Online Predictive Energy Scheduling Algorithm for Efficient Query Processing in Web Search Engines

[1] Karnati Susravanthi, [2] B. Narayana Reddy
[1] M.Tech.,(PG Scholar), [2]Assistant Professor
[1]Dept of CSE, [2]Dept of CSE,
[1] Sri Venkateswara Institute of Science and Technology, kadapa
[2] Sri Venkateswara Institute of Science and Technology, kadapa

Abstract : At present, web search is a common act in the everyday life of a lot of people. To achieve it on a large scale, Web companies' required energy-usage data center, which raise ecological & economical challenges. Web search engines are collected by number of query processing nodes, i.e., servers enthusiastic to practice user queries. Such lots of servers use a significant quantity of energy, mostly responsible to their CPUs, but they are essential to guarantee small latencies, as users are expecting sub-second response times. Parallel, such lots of servers use a significant quantity of energy, hindering the profitability of the search engines and raising environmental concerns. Since energy consumption has an significant role on the profitability & ecological impact of Web search engines, improving their energy efficiency is an important aspect. However, users can scarcely observe response times that are earlier than their expectations. Therefore, to mitigate energy consumption, Web search engines should answer queries no faster than user expectations. Hence, we introducing the Predictive Energy Saving Online Scheduling Algorithm (PESOS) to choose an suitable CPU frequency to execute a query on a per-core source and with help of proposed algorithm we can mitigate the energy consumption of servers' CPUs

IndexTerms **- Predictive Energy Saving Online Scheduling Algorithm (PESOS), CPUs, Dynamic Voltage and Frequency Scaling (DVFS).**

1. INTRODUCTION

Web search engines continuously crawl and index an im-mense number of Web pages to return fresh and relevant results to the users' queries. Users' queries are processed by query processing nodes, i.e., physical servers dedicated to this task. Web search engines are typically composed by thousands of these nodes, hosted in large datacenters which also include infrastructures for telecommunication, thermal cooling, fire suppression, power supply, etc [1]. This complex infrastructure is necessary to have low tail latencies (e.g., 95-th percentile) to guarantee that most users will receive results in sub-second times (e.g., 500 ms), in line with their expec-tations [2]. At the same time, such many servers consume a significant amount of energy, hindering the profitability of the search engines and raising environmental concerns. In fact, datacenters can consume tens of megawatts of electric power [1] and the related expenditure can exceed the original investment cost for a datacenter [3]. Because of their energy consumption, datacenters are responsible for the 14% of the ICT sector carbon dioxide emissions [4], which are the main cause of global warming. For this reason, governments are promoting codes of conduct and best practices [5], [6] to reduce the environmental impact of datacenters. Since energy consumption has an important role on the profitability and environmental impact of Web search engines, improving their energy efficiency is an important aspect. Noticeably, users can hardly notice response times that are faster than their expectations [2]. Therefore, to reduce energy consumption, Web search engines should answer queries no faster than user expectations. In this work, we focus onreducing the energy consumption of servers' CPUs, which are the most energy consuming components in search systems [1]. To this end, Dynamic Frequency and Voltage Scaling (DVFS) technologies [7] can be exploited. DVFS technologies allow to vary the frequency and voltage of the CPU cores of a server, trading off performance (i.e., longer response times) for lower energy consumptions. Several power management policies leverage DVFS technologies to scale the frequency of CPU cores accordingly to their utilization [8], [9]. However, core utilization-based policies have no mean to impose a required tail latency on a query processing node. As a result, the query processing node can consume more energy than necessary in providing query results faster than required, with no benefit for the users. In this work we propose the Predictive Energy Saving On-line Scheduling algorithm (PESOS), which considers the tail latency requirement of queries as an explicit parameter. Via the DVFS technology, PESOS selects the most appropriate CPU frequency to process a query on a per-core basis, so that the CPU energy consumption is reduced while respecting a required tail latency. The algorithm bases its decision on query efficiency predictors rather than core utilization. Queryefficiency predictors are techniques to estimate the processing time of a query before its processing. They have been proposed to improve the performance of a search engine, for instance to take decision about query scheduling [10] or query processing parallelization [11], [12]. However, to the best of our knowl-edge, query efficiency predictor have not been considered for reducing the energy consumption of query processing nodes. We build upon the approach described in [10] and propose two novel query efficiency predictor techniques: one to esti-mate the number of postings that must be scored to process a query, and one to estimate the response time of a query under a particular core frequency given the number of postings to score. PESOS exploits these two predictors to determine which is the lowest possible core frequency that can be used to process a query, so that the CPU energy consumption is reduced while satisfying the required tail latency. As predictors can be inaccurate, in this work we also propose and investigate a way to compensate prediction errors using the root mean square error of the predictors. We experimentally evaluate PESOS upon the TREC ClueWeb09 corpus and the query stream from the MSN2006 query log. We compare the performance of our approach with those of three baselines: perf [8], which always uses the maximum CPU core frequency, power [8], which throttles

