This is especially true for high-energy-proportional[1] servers. To proactively balance loads and have more servers stay at their peak energy efficiency level, the team developed a DNN-accelerated server power model. The DNN-accelerated power model provides superior prediction accuracy (about 96 percent) by considering tens of system parameters, compared to conventional models that mostly rely on CPU use. The load balancing algorithm that incorporates the DNN-accelerated power model trigger for load migration demonstrated 25 percent electricity cost savings for a data center.
- **Data center cluster level:** The operational flexibility of the data centers can be leveraged to provide valuable frequency regulation services in the smart grid. A comprehensive frequency regulation service provision framework was developed in this project. A risk constrained hour-ahead bidding strategy and a real-time frequency regulation signal following algorithm were developed. The introduction of a dummy load and the realistic server power consumption model allows data centers to follow real-

[1] In an energy-proportional server, the power consumption of the server is proportional to its load.

world frequency regulation signals with over 95 percent accuracy. A numerical study with Wikipedia's access trace shows that, with reliable energy and frequency regulation service price forecast, data centers can reduce their electricity bill by about 8 percent without violating service level agreements.

In this project, the team applied and evaluated the management algorithms and prototype implementation in a small-scale data center. Follow-up research should include:

1. Evaluating the design for large-scale data centers that have highly diverse computing elements with different energy proportionality and computing capabilities.
2. Updating the DNN model to accommodate up-to-date server statistics.
3. Investing the impact of emerging data center hardware trends such as wider adoption of computing accelerators (GPUs, FPGAs, etc.) and ARM processors.
4. Evaluating the impact of a wider-range of data center workloads, including emerging microservice workloads and machine learning (ML) or artificial intelligence (AI) workloads.

Technology/Knowledge Transfer/Market Adoption

The project released an open-source software package to the data center industry. Commercial entities such as data center solution providers and startup companies can leverage the freely available software to build commercial products for large-scale adoption. The team expects higher attention from the community after it finishes presenting the project results at conferences and in published papers. This project highlights the need to improve energy efficiency for data centers and provide open-source solutions to the data center industry to improve energy efficiency.

The team published five academic papers in technical journals and presented at two industry and academic conferences. In addition, team members drafted the standard with the Institute of Electrical and Electronics Engineers (IEEE) outlining recommended practices for this technology.

The project team presented results to several data center industry managers and operators. They plan to adopt the results of this project to improve their data centers' energy efficiency. The near- term target markets for the technology are the on-site data centers in California. The mid-term target market expands to all types of data centers in California. The long-term target market includes all data centers in the U.S. The industry advisors provided useful feedback about making the products easier to use and adopt. It is challenging to track facilities that actually adopted the research results, but citations of the team’s publications show adoption and extension of the research outcomes to some extent. The team’s publications have been cited by researchers of the National Renewable Energy Laboratory, Pacific Northwest National Laboratory, the University of Colorado, Boulder, Google, and Stanford University.

Benefits to California

The proposed technology improvements in this project are estimated to reduce data center energy use by 48 percent annually. If all data centers in California adopted the proposed technology, an estimated annual electricity cost savings of $163 million could be realized.

The energy efficiency technologies developed in this project will reduce the data center energy consumption at the server, rack, and data center system levels by 7.8 percent, 25 percent, and 25 percent, respectively. If all data centers in California adopted the technology developed in this project, an estimated annual energy saving of 1,342 GWh, and a corresponding reduction of 596,114 metric tons of greenhouse gas emissions will be realized. Servers consume roughly 50 percent of data center energy or 2,790 GWh. Assuming a commercial electricity price of $0.20 per kilowatt hour (kWh), the electricity cost of the data center servers is $558 million. Therefore, an estimated energy savings of $163 million is anticipated, with an estimated total avoided electricity production of 1,342 GWh.

CHAPTER 1: Introduction

Data centers currently consume approximately 5,580 GWh per year of electricity in California (2% of California's 2019 electricity demand). By 2030, data centers are expected to be responsible for 7 percent of global carbon emissions (Andrae & Edler, 2015). Without additional energy efficiency technology, electricity consumption from data centers is expected to double in the next 10 years.

To contribute and meet California's goal of reducing greenhouse gas (GHG) emissions, data centers must employ more energy efficient techniques. However, a major barrier to achieving energy efficient data centers involves server underuse and poor low power management policies across different levels in the datacenter. By evaluating techniques to improve data center energy efficiency at the server level, rack/cluster level, and data center level, the technologies developed during this project can reduce the energy consumption of and electricity cost to ratepayers. In addition, through the participation of ancillary services, the project technologies can help improve the reliability of power grids.

To combat energy inefficiencies resulting from underuse and poor idle power consumption, a common approach is to consolidate workloads to a subset of active servers in the cluster and turn off idle servers. Active servers run at a higher utilization, which is more energy efficient. Meanwhile idle servers, which still consume up to 60 percent of peak power, are shut down, saving significant wasteful idle power. Many workloads in the data centers cannot tolerate servers being turned off due to performance requirements. Therefore, servers must rely on processor sleep states and dynamic voltage frequency scaling (DVFS). However, sleep states and DVFS face many limitations. Due to silicon technology scaling, DVFS alone has limited effectiveness because of limitations to how low voltage can scale. Sleep states are not used effectively due to unpredictable short idle periods, leading to energy inefficient shallow sleep states. To solve this problem, the project team developed a server level coordination technology to achieve longer idle periods for effective deep sleep states and extend the amount of time DVFS can be in a low frequency state.

Additionally, workload consolidation tends to pack servers to their peak utilization, based on the conventional wisdom that peak energy efficiency occurs at maximum server utilization. However, as data center server energy efficiency improves, this trend no longer holds. Modern servers exhibit peak energy efficiency between 50 percent to 60 percent utilization. Thus, it is more energy efficient to balance loads across servers than consolidating workloads to fewer servers. During this project, the team developed and tested a new deep-learning-based load management and load migration algorithm for energy efficiency. Most of the previous studies used heuristic approaches that incorporated hardware resource use and inter-application dependencies to determine the target server to migrate applications. Instead of using heuristics, the team used deep learning to determine the optimal migration that guarantees peak energy efficiency for all servers.

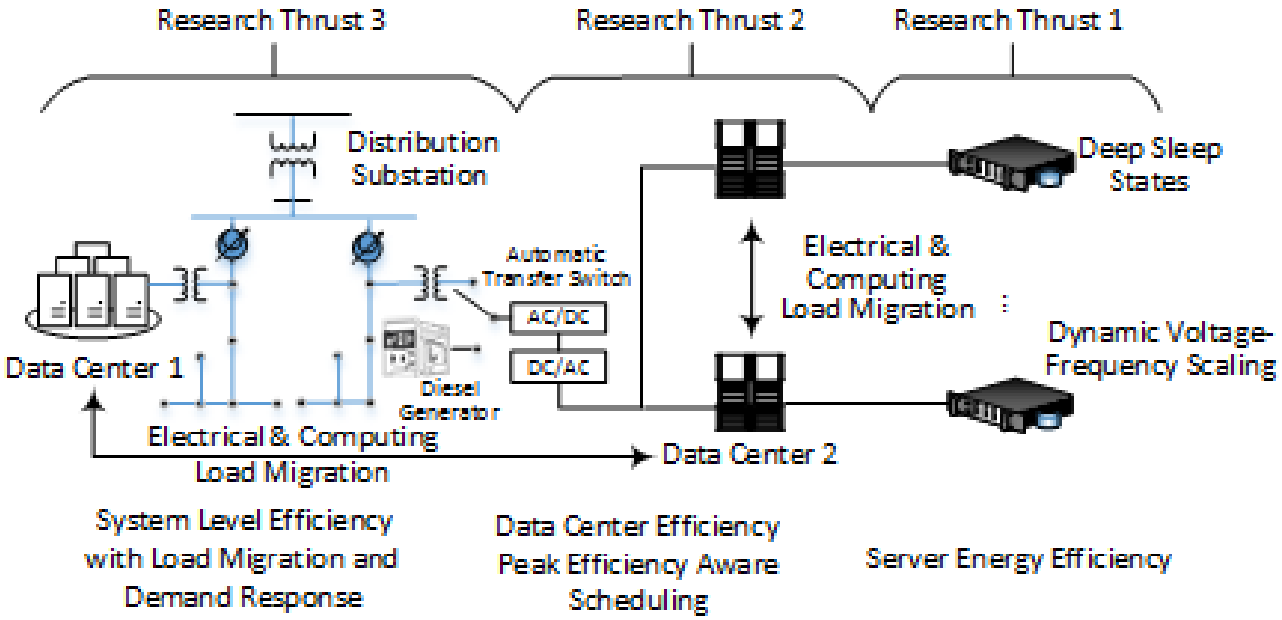
Previous research on data center energy efficiency focused on reducing the power consumption of a single data center. To improve data center energy efficiency, hardware and software layer improvements were proposed. On the hardware layer, device level chip multiprocessing, core parking, memory bank parking, and new cooling technologies were developed. On the software layer, virtualization, server consolidation, and application placement techniques were developed. In many of these prior data-center-level techniques, data centers passively react to varying electricity prices and load reduction instructions from the electric utility companies. During this project, the team developed a proactive data center demand participation scheme. This scheme created price-sensitive demand bids and ancillary service bids based on customer request forecasts, quality of service, electric load forecast, and price forecast. This approach makes it possible to integrate data centers into the smart distribution systems seamlessly and to fully integrate them into the resource dispatch and price formation processes.

The goal of this project was to improve energy efficiency and enable demand response for data centers in smart power distribution systems. The objectives of the project were to 1) develop pre-commercial server, data center, and data center cluster energy efficiency technologies and strategies and 2) facilitate the adoption of data center energy efficiency technology by providing easily accessed software solutions. To achieve the energy efficiency and demand response goals, the project advanced three areas/levels.

- At the server level, the team improved energy efficiency through coordinated deep sleep states and DVFS, which selects the optimal power state configuration for a given workload and traffic pattern.
- At the rack/data center level, the team migrated computational load among servers to operate servers at peak efficiency points and to migrate workload across racks to balance the electrical loads across three phases in the electric power distribution systems.
- At the system level, the team improved the electricity cost efficiency of data center clusters through participation in demand response and frequency regulation ancillary service.

The overall project framework is depicted in Figure 1.

Figure 1: Project Framework



Source: UC Riverside

CHAPTER 2:
Project Approach

Improve Server Level Energy Efficiency

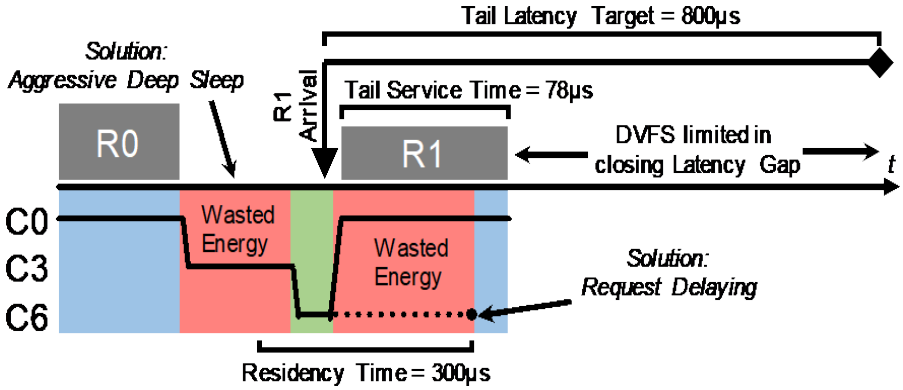
Server low power management is hindered by three main issues, illustrated in Figure 2. This illustrative example shows two requests, R0 and R1, along with the sleep state at which the processor is operating. Sleep states are commonly called C-states, with C0 indicating the processor is active, C3 being a shallow sleep state that consumes less power, and C6 being a deep sleep state that consumes near-zero power. Consider the case where R0 is processing and then completes the task, entering an idle period.

