Utilization-Aware Redirection Policy in CDN: A Case for Energy Conservation

Saif ul Islam[1], Konstantinos Stamos[2], Jean-Marc Pierson[1], and Athena Vakali[2]

[1] IRIT, University of Toulouse
118 Route de Narbonne, F-31062 Toulouse CEDEX 9, France
[2] Aristotle University of Thessaloniki
54124, Thessaloniki, Greece
{islam,pierson}@irit.fr
{kstamos,avakali}@csd.auth.gr

Abstract. Due to the gradual and rapid increase in Information and Communication Technology (ICT) industry, it is very important to introduce energy efficient techniques and infrastructures in large scale distributed systems. Content Distribution Networks (CDNs) are one of these popular systems which try to make the contents closer to the widely dispersed Internet users. A Content Distribution Network provides its services by using a number of surrogate servers geographically distributed in the web. Surrogate servers have the copies of the original contents belonging to the origin server, depending on their storage capacity. When a client requests for some particular contents from a surrogate server, either this request can be fulfilled directly by it or in case of absence of the requested contents, surrogate servers cooperate with each other or with the origin server. In this paper, our focus is on the surrogate servers utilization and using it as a parameter to conserve energy in CDNs while trying to maintain an acceptable Quality of Experience (QoE).

Keywords: CDNs, Energy conservation, QoE.

1 Introduction

The investigation of new techniques and technologies for the protection of environment is considered one of the most prominent and urgent issues of the 21st century [1]. The governments and environmental agencies are active to face the challenge of global warming. Internet traffic is increasing very rapidly. Internet vendors are obliged to enlarge their networks to provide their services to a maximum number of users. This change caused a trend towards the grand, geographically distributed systems. These systems can have a huge amount of servers and many data centers. In order to provide quick and better services, Internet vendors are forced to install energy-hungry devices to cope with the intensive requirements for the traffic e.g. real time media [2]. Industries and customers are interested in less cost options. Reduction in energy consumption may play an important role to decrease over all cost [2].

D. Kranzlmüller and A M. Tjoa (Eds.): ICT-GLOW 2011, LNCS 6868, pp. 180–187, 2011.

A popular type of such a network is a Content Distribution Network (CDN) [3] which is responsible to manage large amounts of content traffic originating from Web users through geographically distributed set of servers. With this approach, content is located near to the user yielding low response time and high content availability since many replicas are distributed. The origin server is relieved from requests since the majority of them are handled by the CDN, whereas, Quality of Service (QoS) and efficiency are guaranteed in a scalable way.

Our key motivation lies on finding a delicate balance between users' satisfaction and reduction in infrastructure energy consumption. We aim at defining an energy-aware forwarding strategy that enhance previous work [3] for energy savings, relying on utilization model of the surrogate servers. In the next section, we discuss some previous related work. Section 3 describes the utilization of the surrogate servers, a model proposed to calculate the surrogate servers' utilization and our proposed energy-aware policy for the redirection of the client requests to the surrogate servers. In section 4, simulation testbed and results are presented. Section 5 concludes the paper.

2 Related Work

In CDN setup, several issues and decisions are involved to manage and distribute the contents. Till now, different policies have been defined in order to distribute the contents in CDNs. In [4], cooperative and uncooperative push-based policies are presented. [5], [6], [7] describe cooperative and uncooperative pull-based policies. In cooperative policies surrogate servers cooperate with each other on the cache miss while in uncooperative policies surrogate servers don't cooperate with each other. In push-based policy, the content is pushed from the origin server to the surrogate servers. In pull-based policy, clients' requests are directed (through DNS redirection) to their closest (in terms of geographic proximity or load, etc.) surrogate server.

In recent years, substantial research is carried out to propose and to develop energy-aware solutions in networks. Some of them can be described as follows, a) dynamically changing the link rate, adapting to the utilization of the network, in order to reduce energy consumption [8], b) to put the idle end devices in sleeping mode and to use proxy to maintain Internet connectivity [1], c) diverting the network traffic towards fewer links during less activity period and to enable, the network devices (e.g. routers and switches) connected to the idle links, to sleep [9], d) frequency change and Dynamic Voltage Scaling (DVS) for energy reduction of integrated circuits [10], e) greening P2P protocols i.e. Green BitTorrent [11], f) to change the network architecture for energy-efficient content dissemination e.g. Content Centric Networking (CCN) [12]. Our work is related to the above cited works. Main inspiration is taken from the concept to divert the load to the fewer devices to reduce the energy consumption.