CPU core frequencies according to the core utilizations, and cons [13], which performs frequency throttling according tothe query server utilization. PESOS, with predictors correc-tion, is able to meet the tail latency requirements while re-ducing the CPU energy consumption from ∼24% up to ∼44% with respect to perf and up to ∼20% with respect to cons, which however incurs in uncontrollable latency violations. Moreover, the experiments show that energy consumption can be further reduced by PESOS when prediction correction is not used, but with higher tail latencies. The rest of the paper is structured as follows: Section 2 provides background information about the energy consump-tion of Web search engine datacenters, the query processing activity, and the query efficiency predictors. Section 3 for-mulates the problem of minimizing the energy consumption of a query processing node while maximizing the number of queries which meet their deadlines. Section 4 illustrates our proposed solution to the problem, describes our query efficiency predictors, and the PESOS algorithm. Section 5 illustrates our experimental setup while Section 6 analyzes the obtained results. Related works are discussed in Section 7. Finally, the paper concludes in Section 8. II. BACKGROUND In this section we will discuss the energy-related issues in-curred by Web search engines (Sec. 2.1). Then, we will explain how query processing works and some techniques to reduce query response times (Sec. 2.2). Finally, we will discuss about query efficiency predictors, which we exploit to reduce theenergy consumption of a Web search engine while maintaining low tail latencies (Sec. 2.3). 2.1 Web search engine and energy consumption In the past, a large part of a datacenter energy consumption was accounted to inefficiencies in its cooling and power supply systems. However, Barroso et al. [1] report that modern datacenters have largely reduced the energy wastage of those infrastructures, leaving little room for further improvement. On the contrary, opportunities exist to reduce the energy consumption of the servers hosted in a datacenter. In par-ticular, our work focuses on the CPU power management of query processing nodes, since the CPUs dominate the energy consumption of physical servers dedicated to search tasks. In fact, CPUs can use up to 66% of the whole energy consumed by a query processing node at peak utilization [1]. Modern CPUs usually expose two energy saving mecha-nism, namely C-states and P-states. C-states represent CPU cores idle states and they are typically managed by the operating system [14]. C0 is the operative state in which a CPU core can perform computing tasks. When idle periodsoccur, i.e., when there are no computing tasks to perform, the core can enter one of the other deeper C-states and become inoperative. However, Web search engines process a large and continuous stream of queries. As a result, query processing nodes are rarely inactive and experience particularly short idle times. Consequently, there are little opportunities to exploit deep C-states, reducing the energy savings provided by the C-states in a Web search engine system [15], [16]. When a CPU core is in the active C0 state, it can op-erate at different frequencies (e.g., 800 MHz, 1.6 GHz, 2.1 GHz, . . .). This is possible thanks to the Dynamic Frequency and Voltage Scaling (DVFS) technology [7] which permits to adjust the frequency and voltage of a core to vary its perfor-mance and power consumption. In fact, higher core frequen-cies mean faster computations but higher power consumption. Vice versa, lower frequencies lead to slower computations and reduced power consumption. The various configurations of voltage and frequency available to the CPU cores are mapped to different P-states, and are managed by the operating system. Datacenters are buildings where multiple servers and communique tools are co-located because of their commonplace environmental necessities and bodily safety needs, and for ease of renovation. In that feel, a WSC is a sort of datacenter. Traditional datacenters, however, usually host a huge wide variety of highly small- or medium-sized packages, every running on a devoted hardware infrastructure this is de-coupled and protected from other systems within the same facility. Those datacenters host hardware and software program for more than one organizational gadgets or maybe distinct groups. Different computing systems within this type of datacenter often have common in terms of hardware, software, or protection infrastructure, and generally tend no longer to communicate with every other in any respect. Improving the high-quality of search outcomes often required coming up with state-of-the-art or highly-priced answers (e.g., storing greater facts within the inverted index or using gadget found out rating strategies), consequently growing query processing times. This, while coupled with the continuous boom of the indexable Web and the ever-growing query volumes of industrial search engines like Google and Yahoo in the last two decades, shifted a few research attentions to the efficiency of search systems. This line of research was regularly orthogonal to the aforementioned research on search result nice.