First, idle periods are not handled effectively due to inefficiencies in selecting sleep states. For example, in the Linux operating system, the idle governor either estimates the idle period length and selects the best state for that idle period, or it enters the shallowest state and moves to a deeper state if the processor remains in a long enough sleep state. Figure 2 illustrates the latter policy, where the processor first enters C3, then after a while enters C6. If idle periods are short, as is common with short request service times, governors will consistently select shallow sleep states. It is possible for the Linux governor to mispredict the expected idle period length and select a shallower sleep state than is supported. These shallower sleep states lead to wasted power when a server initially enters an idle state. Such sleep state management behaviors are not unique to Linux; they are also present in other operating systems.

Second, it is possible for the server to wake up, due to a request arrival, before it has achieved its target residency time, which is the amount of time a processor needs to spend in a certain sleep state before the energy saved exceeds the energy overhead of entering a sleep state. For example, in Figure 2, request R1 arrives before the target residency time. The server wakes from sleep too early and ends up consuming more energy than it saves. Currently, server operating systems are not aware of the energy efficiency of low power states and, therefore, wake up processors to prioritize performance over power savings.

Third, sleep states are not coordinated with frequency states, so that, upon wakeup, the frequency governor typically selects the highest clock frequency and processes the request as fast as possible. This is wasteful, as the request does not have to be completed until a certain deadline. For example, in Figure 2, request R1's deadline for completing processing (also known as the tail latency target) is 800 microseconds (µs). However, the tail service time to process the request is only 78 µs. This leads to a waste of the DVFS policy, where request R1's processing time could have been slowed down to save power but was not. Furthermore, due to the limited range of frequency states, even if request R1 could be slowed down, it cannot be slowed down to the point where it completes just-in-time to meet the deadline and maximize the amount of time in a low frequency state. Uncoordinated sleep and DVFS can lead to significant inefficiencies.

Figure 2: Existing Dynamic Power Management



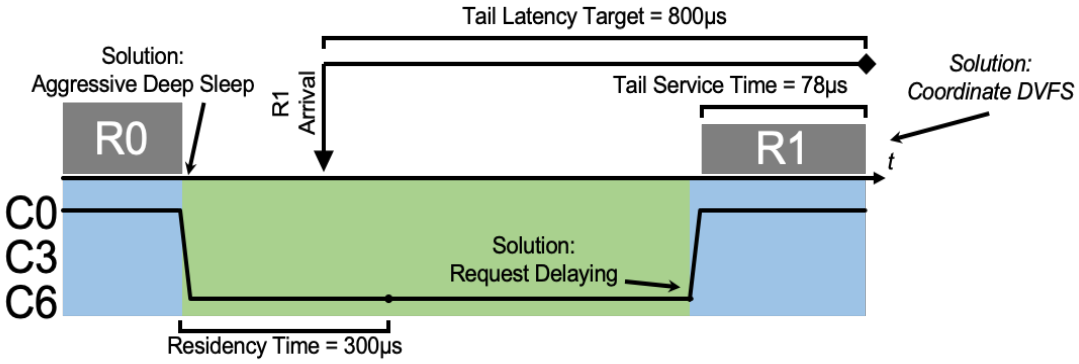
Existing dynamic power management limitations include inefficient sleep state management, limited range of frequency states, and significant low-power state transition overhead. This illustrative figure highlights these inefficiencies when an idle period exists between processing two requests, R0 and R1. Sleep states are commonly known as C-states and are shown as C0 (active), C3 (shallow sleep), and C6 (deep sleep).

Source: UC Riverside

Overall Framework

The overall framework for the server level low power management, called µDPM (micro dynamic power management), is shown in Figure 3. µDPM aggressively deep sleeps to minimize idle periods, delays wakeup to meet C-state target residency time, and coordinates frequency scaling to complete the request just-in-time to meet the target tail latency constraint. This is carried out by coordinating the operating system's sleep state driver and frequency state driver, and the application's tail latency requirements. The key insight driving this solution is that deep sleep modes can be beneficial if it is tail latency aware. The details of the mechanism are explained in the following subsections.

Figure 3: µDPM



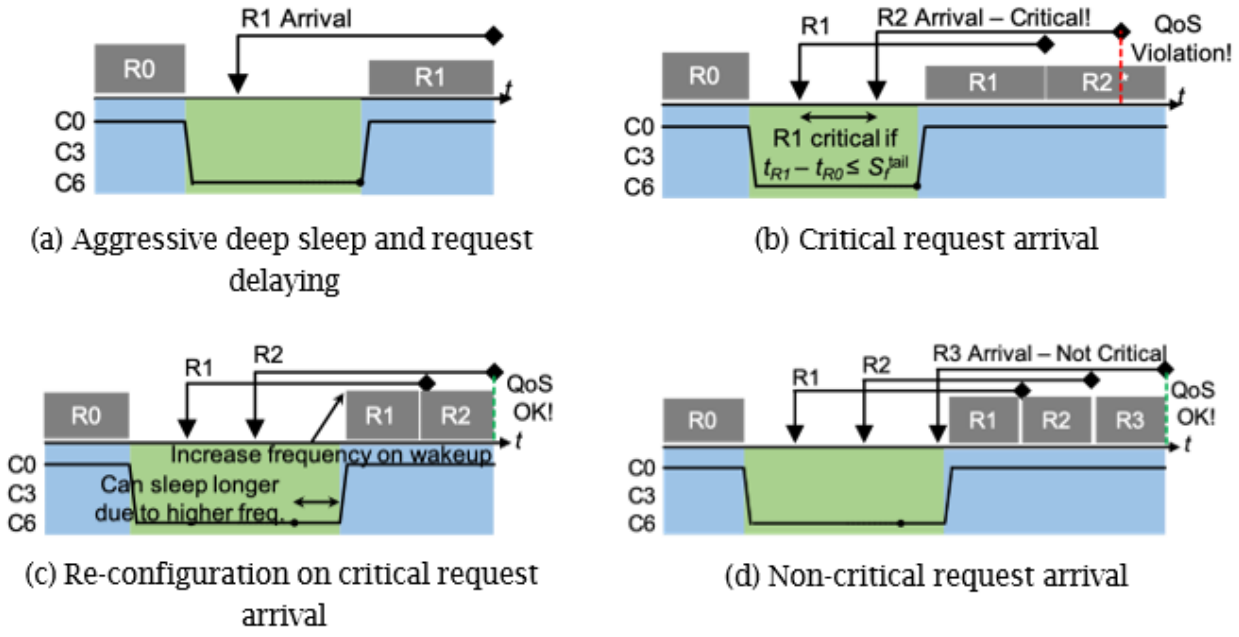
Careful coordination of aggressive deep sleep (eliminates idle power), request delaying (meets residency time), and DVFS (just-in-time target tail latency) can achieve energy savings in microsecond workloads.

Source: UC Riverside

Technical Methods

The technical methods of µDPM are illustrated in Figure 4. In Figure 4(a), the moment a core is idle (such as when all prior requests complete), µDPM immediately goes into the deepest sleep state (C6) to save idle power. µDPM needs to maintain (1) when to wake up, and (2) what frequency to run upon wakeup. These two parameters are referred as a *configuration*. Upon entering sleep, the configuration is reset with a null wake-up time (representing stay asleep indefinitely) and with the lowest frequency setting.

Figure 4: µDPM Run-time Illustrative Example



µDPM run-time illustrative example: (a) aggressively deep sleep when idle, and delay wakeup until the request can finish just-in-time at the lowest frequency setting; (b) arriving request R1 is a critical request and will miss the latency target; (c) increase frequency on wakeup to meet request R1's latency target; due to the higher frequency, we can sleep longer, as request R1 will complete more quickly; (d) a normative case with a non-critical request arrival.

Source: UC Riverside

When request R1 arrives, the tail service time of R1 is predicted while running at the current frequency configuration, and a wake-up time is set such that R1 finishes just-in-time. Instead of waking up the core at the latest moment and processing at full speed to meet tail latency targets, µDPM wakes up the core earlier and processes the request at a slower frequency to achieve a better trade-off between latency and power savings.

Figure 4(b) illustrates a scenario where a second request, R2, arrives during an idle period and is determined to violate quality of service (QoS) constraints given the current wake-up time and frequency configuration. We define a request that will violate QoS constraints as a *critical request*. Since we know that the previous request satisfies QoS constraints, we can detect a critical request by simply determining if the interarrival time between these two most recent requests is less than the predicted tail service time of the incoming request.

Whenever a critical request arrives, μDPM reconfigures the wake-up time and frequency configuration (Figure 4[c]). μDPM increases the frequency until R2 meets QoS. The increase in frequency is symbolized by the higher, but narrower, gray box of requests R1 and R2. Frequency is only increased, and not decreased, to limit DVFS transition overhead. Since frequency increased, R1 will complete more quickly, enabling μDPM to sleep longer, increasing the idle period length and still satisfying QoS.

Figure 4(d) shows a normative case where another arriving request, R2, is not critical. In this scenario, R2 is satisfied with the given wake-up time and frequency and will therefore simply queue. Also, if a critical request arrives during an active period, this simply triggers a frequency increase, as the wake-up time is void.

μDPM needs to determine: (1) when to wake up after sleeping and (2) what frequency to run at. The key is to estimate the incoming request's service time. μDPM uses a statistically based performance model and criticality-aware scheduling to recalculate wake-up time and frequency at every critical request arrival. In addition, μDPM will consider transition overheads while determining the optimal wake-up time and runtime frequency.

Estimating Request Tail Service Time: A statistical performance model is used to estimate the tail service time of processing and queued requests. At a high level, this model breaks down request processing into two probability distributions: *cycles* spent in computing, $P[C = c]$, and *time* spent memory-bound, $P[M = t]$. These probability distributions can be sampled online through performance counters for $P[C = c]$, and through cycles-per-instruction (CPI) stacks for $P[M = t]$. Because of the nondeterministic request demands, the service time of a request is often considered as a random variable. A single distribution for simplicity was used, trading off a small amount of power savings opportunity.

When multiple requests are in the queue, it is not sufficient to just estimate request tail service time. The completion time of the requests upon wakeup needs to be estimated. Therefore, the estimated completion cycle of a request Ri is a random variable S_i, with probability distribution $P[S_i = c]$. The completion cycle distributions all draw from a single distribution $P[S = c]$, where S is how many cycles it takes to process one request. S is essentially a combination of the compute cycle distribution, C, and memory time distribution, M; $S = C + M f$. A 95th percentile of the distributions is used to obtain the tail service request time.

The cycle at which Ri completes, $P[S_i = c]$, can then be computed as the n-fold convolution ($*$) of S , where n is the number of queued requests and processing requests. The model is simplified by not conditioning the currently processing request on elapsed cycles completed. For example, in Figure 4(d), the estimated completion cycle of R2 (the random variable S_2) is the sum of the random variables S_0, S_1, and S, and it is estimated as the following convolution: $P[S_2 = c] = P[S_0 = c] * P[S_1 = c] * P[S = c]$. The completion cycle can be simply converted to completion time by dividing the core's frequency.

Estimating Request Tail Latency: To determine whether a request is critical or not, the latency of a given request is estimated. The estimated tail latency of the request, L_i, is given as follows:

$$LL_i = WW + \underset{TT_{wwwwwwww}}{} + \underset{TT_{dddddddd}}{} + \frac{SS_{ii}}{dd} \quad (1)$$

where W is the time until the core is scheduled to wake up, T_{wake} is wake-up transition time, T_{dvfs} is the DVFS transition time, S_i is the estimated tail completion cycle to service request Ri as discussed prior, and f is the operating frequency. Based on this latency model, target tail latency, core frequency, and wake-up time are used to determine μDPM configurations.

Determining Critical Requests: After estimating the latency of arriving requests, it is necessary to check whether or not the arriving request is critical to determine whether a configuration update is needed. By observing Figure 4 and equation (1), a request is critical if:

$$tt_{RR_{ii}} - tt_{RR_{ii-1}} \leq \frac{SS^{ttttiitt}}{dd} \quad (2)$$

where t_R is the arrival time of a request R, and S is the completion cycle distribution for a single request (service only, no queueing). This is because the arrival time between the current and the previous requests is too short for processing one request. Intuitively, for a given wake-up/VFS configuration, when the processor wakes up, it can process a request every S_i seconds and meet the target tail latency just-in-time. If requests arrive too closely together, if the previous request is scheduled to finish just-in-time to meet the target tail latency, the current request will experience longer latency than the previous one, exceeding the tail latency. This insight is used to simplify the calculation for new wake-up time and frequency configurations.