3 Surrogate Servers' Utilization and Energy

3.1 Surrogate Servers' Utilization

In a CDN, when a client sends a request for some particular contents, the request is forwarded to a surrogate server according to the redirection policy. When a surrogate server s_1 receives a request for an object from client c_1. s_1 locks a resource. It checks for the demanded object in its cache. If s_1 has the requested object in the cache, it sends the contents to the client c_1 and unlocks the resource. In other case, if s_1 doesn't have the requested contents in its cache, it can get the object from another surrogate server s_2 or from origin server (depending upon the redirection policy). At the reception of the requested object, surrogate server s_1 stores the object in its cache and sends it to the client c_1.

We first propose a simple utilization model based on computing the connections duration that reflects the usage of the server over the time.

3.2 Utilization Model

Our utilization model for surrogate servers is composed of three main parts: a) Dark areas b) Maximum number of connections c) Total execution time

Dark Areas. As shown in Figure 1, surrogate server s_1 gets three locks (connections) l_1, l_2 and l_3 because of requests for some contents from the clients at time t_1, t_2 and t_3 respectively. Then at time t_4, t_5 and t_6, server s_1 gets unlocks u_1, u_2 and u_3 respectively and then it doesn't have any lock until it gets a lock at t_7 and this lock is unlocked at t_8. So as shown in the Figure 1, DA_1 (Dark Area One) can be calculated as $DA_1 = (t_2-t_1)1+(t_3-t_2)2+(t_4-t_3)3+(t_5-t_4)2+(t_6-t_5)1$ and the DA_2 can be calculated as $DA_2 = (t_8 - t_7)1$.

Finally, the surrogate servers' utilization can be written as follows in the equation form,

$$Us_i = \frac{\Sigma_j \; DA_j}{LockMax_i \times T} \tag{1}$$

Where Us_i is the utilization of the surrogate server s_i, DA_j is the dark area j and $LockMax_i$ are the maximum number of locks server s_i can have at the same time and T the total execution time. From this utilization model one could derive an energy consumption model linking the utilization to the consumed energy.

3.3 Energy-Aware CDNs Redirection

In this context, the purpose is to identify the following research question and to introduce the means of exploring possible solutions. How a CDN can redirect the users' requests for content to its surrogate servers in such way that the energy consumption is minimized while trying to maintain an acceptable Quality of Service (QoS)? In order to answer this question, a set of discrete milestones have been achieved, starting from theoretical definitions leading to actual implementations.

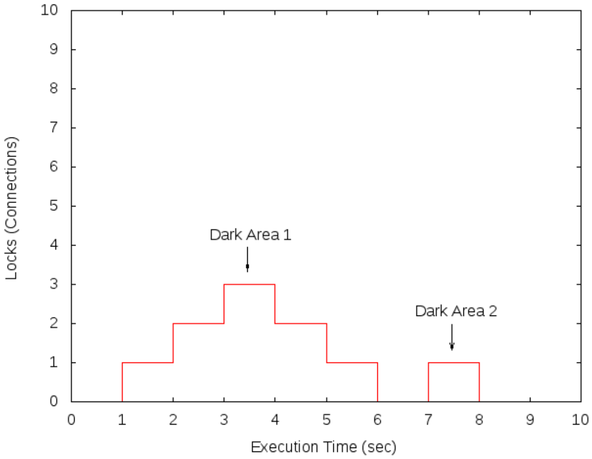

Fig. 1. Surrogate server's utilization

The traditional approach is to shut down the under-utilized surrogate servers in order to reduce the energy consumption. The logic behind this statement is that not all surrogate servers are usually necessary in normal traffic conditions. Therefore we can approach the problem in a manner that causes purposely under-utilization in surrogate servers. When some surrogate servers are underutilized they can be switched-off completely or their processor frequency can be adjusted accordingly. This leads to the need of defining a tunable mechanism for causing such an under-utilization.

Proposal of Client Redirection Policy. The following client redirection policy has been defined. We consider the Zipfian distribution with the parameter $z \in \{0, .., 1\}$. For the value 0 we get the uniform distribution and for the value 1 we get an exponential distribution where only a small percentage gathers the majority of the distribution. Then, the client redirection algorithm works like this a) sort the surrogate servers by their current utilization, b) set the parameter z, and c) pick a random surrogate server according to a probability drawn from the respective Zipfian distribution with slope parameter z. The obvious advantages of the proposed method are the generation of under-utilized servers and the ability to smoothly balance the energy consumption vs. the surrogate servers availability.

4 Simulation Testbed and Results

We used CDNsim [3] as a simulation environment that simulates a main CDN infrastructure and is implemented in the C++ programming language. It is based on the OMNeT++ library which provides a discrete event simulation environment. All CDN networking issues, like surrogate server selection, propagation, queuing, bottle-necks and processing delays are computed dynamically via

CDNsim, which provides a detailed implementation of the TCP/IP protocol, implementing packet switching, packet retransmission upon misses, freshness, etc. One of the central features of CDNsim is the ability to add new client redirection policies which is fitting in our case.