II LITERATURE SURVEY

William Lintner, Bill Tschudi & Otto VanGeet provides a top level view of excellent practices for energy-efficient records middle layout which spans the types of Information Technology (IT) systems & their environmental conditions, statistics center air management, cooling and electric systems, on-website era, and heat healing. IT system strength performance and environmental conditions are supplied first due to the fact measures taken in those regions have a cascading effect of secondary energy financial savings for the mechanical & electric systems. This concludes with a phase on metrics and benchmarking values by means of which a data center & its structures electricity performance can be evaluated. No design manual can offer 'the most energy-efficient' information middle design however the recommendations that observe provide guidelines that offer efficiency blessings for a huge type of data center situations.

Based on interpretation, the voltage scaling functionality of a processor helping Speed Step® generation is explored. The authors roposed DVS set of rules in existing and the proposed DVS algorithm, which combines the goodness of static EDF and appearance beforehand EDF algorithm, is carried out as a modular system inside the Linux kernel and overall performance is analyzed with the existing algorithms. The proposed set of rules achieves big electricity savings while retaining timeline ensures in comparison to the formerly proposed algorithms. Due to the modularity of the implementation, extra algorithms can be applied and confirmed the usage of the system. Dynamic pruning techniques can improve the efficiency of queries; however result in extraordinary queries taking specific quantities of time. Craig Macdonald, Nicola Tonellotto, Iadh Ounis empirically investigated the performance of various queries for in-memory inverted indices, and confirmed how and why the amount of pruning that would be carried out for a query can vary. Next, Craig Macdonald, Nicola Tonellotto, Iadh Ounis proposed a framework for predicting the efficiency of a query, which uses linear regression to research a mixture of aggregates of time period statistics. Experiments for 10,000 queries retrieving from an in-reminiscence index of the TREC ClueWeb09 series confirmed that our learned predictors encapsulating forty two functions ought to efficiently predict the reaction time of the modern-day Wand dynamic pruning retrieval strategy. Moreover, they proposed the net scheduling of queries across replicated query servers, driven by means of predicted response times of queries. Two exclusive scheduling architectures have been

proposed, differing within the area of the queuing. Simulation experiments showed not most effective the benefits of scheduling, but also the advantage of greater correct expected response instances within the scheduling algorithms. Saehoon Kim et.Al studies reducing the extreme tail latency of internet seek server by predicting and accelerating long strolling queries. They first show that the predictor needs to provide excessive keep in mind and appropriate precision as our prediction necessities. They propose a novel prediction framework, DDS, combining behind schedule prediction, dynamic capabilities and prediction errors estimation. In particular, DDS delays the prediction till we accumulate dynamic alerts that are exceedingly powerful on improving prediction accuracy. Moreover, DDS estimates both latency and blunders to selectively accelerate queries, achieving excessive don't forget target with true precision. Their simulation outcomes display that DDS efficiently reduces the tail latency & substantially improves server throughput.