Determining New Wake-Up Time and Frequency Configuration: A critical request occurs when the incoming request cannot meet QoS requirements and requires a two-step approach. The first step is to determine the new frequency. Conceptually, this can be illustrated as: What is the frequency required to squeeze R1 to fit between R0 and the red line? This can be achieved as:

$$TT_{RR_{ii}TTwwTTTTwwTTTTTTTTTTTTwwTTiiTTTT} - TT_{RR_{ii-1}TTTTTTTTTwwTTiiTTTT} - \frac{SS^{ttttiitt}}{dd'} \quad (3)$$

where $T_{R_i\ TargetCompletion}$ is the target tail latency completion time of Ri, $T_{R\ i\text{-}1\ Completion}$ is the completion time of R_{i-1}, S is the completion cycle distribution for a single request, and f' is the new frequency. Since all of these variables are known by the time a request is determined to be critical, the new frequency can be computed directly. To minimize DVFS transition overheads, frequency changes are limited to only increase and not decrease. The frequency would then reset to a lower level upon the next idle period.

The second step is to determine the wake-up time. If a critical request arrives during an active period, this step is not necessary. This can be achieved by essentially shifting all requests to the right, or as late as possible, while still satisfying latency constraints. This requires re-estimation of completion cycles for all queued requests. To determine the new wake-up time, the new completion time for the first queued request, S_0, is used in Equation (1). If the wake-up time is determined to be shorter than the residency time, μDPM will wake up at the cost of some energy overhead. Project experiments measured the overhead of this computation to be 2μs, which is trivial in comparison to the service time of requests.

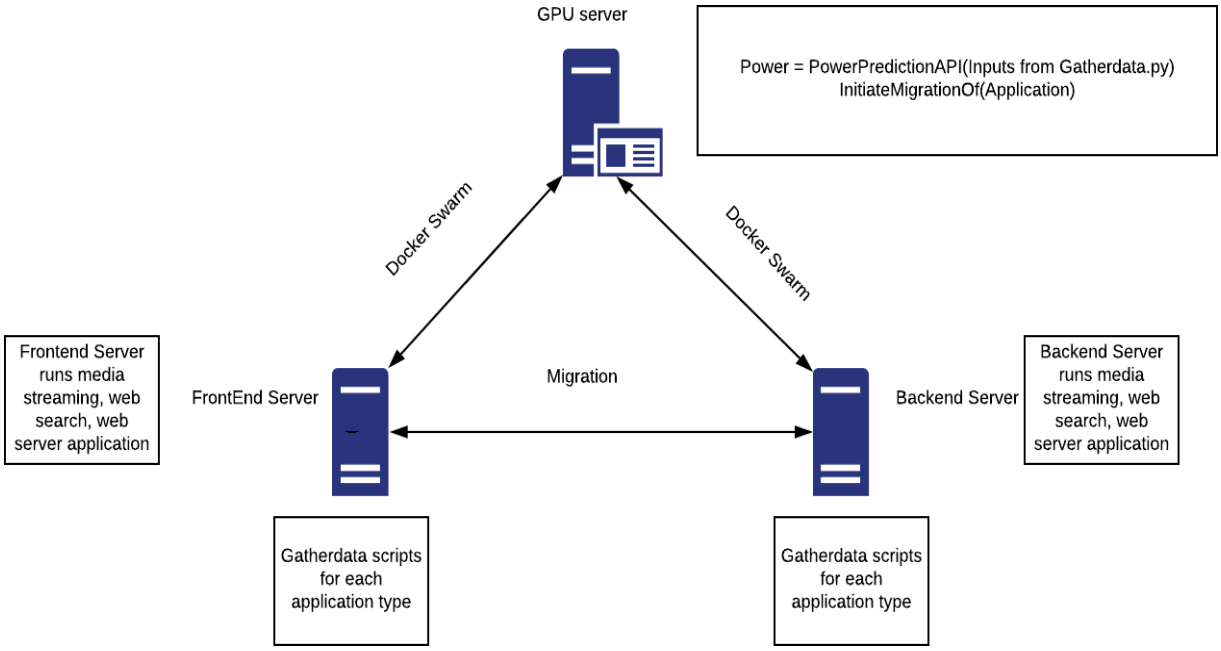
Improve Data Center Level Energy Efficiency

With the emergence of high energy proportional servers and increasing workload complexity, data center level energy efficiency can be improved with a new workload scheduling algorithm that exploits the high energy proportionality of individual data servers, dynamically detects individual servers' power level, and balances workloads across servers, such that all servers are forced to stay in peak-energy efficiency level.

Overall Framework

The overall framework of the new workload scheduling algorithm for data center level energy efficiency is illustrated in Figure 5. The overall framework includes coordinating servers and worker servers, where coordinating servers collect various system statistics of worker servers to predict power levels of the servers and trigger load migration to force all servers to run under the peak-energy efficiency levels. The details of the mechanism are explained in the following subsections.

Figure 5: Overall Framework of Workload Scheduling Algorithm for Data Center Level Energy Efficiency



Source: UC Riverside

Technical Methods

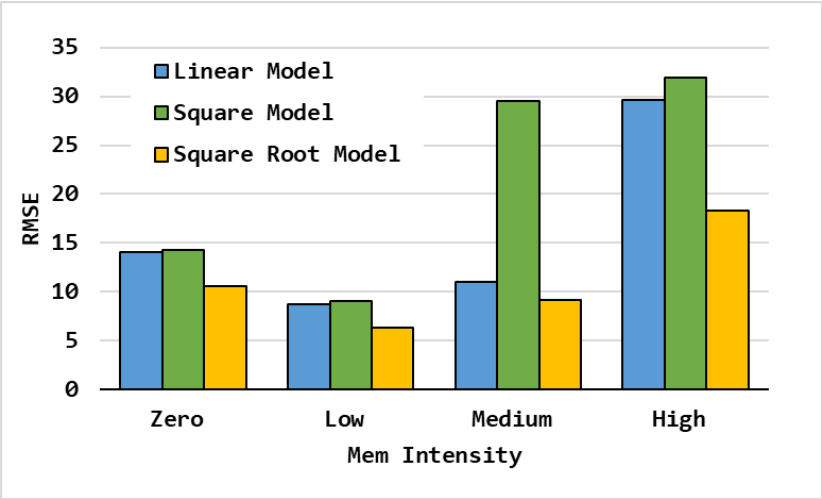
DNN-based Current Server Power Prediction

Many previous studies proposed dynamic load control algorithms leveraging linear or polynomial power prediction models that relied heavily on CPU utilization to find the migration triggering point, which may not be accurate for realistic data center environments, since present data center applications have diverse performance characteristics. Especially in the big data era, system memory and disk accesses contribute significantly towards the overall system power. Thus, the conventional CPU-utilization-based power models may not reflect this trend. Figure 6 shows the prediction accuracy of various power models with regard to memory intensity of applications in root mean square error. The models are from an energy model study in cloud simulation frameworks (Makaratzis et al., 2018). The models use the linear, the square, and the root square of CPU utilization in the following equation for power prediction. P_{min} and P_{max} are the minimum and the maximum server power, respectively, and $Util^{\alpha}$ is the weight value based on CPU utilization. The linear, square, and square root models set α value to 1, 2, and 1/2, respectively.

$$\text{Server Power} = P_{min} + P_{max} - P_{min} \times Util^{\alpha}_{CPU}$$

Figure 6 shows the prediction error rate of various models in root mean square error (RMSE) values. The RMSE is the standard deviation of the residuals and is widely used to measure prediction errors. As can be seen in Figure 6, across all the models, the error rate is exponentially increasing as the memory intensity increases, where the averaged power prediction error for the high memory intensive applications is greater than 30, which is not negligible. Instead, the project team's approach uses machine learning algorithms to predict individual server power levels accurately. Machine learning algorithms are widely used for various cognitive problems, such as self-navigation systems, speech and face recognition, etc. Machine learning algorithms are effective in extracting meanings and key features of big data inputs without human intervention or sophisticatedly designed algorithms. In this project, the project team leveraged machine learning algorithms to extract the relations between various system parameters (including memory/storage usages as well as CPU utilization) and the server power consumption by feeding the machine learning algorithm server power statistics history data.

Figure 6: Server Power Prediction Errors of Conventional Power Models With Application Memory Intensity



Source: UC Riverside

There are numerous algorithms in machine learning. The project team used a fully connected deep neural network (DNN) for current power prediction because DNN is effective for extracting common patterns from input data where individual inputs consist of relatively fewer parameters (such as tens of values in one set of input). Note that another famous machine learning algorithm, convolutional neural network (CNN), is effective for object detection in image inputs where each image has hundreds of pixel values. For server power prediction, only tens of system statistics were used as a sample of input data, which is insufficient for CNN. It's likely that the other widely used algorithms, such as recurrent neural network (RNN) and reinforcement learning (RL), are not applicable for the power prediction because these algorithms are effective for time-series data (stock-price prediction) or interactive problems (e.g., games). Therefore, the project team used fully connected DNN. To train a DNN model, system parameters were collected, along with power consumption. Out of 28 parameters collected by running two Linux profiling commands, *psutil* and *perf*, not all parameters had a clear relation with power consumption. The project team identified the top 10 parameters that showed a strong relation with server level power consumption by measuring the correlation coefficient between individual parameters and power consumption. The selected parameters and their meanings are summarized in Table 1. The top 10 parameters are noted with a "Y" in the

Table 1: System Statistics Considered for Power Prediction

Parameter	Description	Selected
CPU Frequency	Frequency of the CPU	Y
User Time	Time spent by the CPU in user space	Y
CPU Util	Percentage of CPU used	Y
Interrupts	Number of Interrupts	Y
S/W Interrupts	Number of Software Interrupts	Y
Processes	Number of processes in the system	Y
Cache Miss Ratio	Cache miss ratio	Y
Virtual Mem Usage	Percentage of virtual memory used	Y
System Calls	Number of System Calls	Y
Instructions	Number of instructions executed	Y
Context Switches	Number of context switches	N
TX Network	Network bytes sent	N
RX Network	Network bytes received	N
Network Connec.	Number of Network connections	N
Cache Mem Usage	Number of bytes used in cache memory	N
Swap Mem Usage	Number of bytes used in swap memory	N
I/O Reads	Number of bytes read from I/O	N
I/O Writes	Number of bytes written to I/O	N
System Time	Time spent by the CPU in kernel space	N
Idle Time	Idle time spent by the CPU	N
IO Wait time	Time spent waiting for I/O	N
MPKI	Cache misses per kilo instructions	N

Table 1 includes selected 10 parameters that are used for DNN model development.

Source: UC Riverside

Selected column. In addition to CPU-related parameters such as CPU frequency, CPU Util, Processes, Instructions, and User Time, there are parameters that are related to system memory, cache, and interrupts, which are Interrupts, S/W Interrupts, Cache Miss Ratio, Virtual Mem Usage, and System Calls. In conventional power models, parameters about system memory, cache, and interrupts are not considered. These 10 parameters with power consumption were collected every 5 seconds for 25,000 seconds while varying the combination of concurrently executing workloads to collect 5,000 training data.

As there are no ground rules to design a DNN structure, the project team used heuristic search, which is the de facto standard of DNN design to date. The search space was set to two major hyperparameters, the number of neurons per layer and the number of layers. First, the team evaluated the accuracy impact of the number of layers while varying the number of hidden layers from one to seven and employing an identical neuron count per layer from 10 to 1024 in each experiment. The error rate was lowest for five layers at 4.95 watts (W). Next, the team changed the number of neurons while using five layers. As there are a plethora of combinations of five-layer DNNs, the team referenced various well-known DNN structures and narrowed down the combinations. Of the tested combinations, the model that used an ascending but repeated number of neurons for two consecutive layers derived the best prediction accuracy (an average of 4.5 watts error boundary). Therefore, the DNN was modeled to have five hidden layers of 128, 128, 256, 256, and 512 neurons for each of the five layers, respectively.