In this case, we consider the 100 geographically distributed homogenous CDN surrogate servers where each server is capable of handling 500 connections simultaneously. We used a real Internet topology of AS-level, having 3037 routers, that consists of routing data collected from 7 BGP peers dispersed at different locations. A synthetic but realistic website having 50000 objects of 1GB total size, is generated. A generator is used to generate requests stream that shows the access patterns close to realistic ones. Table 1 shows the summary of the parameters used in two sets of experiments.

Table 1. Summary of simulations parameters

Parameter	Experiment 1	Experiment 2
Website	50000 objects, size 1GB	50000 objects, size 1GB
Number of requests	1000000	1000000
Mean interval time of requests	0.01sec	0.01sec
Distribution of the interval time	exponential	exponential
Link speed	6Mbps	6Mbps
Network topology backbone	type AS, 3037 routers	type AS, 3037 routers
Number of surrogate servers	100	100
Number of client groups	100	100
Number of contetn providers	1	1
Cache size percentage of the website's size	40%	40%
Load-unbalancing paramter z value	0, 1	0, 0.25, 0.50, 0.75, 1

4.1 Surrogate Servers Utilization vs. Load-Unbalancing

Discussion. For simulation setup, see experiment 1 in Table 1. Figure 2 and 3 describe the relation between the load-unbalancing parameter z (see Section 3.3) and the utilization of the surrogate servers. The x-axis presents the surrogate servers from 1 to 100 and axis y shows their average utilization. The value 0 of the load-unbalancing parameter z shows the uniform distribution and the requests are sent to the servers randomly as shown in Figure 2. It shows no peaks and most of the utilization values almost reside in the same region (about $4 - 10\%$ utilization). Figure 3 shows the extreme of this pattern as it has the maximum value of the load-unbalancing parameter z, where only a small number of surrogate servers get most of the requests as shown the high peaks in the start. The surrogate servers with more load have the more probability to get the requests and they become the bottle necks. It is noticed that with the increase in the parameter z value there is an increase in the utilization of a small percentage of the surrogate servers and decrease in the utilization of the rest ones. Our purpose is to get the under-utilized servers that gives a way to massive energy savings: Figure 3 shows that 80% of the surrogate servers have less than 5% utilization and can be considered to be powered-off, after serving the present

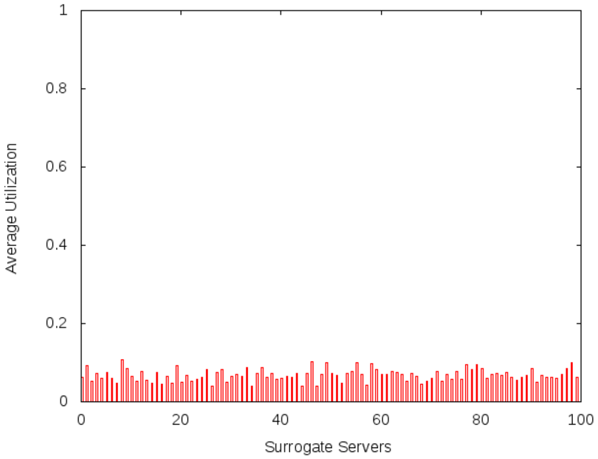

Fig. 2. Surrogate servers utilization vs. Load-unbalancing parameter $z = 0$

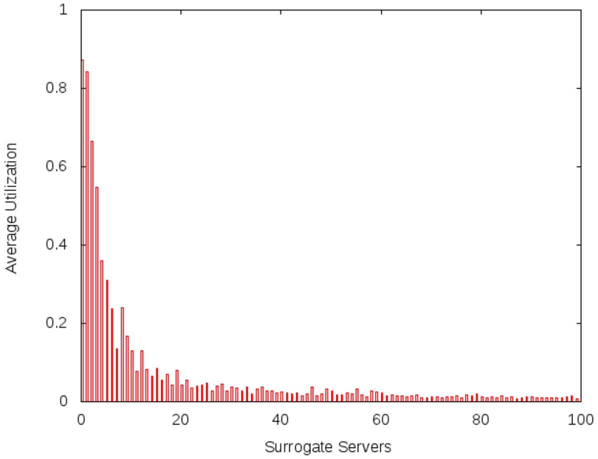

Fig. 3. Surrogate servers utilization vs. Load-unbalancing parameter $z = 1$

requests and the frequency of the processors of the other 20% servers (which have the utilization from 5% and 90%) can be adjusted according to the load.