III. PROPOSED WORK

A. Dynamic Voltage and Frequency Scaling

Dynamic Voltage and Frequency Scaling (DVFS) describes the use of two strength saving strategies (dynamic frequency scaling and dynamic voltage scaling) used to save strength in embedded structures which include mobile phones. This kind of power saving is different from what most folks commonly consider like standby or hibernate power states. All of these are useful of route. We can reduce the strength consumed via your embedded machine by lowering the frequency and/or voltage of the CPU and attached peripherals. Another benefit of reducing power intake is much less heat is generated by using your tool; this has blessings to the mechanical design. Done nicely it could make the distinction between wanting a passive or lively cooling machine. If we are able to keep away from a fan (or maybe lessen the fan pace) we can improve more than a few of things such as cost in keeping with tool and imply time to failure. The pleasant aspect is hardware companies had been adding that electricity saving abilities to gadgets so a part of the paintings is already performed for us, we do but want to apprehend our structures requirements.

B. Proposed Work

We propose the Predictive Energy Saving Online Scheduling algorithm (PESOS), which considers the tail latency requirement of queries as an express parameter. Via the Dynamic Frequency and Voltage Scaling (DVFS) era, PESOS selects the most suitable CPU frequency to method a query on a in keeping with-middle foundation, in order that the CPU power consumption is reduced even as respecting required tail latency. The algorithm bases its choice on query efficiency predictors in place of center usage. Query performance predictors are strategies to estimate the processing time of a query before its processing. In this paper we attention on lowering the CPU strength consumption of single query processing nodes, independently of the followed partition method. A query processing node is a physical server composed by numerous multi-core processors/CPUs with a shared memory. A query server system is achieved on pinnacle of each of the CPU middle of the processing node. All query servers get admission to a shared inverted index held in major reminiscence to method queries.

Each query server manages a queue, wherein the incoming queries are stored. The first query in the queue is processed as soon because the corresponding CPU center is idle. The queued queries are processed following the primary-come first served coverage. The number of queries in a query server's queue represents the server load. Queries arrive to the processing node as a circulation S = q1. . . qn. When a query reaches the processing node it's far dispatched to a query server by a query router. The query router dispatches an incoming query to the least loaded query server, i.e., to the server with the smallest variety of enqueued queries as shown in Fig.1. Alternatively, the query processing node could have a single query queue and dispatch queries from the queue to idle query servers. In this work, we use a queue for each query servers considering that a single queue will no longer permit to take nearby selections approximately the CPU middle frequency to apply for the relative query server.

C. Root Mean Square Error (RMSE)

The Root Mean Square Error (RMSE) is an often used measure of the differentiation among values predicted by a model & the values actually captured from the environment that is being modeled. These individual differentiations are also referred residuals, & the RMSE provides to aggregate them into a single measure of predictive power.

IV METHODOLOGY

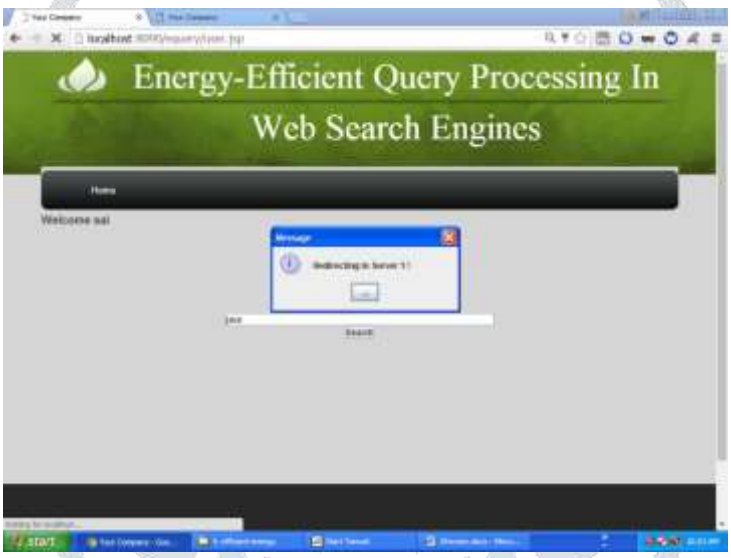
Web information systems such as search engines have to keep up with the growth and changes of the Web. For marketing, research and in business: To get a better handle on search engine optimization, it's important to understand why people use search engines, at all. Generally, people use search engines for one of three things: research, shopping, or entertainment. A query processing node is a physical server composed by several multi-core processors/CPUs with a shared memory which holds the inverted index. The inverted index can be partitioned into shards and distributed across multiple query processing nodes. In this work, we focus on reducing the CPU energy consumption of single query processing nodes, independently of the adopted partition strategy. In the following, we assume that each query processing node holds an identical replica of the inverted index [24].