RNN-based Future Server Power Prediction

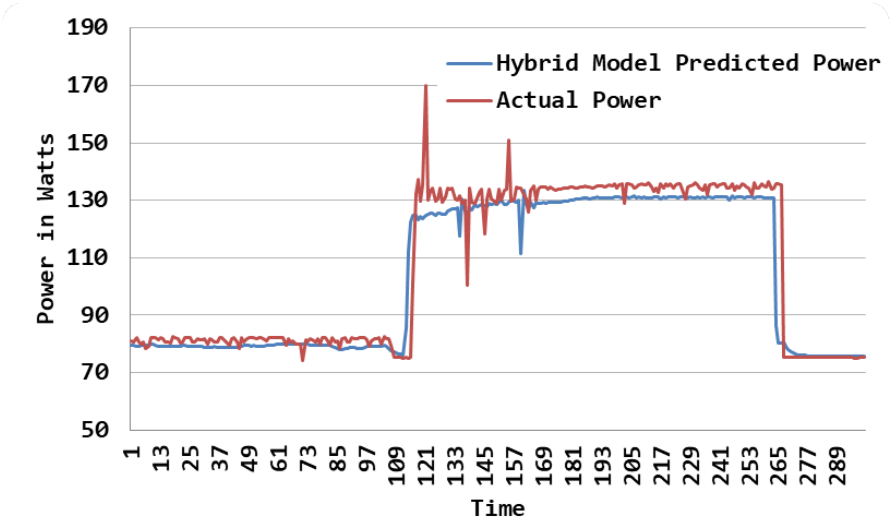
DNN is sufficient to estimate the server power consumption. However, as the DNN is relying on the current system parameters and predicting the current power consumption when it is detected that the server is reaching the peak energy efficiency point, it is already too late to initiate migration due to the migration latency. Thus, the team also designed a future power prediction model by using RNN, which is a type of DNN that predicts time-series data. RNN extracts the trend and dependency between consecutive data in a series of inputs and predicts the next/following data. As discussed earlier, RNN is not effective to predict power consumption data based on system parameter values. However, for future power prediction, RNN is effective if power consumption data are rearranged to time-series data. For example, if RNN is provided a series of past power consumption data (for example, past 30-minute power measurements where each power value is measured every one minute), RNN extracts the power consumption trend of the server and predicts the next likely power value (such as the power consumption of one minute later, which is minute 31). By exploiting this next data prediction feature of RNN, the project team designed an RNN to predict the future power consumption of servers. The same training data set used for DNN development was used for RNN development. The training data was collected while running workloads for a few hours with 5-second intervals, showing the time-series power and system statistics.

As in the DNN design, the project team ran heuristics to find the optimal RNN structure for power prediction, and it found that the error rate was lowest out of the search when four layers with 128 neurons were used. Among various RNN designs, long short-term memory (LSTM) cells were used as individual neurons. With 20 percent dropout per layer, Adam as the

optimizer, and Tanh function as the activation function, the model was trained with 100 epochs. The final accuracy result was 7.57 watts. To enable an RNN to correctly predict future power by extracting the trend from the previous time-series data input, the number of input data should be determined. The project team evaluated the prediction accuracy while varying the number of inputs from 30 to 60, which are widely used by RNN models. Both 30 and 60 inputs derived lower prediction errors than 40 and 50 inputs. However, as 60 data points are 120 seconds worth of data, feeding 60 data points would make the warming-up time unnecessarily long with only a 0.05 error reduction compared to 30 data points. Therefore, 30 inputs per one future power value prediction was used.

The RNN and the DNN are integrated as a hybrid model. DNN is first fed with system parameters, and it predicts the current power consumption. RNN is then fed with 30 DNN prediction results and it predicts one future power consumption. Figure 7 is a captured time frame, where both predicted and actual power are shown synchronously to the time. For the hybrid model, 30 past power values are fed as inputs, which is power trend values for 150 seconds, as each data point is collected every five seconds. As can be seen, the hybrid model (blue) catches the trend of the actual power consumption (red) where the power peak is predicted ahead of time of the actual power peak.

Figure 7: Actual vs. Predicted Power



(X-axis unit: 5 seconds)

Source: UC Riverside

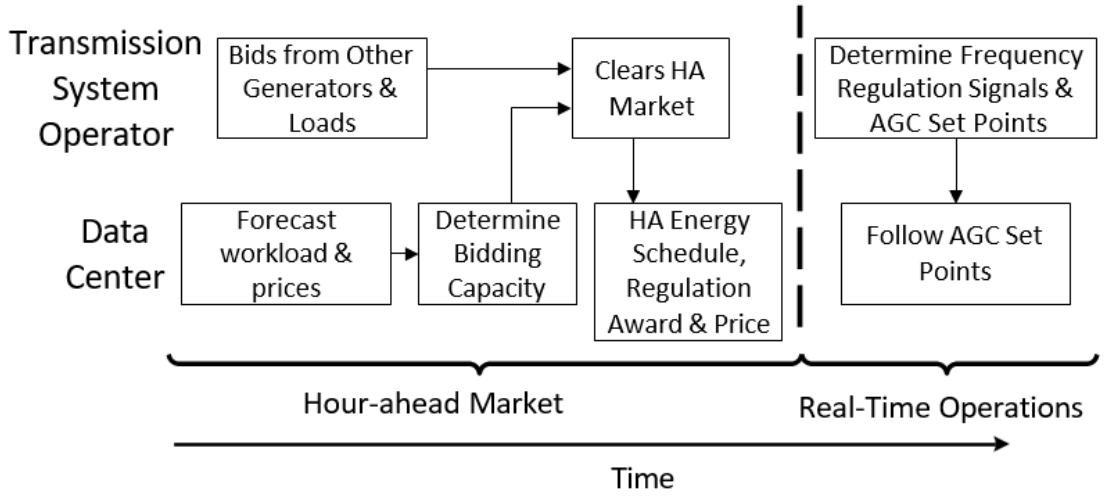
Improve Data Center Cluster Energy Efficiency

The most effective way to improve data center level energy efficiency is to enable the data centers to provide ancillary services to the electricity market. Data centers can adjust their energy consumption and provide frequency regulation services in real time by implementing dynamic voltage and frequency scaling (DVFS) and introducing dummy computing loads.

Overall Framework

The overall framework of the frequency regulation service provision by a data center is depicted in Figure 8. The overall framework involves interactions between the transmission system operator (TSO) and the data center (DC) in two electricity market processes: hour-ahead (HA) market and real-time operations. The details of the frequency regulation service provision framework are described in the next three subsections.

Figure 8: Overall Framework of Frequency Regulation Service Provision by Data Center



Source: UC Riverside

Technical Methods

Data Center's Participation in Electricity Market

To provide frequency regulation services, the data center is required to participate in two electricity market processes: the HA market and real-time operations.

In the HA market, the DC first predicts the prices for energy and frequency regulation service and the workload of the DC for the next operating hour. The DC then determines the optimal bidding capacity for energy and frequency regulation services that maximize its expected net benefits subject to certain risk limits. After the HA market is cleared by the TSO, the DC receives the hour-ahead energy schedule, the award for frequency regulation service, and the cleared prices for energy and frequency regulation.

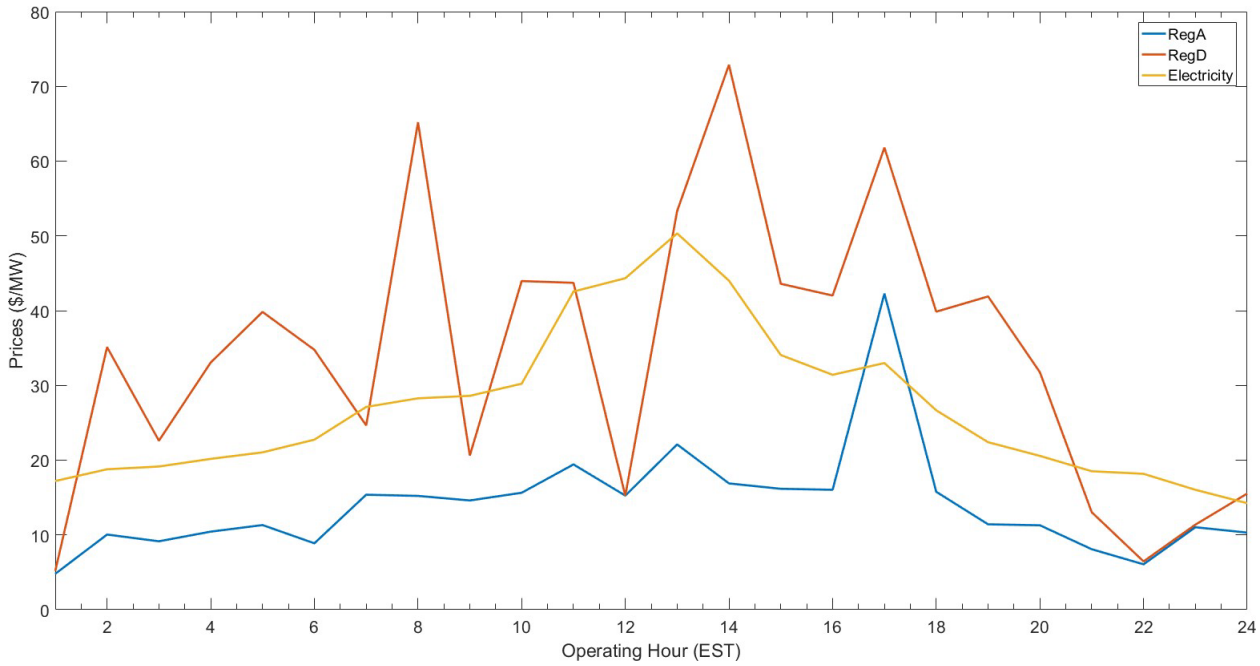
In real-time operations, the DC receives the frequency regulation signals and automatic generation control (AGC) set points from the TSO every two seconds. The frequency regulation signal ranges from -1 to 1. The signals are negative (positive) when the system requests frequency regulation down (up) services. The AGC set points specify the amount of load the DC should consume. The AGC set points equal the summation of the HA market energy schedule plus the product of the frequency regulation signals and frequency regulation service awards. Upon receiving the AGC set points, the DC adjusts its energy consumption to follow the set points. It can accurately follow the AGC set points by dynamically routing arriving

requests to various servers, changing the operating frequency of CPUs, and inserting dummy loads at the server level.

The physical and contractual constraints of the DC need to be taken into consideration when it participates in the electricity market. First, the bidding capacity for energy, Pbase, and frequency regulation service, Bcap, should be determined in such a way that the maximum and minimum power consumption limits, Pmax and Pmin, of the DC will not be violated. If the submitted bids are accepted, in real-time operations the AGC set points for the data center range from $PP_{bbwwddww} - BB_{ccwwTT}$ to $PP_{bbwwddww} + BB_{ccwwTT}$. The DC needs to make sure that $PP_{bbwwddww} + BB_{ccwwTT} \leq PP_{TTwwxx}$ and $PP_{bbwwddww} - BB_{ccwwTT} \geq PP_{TTiiTT}$. Second, the DC needs to satisfy the service level agreement (SLA) and maintain the quality of service (QoS) as a cloud computing service provider. Hence, the control of request routing, CPU frequency, and dummy loads are limited by the SLA requirements.

Finally, note that, in an electricity market such as the Pennsylvania-New Jersey-Maryland Interconnection (PJM), there are two types of frequency regulation services, RegA and RegD. RegD is a frequency regulation service with fast response. RegA is a frequency regulation service with slow response. The real-time regulation signal of RegD service is much more volatile than that of RegA service, and the price of RegD service is higher than that of RegA service. The DC can control its server energy usage in real time to follow the volatile RegD service signals. Hence, it is suitable for the DC to provide such premium frequency regulation services and receive higher compensation from the electricity market. Figure 9 shows an example of daily prices for frequency regulation services and energy in the PJM market.

Figure 9: Prices for Frequency Regulation Services and Energy in the PJM Market



The yellow electricity line indicates the locational marginal price of electricity.