4.2 Load-Unbalancing vs. Mean Response Time

Discussion. For dataset see Table 1, Experiment 2. Figure 4 shows the relation between mean response time and the load-unbalancing parameter z. It shows how the change in load-unbalancing parameter z value affects the time for the clients to get their requested contents. Axis x presents the different values of load-unbalancing parameter z from 0 to 1, with a step of 0.25. The y-axis

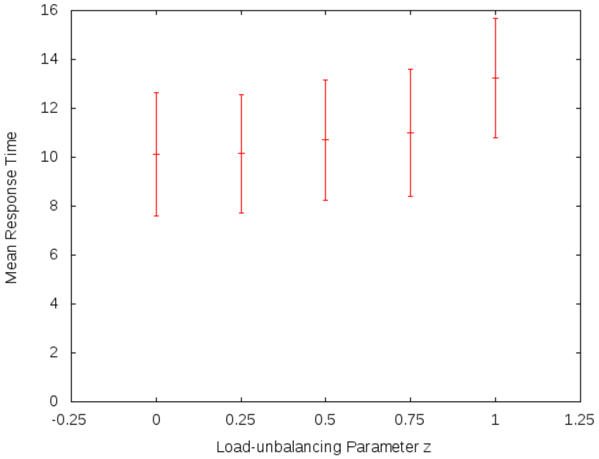

Fig. 4. Load-unbalancing parameter z vs. Mean response time

shows the mean response time values. Not surprisingly, it can be noticed in the Figure 4 that with the increase in the value of the load-unbalancing parameter z, the mean response time is increased. For the values of z from 0 to 0.25, the difference is very small but as the z value increases mean response time also increases. So by saving energy to make the surrogate servers underutilized we have a cost of increase in response time. Knowing which level of satisfaction or degradation the user accepts would guide us towards an acceptable value of the load-unbalancing parameter z.

5 Conclusion and Future Work

This paper has presented that there is a significant potential to save energy in Content Distribution Networks. Surrogate servers' utilization can be used as a parameter to cope with the energy savings opportunities in CDNs. We show that energy savings can be achieved by diverting the load to fewer surrogate servers with a small penalty of performance in increased response time.

Our next step is to enable the underutilized surrogate servers to be switched-off dynamically and to adapt the frequency of processors of the active surrogate servers according to the load. Also we will propose and implement an energy consumption model in the context of Content Delivery Networks.

References

1. Jimeno, M., Christensen, K., Nordman, B.: A Network Connection Proxy to Enable Hosts to Sleep and Save Energy. In: IEEE International Performance, Computing and Communications Conference, Austin, Texas, pp. 101–110 (2008)

2. Vasic, N., Kostic, D.: Energy-aware traffic engineering. In: Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking, pp. 169–178 (2010)
3. Stamos, K., Pallis, G., Vakali, A., Katsaros, D., Sidiropoulos, A., Manolopoulos, Y.: CDNsim: A Simulation Tool for Content Distribution Networks. ACM Transactions on Modeling and Computer Simulation 20, 10:1–10:40 (2010)
4. Pallis, G., Vakali, A.: Insight and perspectives for content delivery networks. Communications of the ACM 49, 101–106 (2006)
5. Annapureddy, S., Freedman, M.J., Mazires, D.: Shark: Scaling File Servers via Cooperative Caching. In: Proceedings of the 2nd USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI), pp. 129–142. IEEE, Los Alamitos (2005)
6. Vakali, A., Pallis, G.: Content Delivery Networks: Status and Trends. IEEE Internet Computing 7, 68–74 (2003)
7. Yu, H., Vahdat, A.: Minimal replication cost for availability. In: Proceedings of the twenty-first annual symposium on Principles of distributed computing, Monterey, California, pp. 98–107 (2002)
8. Anand, H., Reardon, C., Subramaniyan, R., George, A.D.: Ethernet Adaptive Link Rate (ALR): Analysis of a MAC Handshake Protocol. In: Proceedings of the 31st IEEE Conference on Local Computer Networks, Tampa, FL, pp. 533–534 (2006)
9. Gupta, M., Singh, S.: Greening of the Internet. In: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, Karlsruhe, pp. 19–26 (2003)
10. Zhai, B., Blaauw, D., Sylvester, D., Flautner, K.: Theoretical and practical limits of dynamic voltage scaling. In: Proceedings of the 41st annual Design Automation Conference, San Diego, CA, pp. 868–873 (2004)
11. Blackburn, J., Christensen, K.: A Simulation Study of a New Green BitTorrent. In: Proceedings of the First International Workshop on Green Communications, Dresden, pp. 1–6 (2009)
12. Lee, U., Rimac, I., Hilt, V.: Greening the internet with content-centric networking. In: Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking, Passau, pp. 179–182 (2010)