Data Uploaded Scuccessfully

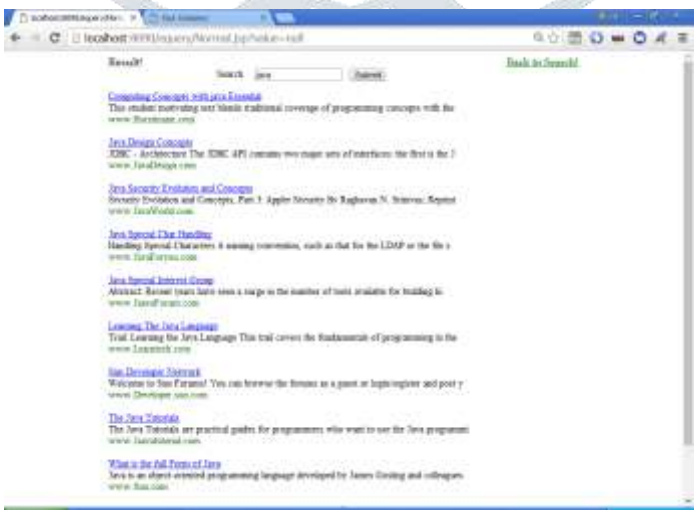
Server Status Page

User Query Search Page

User Search for a Keyword



Redirecting to Server 1



Query Result Page

Server Status Page

Queries executed at server

V. CONCLUSION

Finally, we introduced an energy efficient algorithm which is called Predictive Energy Saving Online Scheduling Algorithm and by implementing this algorithm, we can minimize the CPU energy usage on query processing node in the circumstance of web search engines. The presented algorithm worked by using CPU Dynamic Voltage & Frequency Scaling technique. By applying query efficiency predictors on query we can estimate the query processing time in this paper.

REFERENCE

[1]L. A. Barroso, J. Clidaras, and U. H¨olzle, The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines, 2nd ed. Morgan & Claypool Publishers, 2013.
[2]I. Arapakis, X. Bai, and B. B. Cambazoglu, “Impact of response latency on user behavior in web search,” in Proc. SIGIR, 2014, pp. 103–112.
[3]The Climate Group for the Global e-Sustainability Initiative, “Smart 2020: Enabling the low carbon economy in the information age,” 2008.
[4]U.S. Department of Energy, “Best Practices Guide for Energy-Efficient Data Center Design.” [Online].
[5]D. C. Snowdon, S. Ruocco, and G. Heiser, “Power Management and Dynamic Voltage Scaling: Myths and Facts,” in Proc. of Workshop on Power Aware Real-time Computing, 2005.
[6]D. Brodowski, “CPU frequency and voltage scaling code in the Linux kernel.”
[7]C. Macdonald, N. Tonellotto, and I. Ounis, “Learning to predict response times for online query scheduling,” in Proc. SIGIR, 2012, pp. 621–630.
[8]M. Jeon, S. Kim, S.-w. Hwang, Y. He, S. Elnikety, A. L. Cox, and S. Rixner, “Predictive parallelization: Taming tail

latencies in web search," in Proc. SIGIR, 2014, pp. 253–262.
[9]S. Kim, Y. He, S.-w. Hwang, S. Elnikety, and S. Choi, "Delayeddynamic-selective (dds) prediction for reducing extreme tail latency in web search," in Proc. WSDM, 2015, pp. 7–16
[10]M. Catena, C. Macdonald, and N. Tonellotto, "Loadsensitive cpu power management for web search engines," in Proc. SIGIR, 2015, pp. 751–754.
[11]V. Pallipadi, S. Li, and A. Belay, "cpuidle: Do nothing, efficiently," in Proc. Linux Symposium, vol. 2, 2007, pp. 119–125.
[12]D. Lo, L. Cheng, R. Govindaraju, L. A. Barroso, and C. Kozyrakis, "Towards energy proportionality for large-scale latency-critical workloads," in Proc. ISCA, 2014, pp. 301–312.