Source: UC Riverside

Transmission System Operator

In the hour-ahead market process, the TSO first receives both energy and frequency regulation service bids from generators and DCs. It then clears the hour-ahead market to determine the hour-ahead energy schedule and prices for energy and frequency regulation services. The objective is to minimize the total energy and frequency regulation service costs while satisfying the electric loads. The market-clearing results are sent to DCs and other market participants.

In real-time operations, the TSO first measures the area control error and computes the frequency regulation signals of the system, aiming to reduce the area control error to zero in a distributed fashion. The individual generator and DC's AGC set points are calculated based on the frequency regulation signal, hour-ahead energy schedule, and frequency regulation service awards. The updated AGC setpoints are sent to the generators and DCs every two seconds.

Performance-based Compensation

The final compensation for providing frequency regulation service depends on the frequency regulation service award amount and the real-time AGC set points signal following the performance. The signal following performance is quantified by the performance score in the PJM market. It consists of three components: accuracy, delay, and precision.

The accuracy score is the correlation between the AGC set point signals and the DC's response. It is calculated over a five-minute period with 10-second granularity. The calculation is performed repeatedly with 10-second delays propagated over five minutes, where the best score is used. The delay score is based on the time delay between the control signal and the point of the highest correlation. The delay score will be 100 percent if the best correlation is at 0 or a 10-second delay. It decreases as the delay time increases until the five-minute mark. The precision score is calculated based on the instantaneous error between the control signal and the regulating unit's response. The final performance score is the average of the three components.

CHAPTER 3:
Project Results

Improvement in Server Level Energy Efficiency

Evaluation Setup

The project team constructed a software prototype that implements μDPM, the server low power management technique, in user-space. μDPM was evaluated with Memcached, a high-performance key-value store application. Memcached was set up using the data caching benchmark from the Cloudsuite benchmark suite. Memcached had one of the shortest service times in the team's simulation experiments and, hence, represented one of the most challenging workloads for server low power management. This workload has an average service time of 30 μs, tail service time of 33 μs, and target tail latency (performance target) of 150 μs.

Two multicore servers were used, one client and one request-processing server. The client server established multiple TCP (transmission control protocol) connections with the server, emulating multiple clients. The inter-request time distribution for each client was exponential and was scaled to see the impact of different traffic loads. The server had dual Intel Xeon E5-2620 v4 processors running at 2.1 GHz (gigahertz), 128 GB (gigabytes) main memory, and 1TB HDD (1 terabyte hard disk drive). Hyperthreading and Turbo Boost were disabled during the experiments. Memcached was run across 12 threads, with a fixed one-to-one thread-to-core mapping to avoid run-to-run variance caused by the Linux thread scheduler. All scheduler threads were running on one core mapped to the second processor, mimicking a dedicated μDPM scheduler hardware.

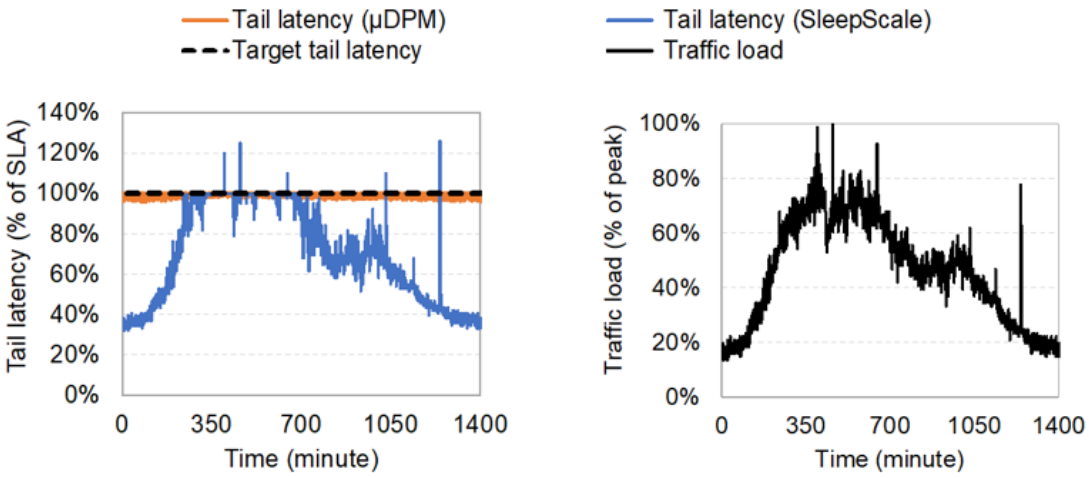
Performance of Server Low Power Management

The main outcome of the pilot test was the demonstration that the server low power management policy, μDPM, could achieve significant power savings under real-world data center conditions. This was demonstrated by running Memcached, a common data center workload, in a containerized Docker environment. Data center workloads are typically virtualized to simplify the deployment and manageability of workloads. The most light-weight (low software overhead) and widely used virtualization technologies used are referred to as containers.

The main performance metric for this workload was the request response time (or latency). Specifically, the main performance factor was the 95th percentile tail latency, which is the time where 95 percent of requests completed beforehand; it is shown in Figure 10(a) as the black dashed line. The tail latency is normalized on the y-axis. The blue line shows traditional DVFS + Sleep state policies (the specific algorithm implemented is called SleepScale), which are uncoordinated. Due to the uncoordinated nature of DVFS and sleep states, it is possible that these policies can miss the target tail latency, resulting in significant performance violation to the data center operators.

The project team subjected the Memcached workload to a real-world varying traffic pattern, as shown in Figure 10(b). The request per second of the trace was scaled such that the peak traffic load corresponded to the peak server load. The traffic pattern covered a wide range of use, from low utilization (20 percent) to high utilization (80 percent), with traffic spikes of up to 100 percent utilization.

Figure 10: Tail Latency Under Varying Traffic Loan



(a) Tail latency observed under test

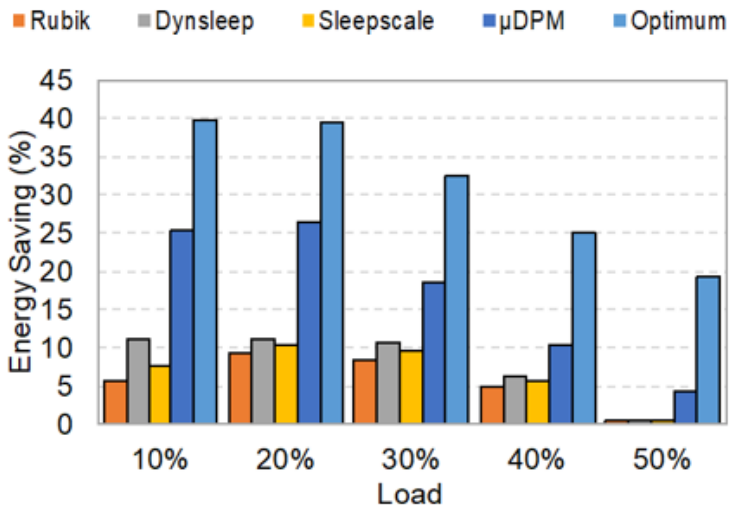
(b) Traffic pattern under test

Source: UC Riverside

The server low power management policy, μDPM, is shown in orange. Due to being aware of the target tail latency and being able to change the sleep and DVFS condition at every request arrival, μDPM is deadline-aware and can adjust on the fly to varying traffic loads. In addition, μDPM can coordinate both low power states in order to finish processing just-in-time, conserving as much power as possible. μDPM can achieve tail latency that is always just below the target tail latency.

In terms of energy savings, Figure 11 demonstrates the benefits of μDPM under varying loads, compared to other existing state-of-the-art server low power management algorithms. The Rubik algorithm represents a DVFS-only low power policy, Dynsleep represents a state-of-the-art sleep-state-only low power policy, Sleepscale represents a state-of-the-art DVFS+Sleep state low power policy. Optimum represents that hypothetical best-case energy savings where the server does not consume any idle power and sleep transitions do not incur any additional overhead. Overall, the μDPM outperformed existing state-of-the-art server low power management algorithms, consistently achieving ~2x (approximately two times) more energy savings. In the best-case scenario, we observed that μDPM achieved server energy savings of up to ~25%.

Figure 11: Energy-saving Comparisons Among Different Power Management Schemes



Source: UC Riverside

Improvement in Data Center Level Energy Efficiency

Simulation Setup

The pilot test used a minimum of three servers that ran the Linux operation system, which is the most commonly used operating system in large-scale data centers. One of the servers ran the workload manager (marked as "GPU Server" in the figure), and the others ran the data center workloads. The workload manager server periodically collected various system stats of the worker servers, predicted power consumption, and triggered load migration if any server reached the peak efficiency threshold. The workload manager estimated the power consumption of the worker servers by using DNN models. To accelerate the processing speed of deep learning solutions, the workload manager server was equipped with a GPU. One of the state-of-the-art open-source deep learning frameworks such as TensorFlow was executed to run the DNN model on GPUs. The system stats used for the power estimation are collected by each of the worker servers by using Linux commands that are *psutil* and *perf*.

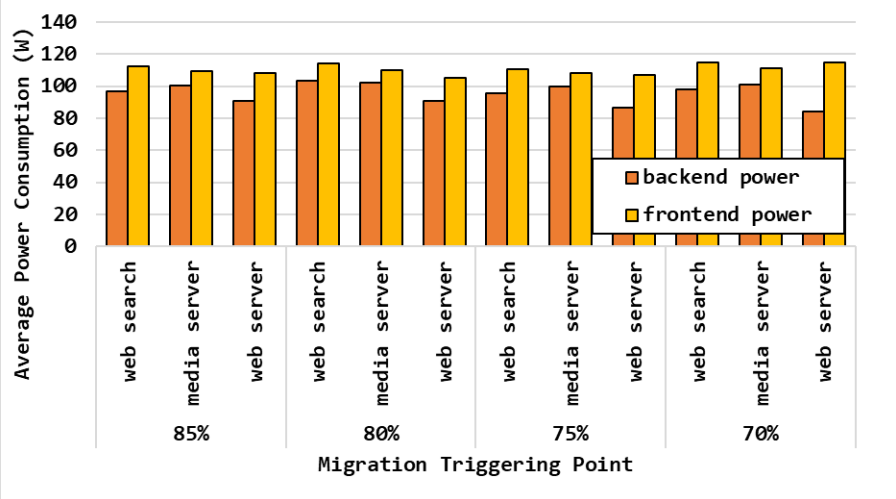
Applications are typically "virtualized" to provide isolation from other applications and enable easy mechanisms to migrate and move applications from server to server. The project team chose to use light-weight containers, such as Docker, to provide virtualization. To manage the workload migration among servers, the team used an open-source container orchestration platform, such as Docker Swarm. The workload manager server initiated Docker Swarm for the worker servers, such that each of the workloads in the worker servers could run in a docker container. Once the deep learning model detected an overloaded server, the workload manager server commanded the workload migration in a unit of a Docker container. More specifically, a checksum image of the target workload was created in a compressed file, the image was copied to the migration target server, and a new Docker container resumed the workload with the copied checksum image. The basic migration was supported by the Docker

Swarm. For further performance improvement, various optimization techniques were applied that the project team was exploring, such as creating a checksum only for the data generated at run-time rather than creating one for both static and dynamic parts of the workload. All these virtualization and migration frameworks and optimization techniques are supported by open-source solutions and Linux systems.

Performance of DNN-accelerated Load Scheduling Algorithm

The project team integrated the DNN-RNN hybrid model with a migration script. The migration script checks RNN's future power prediction data and determines whether the power value is exceeding the peak energy efficiency point. The team evaluated the overall power efficiency while varying the migration triggering point from the top 70 to 85 percentile power consumption and the migrating applications. Evaluations of other percentile levels were excluded because there were too frequent migrations when using a percentile below 70 and no migration when using a percentile above 85. Figure 12 shows the summary of the mean power consumption of two servers, which are migration source and migration target servers (named backend and frontend servers to distinguish) when using different migration triggering points and migrating applications. All experiments started running all three specified applications on the backend server and measured the power consumption while migrating one of the applications when the backend reached the specified migration triggering point. For the same choice of migrating application, the top 75 percentile power level derived the minimum overall power consumption for both backend and frontend servers, which means that 75 percentile power consumption was corresponding to the peak energy efficiency level. Therefore, the migration script was implemented to trigger the migration once the RNN predicted that future power reached the top 75 percent of the power consumption of the migration source server. To evaluate the effectiveness of the load migration algorithm, all possible combinations of workloads with the four workloads were formed, a total of 15 cases. Each experiment ran for one and one-half hours and monitored the power trend and checked to see whether the total aggregated power was decreasing after the migration.

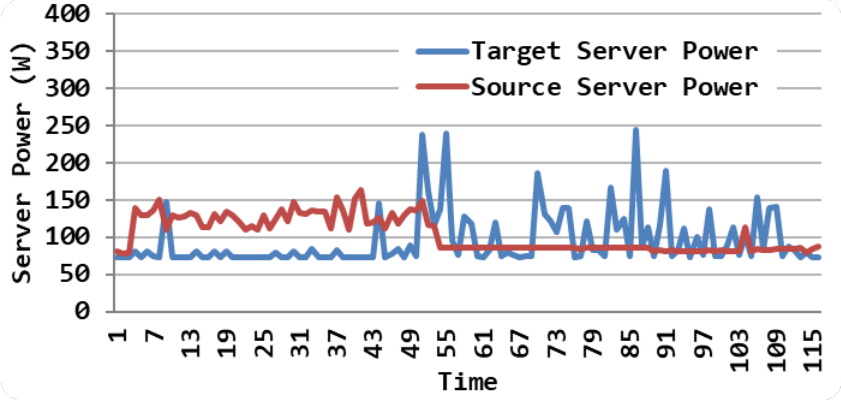
Figure 12: Mean Power on Two Servers With Respect to Migration Triggering Point



Source: UC Riverside

Two results are shown as examples. Figure 13 shows individual server power consumption while the migration was triggered. While the Media streaming and Graph Analytics were running on the migration source server, between 40 to 50 timestamps, the RNN's power prediction indicated that the power value would be exceeding the migration threshold power level and triggered the migration of the Graph application. As can be seen in the figure, the migration source server power significantly dropped, and the target server power spiked slightly when the migration was started and then slowly saturated.

Figure 13: Example Migration Result: Media and Graph

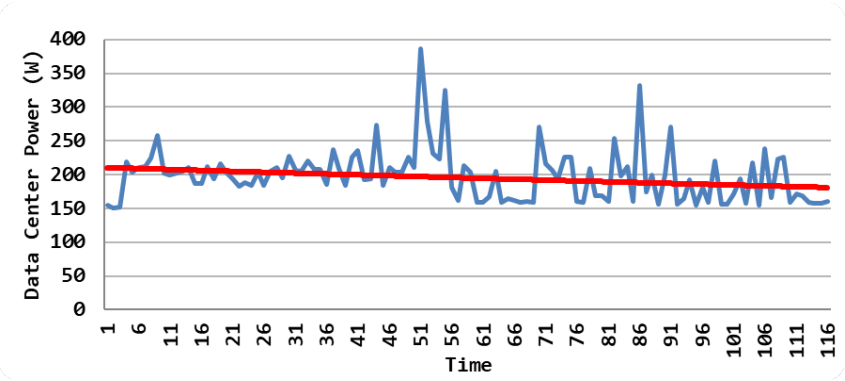


Graph and Media ran together, and Graph is migrated at around 40~50 time points.

Source: UC Riverside

Figure 14 shows the total power of both servers to understand the data center level power efficiency. The red line is the trend line based on the blue total power consumption. The total power is decreasing towards the end of the experiments, which shows that the migration helps save power while servicing both workloads.

Figure 14: Example of Migration Result: Red Line



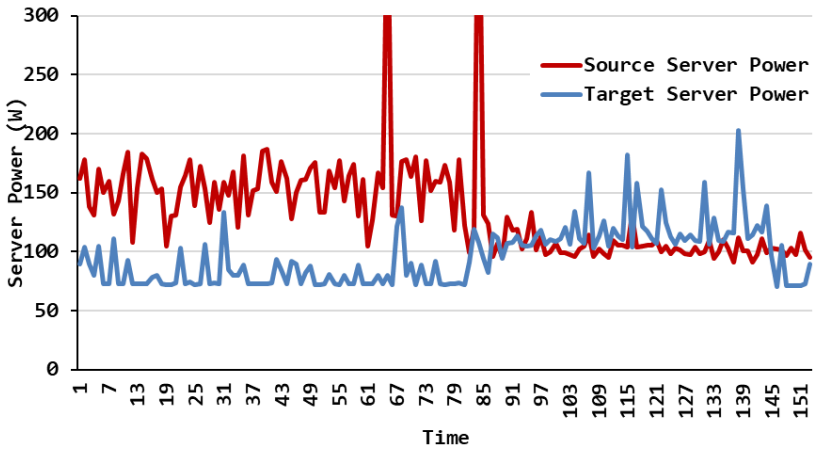
The red trend line shows that total power is decreasing with migration.

Source: UC Riverside

Figure 15 and Figure 16 show more aggressive results, where four workloads run on the source server and one of them, RNN-DNN, which is the most power-hungry application, is

migrated. As the sharp power reduction from the source server shows, the migration happened at around 75-85 time points. After 85, the target server runs the RNN-DNN, and the source server runs the remaining three applications (Graph, Media, and Web search). After the migration, as shown in Figure 15, both servers' power levels become well balanced. Figure 16 shows that the data center level power is also decreasing toward the end of the migration, where 25 percent total power reduction is observed compared to the total power consumption before the migration.

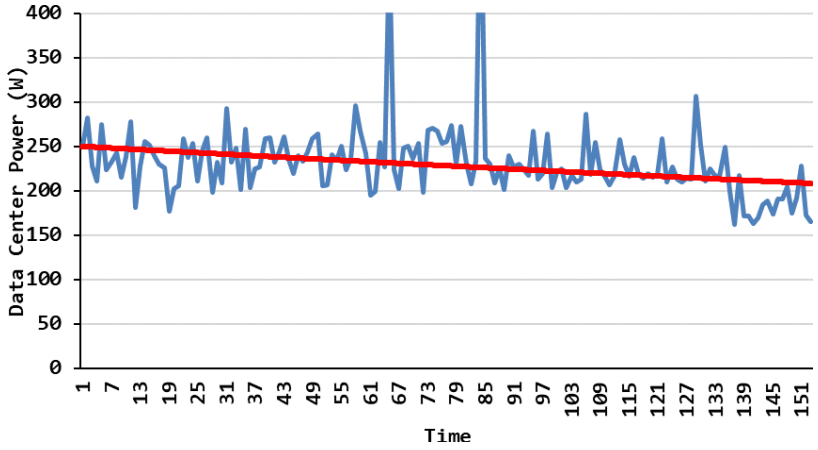
Figure 15: Example Migration Result: RNN-DNN, Graph, Media, Web Search



RNN-DNN, Graph, Media, and Web Search ran together and RNN-DNN is migrated at around 75~85 time points.

Source: UC Riverside

Figure 16: Example Migration Result: Red Color Trend



The red trend line shows that total power is decreasing with migration.

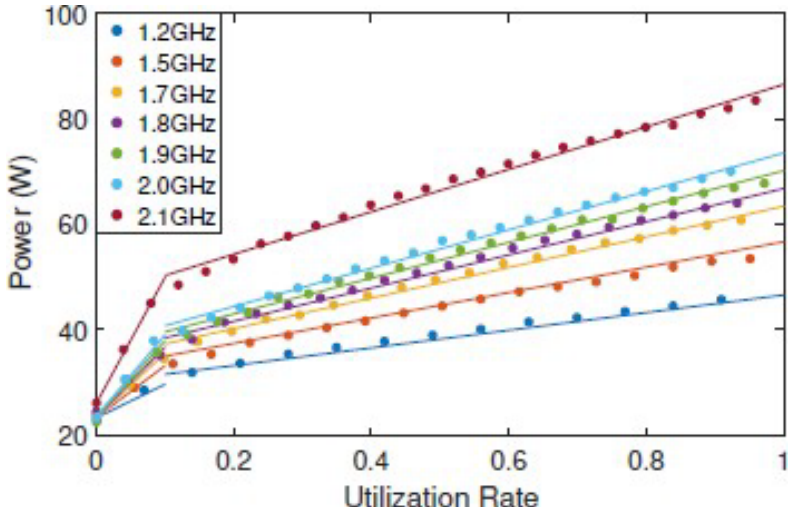
Source: UC Riverside

Improvement in Data Center Cluster Energy Efficiency

Simulation Setup

It was assumed that the data center in the numerical study had 100,000 servers, which were then assumed to have the same power curves as shown in Figure 17, with power consumption ranging from 22W to 85W. The maximum capacity of each server was 1,230 requests per second. The SLA specified that 90 percent of the requests be processed within 115 ms (milliseconds). The corresponding limits on the utilization rate were 0:8 at 2:1 GHz and 0:77 at 2:0 GHz. To simulate the data center's workload, Wikipedia's access trace was adopted from the online repository [24]. The historical prices for frequency regulation and energy from the PJM market were used for electricity market simulation.

Figure 17: Fitted Power Consumption Curves With Default Sleep Policy

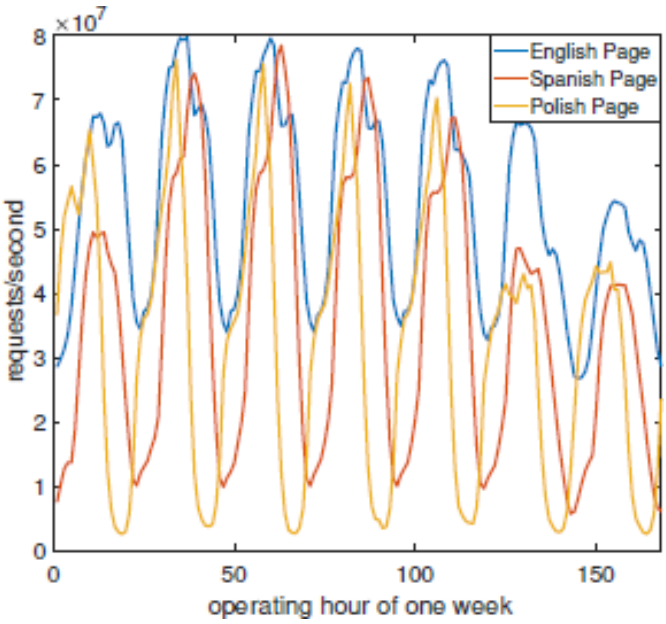


Source: UC Riverside

Performance of Frequency Regulation Service Provision by Data Center

The performance of the frequency regulation service provision by the data center was evaluated from three perspectives: frequency regulation signal following performance, electricity cost, and request response time. The price prediction results of the 12 last weeks for each month of year 2017 were used in the simulation. During the performance evaluation, it was assumed that the data center would provide frequency regulation service to the electricity market whenever the frequency regulation service price was higher than the energy price. In other words, risk constraint was not considered. The risk constraint indicated the level of financial uncertainty in terms of the electricity bill associated with the data center operation. In the real-time operation simulations, the data center was expected to follow the historical frequency regulation signals from the PJM market. The requests served by the data are derived from the scaled request arrival rate of English, Spanish and Polish pages in the last week of the Wikipedia trace, as shown in Figure 18, which was repeatedly used. The use rate of each server was determined by the bi-linear power model. The actual power consumption of the data center was estimated with empirical measurement data with interpolation.

Figure 18: Hourly-averaged Request Arrival Rate After Scaling



Source: UC Riverside

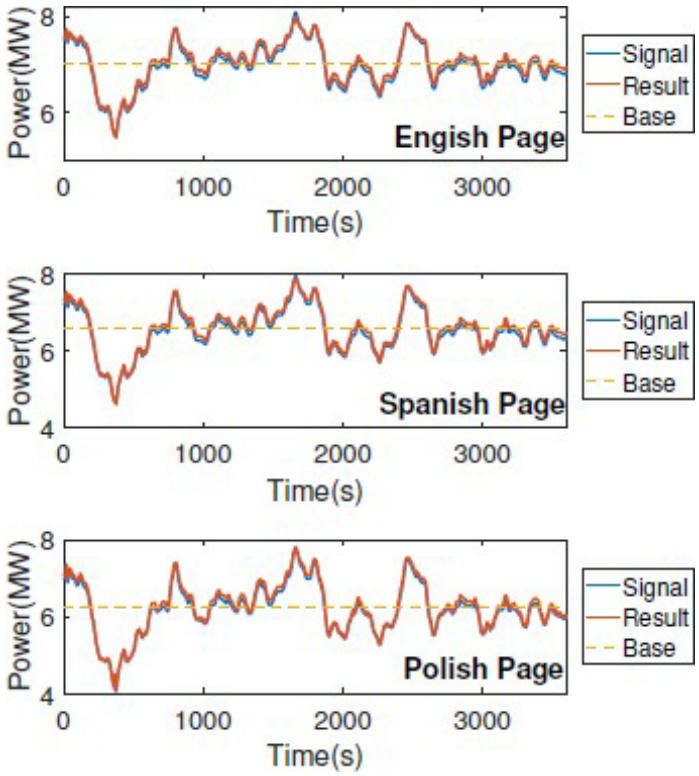
First, the frequency regulation signal following the performance of the proposed data center power consumption control algorithm was quantified by three metrics: accuracy, delay, and precision. The frequency regulation signal and the actual power consumption trajectory of the data center for an hour are depicted in Figure 19, which shows that the proposed data center power consumption control algorithm allows the data center to follow the frequency regulation signals closely. The accuracy, delay, and precision scores for the 12 weeks were calculated and shown in Table 2. The accuracy, delay, and precision scores of the data center are very high for the three types of page visits traced. The small frequency regulation signal tracking errors mainly come from two sources: the requests prediction error and the approximation error of the piece-wise bi-linear server power model.

Table 2: Frequency Regulation Signal Following Performance Scores

Performance Score	English Page	Spanish Page	Polish Page
Accuracy	99.76%	99.69%	99.61%
Delay	1	1	1
Precision	95.36%	95.83%	95.62%

Source: UC Riverside

Figure 19: Frequency Regulation Signal Following One Hour for Three Pages



Source: UC Riverside

Second, the reduction in electricity cost by participating in the frequency regulation market for the data center was calculated. If the data center did not provide frequency regulation service to the power system, it was operated to minimize its power consumption. The electricity costs of the data center under both scenarios are reported in Table 3. As shown, no matter which type of page requests were being served, the electricity cost was always lower when the data center provided frequency regulation services to the power system. For a data center with 100,000 servers, the proposed data center control algorithm resulted in, on average, a $19,100 (7.8 percent) electricity costs reduction for the 12 weeks.

Table 3: Electricity Costs of the Data Center Under Two Operating Scenarios

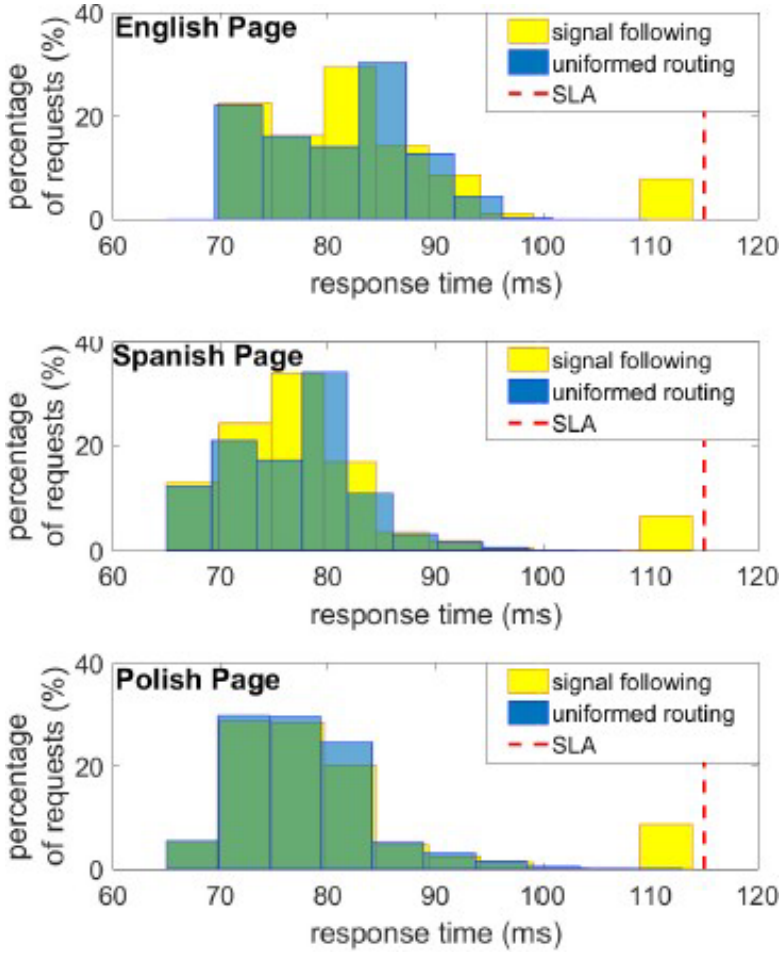
Electricity Costs	English Page	Spanish Page	Polish Page
Costs with Frequency Regulation ($)	266.60K	213.13K	194.51K
Costs with Minimum Power ($)	282.73K	234.48K	214.35K

Source: UC Riverside

Third, the request response time of the data center when providing frequency regulation services was calculated based on the proposed request routing algorithm. The distribution of request response time during the hours when the data center provided frequency regulation is shown in Figure 20. Compared to the uniform request routing strategy, when the data center followed the frequency regulation signals, only a small portion of the request response time

moved closer to the SLA's response time limit. If the data center did not provide frequency regulation service and instead minimized power consumption with packing strategy, the response time of almost all the requests would be very close to the SLA's response time limit. Compared to the minimum power consumption control strategy, the proposed data center control with frequency regulation provision reduced not only electricity costs but also the response time of requests.

Figure 20: Distribution of Request Response Time



Source: UC Riverside

Summary

The developed techniques operated at the server-level, rack-level and data center cluster-level and can operate orthogonal to each other. The evaluation results demonstrated that an estimated 7.8 percent, 25 percent, and 25 percent reduction in energy cost could be achieved through demand response (cluster-level), peak efficiency scheduling (rack-level), and server low power runtime (server-level), respectively. Therefore, the estimated energy cost was calculated as (1- 0.078) x (1- 0.25) x (1-0.25) = 51.9 percent. This represents a total of a 48.1 percent reduction in energy cost.

CHAPTER 4: Technology/Knowledge/Market Transfer Activities

Target Market – Enterprise Data Centers

A 2014 study by the Natural Resources Defense Council (NRDC) provided a useful market segmentation for this technology transfer plan, identifying five data center market segments (Whitney & Delforge, 2014):

- Small- to Medium-sized
- Enterprise/Corporate
- Multi-Tenant
- Hyper-scale Cloud Computing
- High Performance Computing

Using this segmentation, the project team's primary target was the Enterprise/Corporate data center segment, which accounts for approximately one-quarter of U.S. data center energy consumption. The study's authors note that this is not meant to be an authoritative model of data center energy consumption but, rather, a rough estimate to help understand the relative contribution of each segment's consumption.

The project team expects medium-scale data centers in the Enterprise/Corporate segment, managing their own systems with others' software, to be able to incorporate the approaches developed during the project and see direct energy-saving benefits. Additionally, some of the more sophisticated operators in the Small- to Medium-sized segment may be candidates for the technology.

A secondary target of the technology is the Hyper-scale Cloud Computing segment. Companies in this segment, which includes Facebook, Microsoft, Google, and Amazon, have their own development efforts and would be unlikely candidates to directly use the software developed on the program, but the concepts and ideas could inform their approaches.

A key attribute of the Enterprise/Corporate segment is the use of open-source products to run their data centers. Common platforms include Linux, Docker and OpenStack. The fact that data centers run on open-source platforms factored significantly into the project team's technology transfer strategy. The technology is applicable to new and existing data centers.

Within the target segment, there is a broad audience of actors, decision-makers, and influencers whom the project team aims to reach with information about the technology. The primary audience includes: data center managers, data center operators, IT managers, data center designers, energy/facility managers, and policymakers.

Overall Strategy and Transfer Activities

The project team planned three types of activities: technology development, passive outreach activities, and active outreach activities. All three strategies were used to reach out to as many audiences of interest as possible. The channels used to transfer the technologies of this project include publishing papers, presenting at conferences, interacting with conference attendees or through industry/TAC meetings, working as a standards association committee, and releasing developed products in public repositories. The complete list of activities so far is summarized below. The project team will continue actively reaching the community to transfer its technology. All proposed ideas were developed by using commercial off-the-shelf servers and existing open-source frameworks/tools, as described in Chapters 2 and 3. Thus, the team expects that the technology can be easily adopted for various commercial or opensource research projects.

- Academic papers and articles
    - Wei Wang, Amirali Abdolrashidi, Nanpeng Yu and Daniel Wong, "Frequency Regulation Service Provision by Data Center," *Applied Energy*, vol. 251, pp. 1-17, 2019.
    - Wei Wang and Nanpeng Yu, "Phase Balancing in Power Distribution Network with Data Center," Greenmetrics, pp. 1-6, Urbana-Champaign, IL, 2017.
    - Chih-Hsun Chou, Laxmi N. Bhuyan, and Daniel Wong, "µDPM: Dynamic Power Management for the Microsecond Era", in Proceedings of the 25th IEEE International Symposium on High Performance Computer Architecture (HPCA), 2019.
    - Ali Jahanshahi, Nanpeng Yu, and Daniel Wong, "PowerMorph: QoS-aware Server Power Reshaping for Data Center Regulation Service", *under review.*
    - Aman Chandan, Savyasachi Jagdeeshan, Namdev Prabhugaonkar, Rutuja Shah, and Hyeran Jeon, "DeepPower: Deep Learning Accelerated Server Power Prediction," *in preparation.*
- Presentations for conferences
    - Wei Wang (student of Nanpeng Yu) presented the conference paper "Phase Balancing in Power Distribution Network with Data Center" at GreenMetrics in Urbana-Champaign.
    - Daniel Wong presented his conference paper titled "µDPM: Dynamic Power Management for the Microsecond Era" at HPCA in Washington, DC.
- Standardization
    - Daniel Wong and Hyeran Jeon are on the IEEE Standards Association Committee for "P1924.1 - Recommended practice for developing energy efficient power-proportional digital architectures." (https://standards.ieee.org/project/1924_1.html)

Daniel Wong is the editor of the system-level writing sub-group and Hyeran Jeon is the editor of the components writing sub-group. Specifically, the lessons learned from this project effort shed light on many best practices for holistically designing the many levels of data center hardware infrastructure. These lessons are being directly incorporated into the standards to guide industry practitioners in designing energy efficient digital architectures. For example, the need to carefully coordinate sleep states and DVFS is recommended to achieve power-proportionality at the server level, and the need to migrate tasks to load balance servers is recommended to achieve power-proportionality at the data center level.

- Open-source repositories
    - The products of the data center level energy efficiency project are available at https://github.com/rrshah/energy_efficient_data_center. The repository includes scripts that operate Docker Swarm-based workload coordination, system parameter collection and transfer, and DNN-RNN hybrid model inference and training, as well as DNN and RNN model files.
- Meetings with domain experts or TAC members
    - October 2017 at MICRO'17 to discuss coordinated deep sleep and DVFS algorithms
    - November 2018 in Riverside, California, to discuss server low power management and load migration algorithm and software
    - March 2019 in San Jose, California, to discuss server low power management and data center workload management
    - February 2020 at HPCA'20 to discuss data center management strategy

Market Adoption

- Extension by Academia and Industry
    - Researchers of the National Renewable Energy Laboratory at Golden, Colorado, Pacific Northwest National Laboratory at Richland, Washington, and the University of Colorado, Boulder, (Yangyang Fu et al., AMC'20 and Applied Energy'20) cited the paper co-authored by Nanpeng Yu and Daniel Wong with the outcomes of this project. The researchers referenced the frequency regulation approaches and proposed extended ideas.
    - Researchers of Google and Stanford (K. Kaffes et al., SoCC'20) cited the paper authored by Daniel Wong and extended the idea of using processor power saving features for the energy efficient data center.

- Potential for Adoption
    - The project team is finding ways to encourage adoption of the research outcomes by industry and data centers. The team believes that the products of this project have a high potential for adoption in future data centers. For example, it is an inevitable trend that systems and architectures adopt deep-learning-assisted designs for better performance, reliability, and energy efficiency (Wang et al., pp. 1-17, 2019). As the developed deep-learning-assisted load migration approach does not need physical power meters, data centers will save facility cost as well as power consumption.
- Plans for Adoption
    - The project team will apply the developed products to its affiliated school data centers. The team is planning to closely communicate with the school IT administrators to check the applicability.
    - The project team will meet with other researchers at conferences and business meetings to further increase possibilities of adoptions and extension of the research outcome.

Possibility of Adoption in Linux OS

Adoption of coordinated deep sleep/DVFS management techniques, such as the proposed µDPM policy, in the Linux Operating System requires close coordination between the sleep state driver (also called the CPU Idle driver) and the DVFS driver (also called the CPUFreq driver). To achieve this, there are two possible approaches: (1) creating a fused driver that handles sleep state and DVFS, or (2) introducing coordination mechanisms between the existing sleep state and DVFS drivers. Option (1) is intrusive and not feasible due to separate Linux subsystems handling sleep states (CPU Idle subsystem) and DVFS (CPUFreq subsystem). Option (2) is more viable and realistic; thus, the project team is focusing on this method of integrating into Linux.

While the project outcome demonstrates the significant benefit possible with close coordination, the nature of Linux device drivers and modern processor design may make adoption challenging for several reasons, as detailed below.

- Abstraction: Linux device drivers are designed to be abstracted and "siloed" from one another. Requiring coordination between both sleep state and DVFS drivers would break this abstraction and create dependencies between drivers. This causes a maintainability issue of the driver's software.
- Ownership of drivers: There exist different CPU Idle drivers and CPUFreq drivers, depending on the processor being used. For example, acpi_idle is the generic CPU Idle driver that supports all processors. However, Intel processors use the intel_idle driver that is maintained by Intel and contains Intel-specific optimizations. As another example, the acpi-cpufreq is the generic CPUFreq driver. The Intel processor uses the intel_pstate driver for DVFS management. Therefore, in order to support coordination

between the CPU Idle and the CPUFreq driver, the mechanism would have to be adopted by driver maintainers, such as Intel in this example. Other processors, such as AMD and ARM, may have their own sets of drivers, adding to the challenge of having all vendors buy in to a new driver design of coupling the sleep state and DVFS driver.

- Shift to hardware managed sleep mode, DVFS and thermal: Modern processors have recently shifted towards managing sleep modes, DVFS, and thermal modes in hardware. That is, modern processors contain Power Management Controllers (PMC) on-chip, which coordinates idle mode, voltage, frequency, and thermals, in order to be as energy efficient as possible while providing performance and staying within a certain thermal limit. This level of coordination requires fast response times, on the order of microseconds; thus direct hardware control is needed without the overhead of software management. Therefore, the Linux drivers simply defer sleep and DVFS state decisions to the hardware.
- Quality-of-Service: While hardware can directly coordinate sleep and DVFS states, the hardware does not take into account quality-of-service (QoS) metrics that are tangible to end users, such as response times. The proposed µDPM technique takes QoS requirements into account when coordinating sleep states and DVFS. Therefore, in order for Linux and the hardware PMC to gain the benefits of the proposed approach, it is necessary to provide quality metrics to the driver and the hardware. To some extent, recent versions of Linux 5.10 (released in December 2020) now support performance hints being passed to the intel_pstate driver to tune and trade off energy savings and performance.

While direct adoption of the proposed µDPM policy in Linux may be challenging, the above highlighted trend of the Linux drivers and processor hardware are showing developments that incorporate many aspects of the project findings, mainly the need for fast coordination between sleep states and DVFS (with hardware managed PMCs in the processor) and the need to be quality-of-service aware (through performance hints). To fully realize the benefits, modifications to software (Linux) and hardware (processor) are necessary. Because of the importance of influencing future software and hardware design, the project team believes the standardization approach can provide greater saturation of the project's developed technology in the long term; this is not limited to data center servers but applies to all computing devices, including mobile and embedded systems.

One of the project findings is that the uncoordinated management of existing sleep state and DVFS drivers leads to significant inefficiencies. If the Linux OS adopts the µDPM policy, energy efficiency improvements are expected in line with the project results, which achieves server energy savings of up to 25 percent.

CHAPTER 5:
Conclusions/Recommendations

The team achieved two key objectives: 1) developed pre-commercial server, data center, and data center cluster energy efficiency technologies and strategies and 2) facilitated the adoption of data center energy efficiency technology by providing easily accessed software solutions. The project efforts advanced the energy efficiency technology for data centers at three levels. At the server level, the project team developed an innovative server-level lower power management system, called µDPM, which coordinates deep sleep states and dynamic voltage-frequency scaling and selects the optimal power state configuration for a given workload and traffic pattern. At the rack/data center level, the project team developed a workload scheduling algorithm to improve the data center level energy efficiency. This new algorithm collects various system statistics of worker servers to predict power levels of servers and trigger load migration to force all servers to run at peak energy efficiency. At the data center level, the project team developed an ingenious solution to enable data centers to provide ancillary services to the electricity market by adjusting their energy consumption. If all data centers in California adopt the three technologies developed in this project, there would be an annual electricity saving of 1,342 GWh, an electricity cost reduction of $163 million, and a GHG emission reduction of 596,114 metric tons.

CHAPTER 6:
Benefits to Ratepayers

Importance and Benefits to Ratepayers

The three proposed techniques work together at the data center level, across servers in the data centers, and within servers to enhance reliability and to improve energy efficiency and safety. For example, data center participation in frequency regulation services considers the operating condition of the distribution network to improve distribution system reliability. DNN-accelerated Load Scheduling aims to schedule servers to run at peak energy efficiency to enable servers to operate at higher levels of efficiency. The coordinated deep sleep and DVFS policy within servers can eliminate idle power consumption, lowering electricity costs to ratepayers.

Monetary, Energy, and Emission Savings for Ratepayers

The potential benefits to California IOU electricity ratepayers come from energy efficiency improvements due to peak efficiency scheduling and low power runtime of servers. The project benefits are estimated by comparing the new data center management policies with those of the existing data center management. The project team estimated that the three proposed techniques would yield annual electricity savings of 1,342 GWh, a quantifiable electricity cost reduction of $163 million per year, and a greenhouse gas emission reduction of 596,114 metric tons.

The assumptions and calculations for estimated benefits for the proposed techniques are described as follows. The project team estimated that 7.8 percent, 25 percent, and 25 percent reductions in energy cost can be achieved through demand response, peak efficiency scheduling, and server low power runtime, respectively, for a total energy cost reduction of 48.1 percent. The estimated energy reduction of these proposed techniques is derived from the project results evaluation shown in Chapter 3: Project Results. These results have been published in major international conference venues and journals relating to energy engineering and computer architecture. California's total electricity consumption is estimated to be 279,402 GWh by combining the grand total of electricity use of residential, commercial, industrial, and agricultural sectors. According to CEC's Data Centers Research website, data centers consume roughly 2 percent of the state's electricity, or 5,580 GWh. Servers consume roughly 50 percent of data center energy, or 2,790 GWh. Assuming a commercial electricity price of $0.20/kWh, the data center servers' cost is $558 million. Therefore, an estimated energy savings of $163 million would be anticipated, with an estimated total avoided electricity production of 1,342 GWh, corresponding with a reduction in GHG emissions of 1,342 GWh x 0.331 kg per kWh, or 596,114 metric tons.

Potential for Technology Adoption

Data centers: Data center operators typically avoid low power modes such as DVFS and sleep states due to the negative impact on latency. The proposed software runtime can meet strict latency requirements and improve the penetration of low power modes in latency-critical environments by at least 10 percent. The project team expects the majority of the on-site data centers to adopt its data center energy efficiency technology. The server and the data center cluster low power techniques proposed and evaluated in this project, along with the lessons learned from this project effort, are being incorporated into IEEE Standards P1924.1 Recommended practice for developing energy-efficient power-proportional digital architectures.

Demand response: The proposed load migration techniques can be applied to coordinate the operations of demand response resources. The proposed technology can be verified and implemented in the existing California demand response programs. In particular, the technology can be easily implemented in the demand response programs which the principal investigator (PI) managed at Southern California Edison Company (1,000 MW).

Potential Societal Benefits to Ratepayers

This project has the potential for environmental benefits and public health. Reduced electrical generation from fossil fuel power plants due to improved data center energy efficiency will result in reduced GHG emissions and criteria air pollutants associated with power generation. This will lead to an improvement in health for California residents.

This project also has the potential for personal computer benefits. This project develops a general power management runtime. Should the server low power management recommended practices be integrated into commercial operation systems (such as Linux), energy savings would not be limited to data centers but could be realized in consumer desktops and laptop devices that are connected to networks.

Power and computer engineering workforce: Co-PI Dr. Wong incorporated the latest research results on energy efficiency management of servers and data centers into the existing undergraduate and graduate computer architecture curriculum. In addition, computer designers can benefit from the open-source releases and IEEE Standards recommended practices. PI Dr. Yu has incorporated the latest research results on computing load migration into the existing undergraduate and graduate power engineering curriculum. This effort can significantly improve the quality of the future power engineering workforce in California. In addition, employees from the electricity utility industry may join the online master's program to gain a deeper understanding of the best practices to renewable integration and distribution system automation.

GLOSSARY AND LIST OF ACRONYMS

Term	Definition
AGC	Automatic generation control
CNN	Convolutional neural network
DC	Data center
DNN	Deep neural network
DVFS	Dynamic voltage frequency scaling
GHG	Greenhouse gas
GHz	Gigahertz
GWh	Gigawatt hour
HA	Hour-ahead
KWh	Kilowatt hour
LSTM	Long short-term memory
PI	Principal investigator
PJM	Pennsylvania-New Jersey-Maryland Interconnection
PMC	Power management controller
QoS	Quality of service
RL	Reinforcement learning
RMSE	Root mean square error
RNN	Recurrent neural network
SLA	Service level agreement
TAC	Technical advisory committee
TSO	Transmission system operator
µDPM	Micro dynamic power management

References

Andrae, Anders SG and Tomas Edler. 2015. "On global electricity usage of communication technology: trends to 2030." *Challenges* 6(1), 117-157.

Chandan, Aman, Savyasachi Jagdeeshan, Namdev Prabhugaonkar, Rutuja Shah, and Hyeran Jeon. Under preparation. "DeepPower: Deep Learning Accelerated Server Power Prediction."

Chou, Chih-Hsun, Laxmi N. Bhuyan, and Daniel Wong. 2019. "µDPM: Dynamic Power Management for the Microsecond Era." In Proceedings of the 25th IEEE International Symposium on High Performance Computer Architecture (HPCA).

Jahanshahi, Ali, Nanpeng Yu, and Daniel Wong. Under review. "PowerMorph: QoS-aware Server Power Reshaping for Data Center Regulation Service."

Makaratzis, Antonios T., Konstantinos M. Giannoutakis, and Dimitrios Tzovaras. 2018. "Energy modeling in cloud simulation frameworks." *Journal of Future Generation Computer Systems.* 79, 715–725.

Wang, Wei, Amirali Abdolrashidi, Nanpeng Yu, and Daniel Wong. 2019. "Frequency Regulation Service Provision by Data Center." *Applied Energy*. 251, 1-17.

Wang, Wei and Nanpeng Yu. 2017. "Phase Balancing in Power Distribution Network with Data Center." *Greenmetrics*. 1-6, Urbana-Champaign, IL.

Whitney, J. and P. Delforge. 2014. "Data Center Efficiency Assessment." *Natural Resources Defense Council (NRDC)*. IP-14-08-A.