International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

A Survey on an Optimizing Cost and Performance for Multihoming

Rashmi Rajendra Kurkure[1], Madhuri Zawar[1]

[1, 2]Department of Computer Engineering, GF"s foundation Godavari College of Engineering, Jalgaon, Maharashtra, India

Abstract: ***Multihoming is often used by large enterprises and stub ISPs to connect to the Internet. In this paper, we design a series of novel smart routing algorithms to optimize cost and performance for multihomed users. We evaluate our algorithms through both analysis and extensive simulations based on realistic charging models, traffic demands, performance data, and network topologies. Our results suggest that these algorithms are very effective in minimizing cost and at the same time improving performance. We further examine the equilibrium performance of smart routing in a global setting and show that a smart routing user can improve its performance without adversely affecting other users.***

Keywords: Multihoming, Smart Routing, Optimization, Algorithms

1. Introduction

Many content publishers on the Internet use multiple content distribution networks (CDNs) to distribute and cache their digital content. We refer to content publishing using multiple content distribution networks as *content multihoming*. In our recent survey, we found that all major content publishers such as Netflix, Hulu, Microsoft, Apple, Face book, and MSNBC use content multihoming.

Content publishers adopt content multihoming to aggregate the diversity of individual CDN providers on features, performance and commitment [7]. For example, one CDN may provide good coverage for locations 1 and 2, whereas another CDN provides good coverage for locations 2 and 3. To deliver content to viewers from all three locations, a content publisher may need to use both CDNs. Given the wide usage and potential benefits of content multihoming, many commercial systems supporting content multihoming have recently been deployed, so that more content publishers can benefit from content multihoming. However, these commercial products either use ad hoc approaches or do not provide details on their designs. No previous studies on how to effectively utilize content multihoming are known. In this paper, we attempt to provide a framework and a set of novel algorithms to optimize the benefits of content multihoming.

Given that content multihoming allows a content object to be delivered from multiple CDNs, which CDN(s) should a content publisher use to deliver each object to each content viewer requesting this object, so that the publisher optimizes its benefits from content multihoming? This question is the key to efficiently utilizing content multihoming, since its solutions can be implemented directly with the flexible request routing mechanisms (*e.g.*, DNS CNAME, HTTP Redirect from servers, and client scripts in end hosts) in modern content delivery infrastructures. An answer to this simple question, however, is not immediately obvious. Consider the current common approach of choosing, for each content viewer, the best performing CDN among all candidate CDNs. This approach, despite its simplicity, has multiple issues.[1] First, although the chosen CDN may provide the highest level of performance, for example, satisfying that 99% viewers do not see quality of experience (QoE) degradation, the cost of the chosen CDN can be much higher than another CDN with a slightly lower, but still high enough level of 95%. Second, there are often multiple CDNs with comparable and sufficient levels of performance at a given region, *e.g.*, in US. One common approach to break ties in such cases is to pick the CDN with the lowest cost. However, the costs of CDNs, in particular, pay-as-you-go CDNs such as Amazon Cloud Front, are volume based and non-linear. The cost of one object assignment depends on the other assignments. Third, there are locations where even the best performing CDN falls short. For example, a content publisher may have a QoE target of 95%, but the best performing CDN at some location achieves only 90%[2].

In this paper, we answer the preceding question by designing two algorithms: (1) an efficient optimization algorithm executing at content publisher to compute content distribution guidance, and (2) a simple algorithm executing at individual content viewers to follow the guidance with local adaptation. Either algorithm can be deployed alone, but together they benefit the most. Specifically, the publisher optimization algorithm, named CMO, computes CDN assignments considering many real factors: nonlinear, multi-region CDN traffic charging, per-request charging, content licensing restrictions, CDN feature availability, and CDN performance variations. The CMO algorithm is novel and highly efficient. For example, when considering traffic cost only, the complexity of CMO is polynomial and independent of the number of content objects, whereas the complexity of simple enumeration is exponential in the number of content objects.[3]

The local viewer algorithm provides a capability for a content viewer to make efficient usage of multiple servers from multiple CDNs, with a preference ordering on the usage of CDN edge servers provided by the content publisher. Inspired by TCP AIMD and using a simple prioritized assignment mechanism, the algorithm adapts the usage of multiple CDNs, achieving a performance level that no single CDN/server can achieve alone.

Providers of online services such as search, maps, and instant messaging are experiencing an enormous growth in demand. Google attracts over 5 billion search queries per month [2], and Microsoft‟s Live Messenger attracts over 330 million active users each month [5]. To satisfy this global demand, online service providers (OSPs) operate a network of geographically dispersed data centers and connect with many Internet service providers (ISPs). Different users interact with different data centers, and ISPs help the OSPs carry traffic to and from the users.

Two key considerations for OSPs are the cost and the performance of delivering traffic to its users. Large OSPs such as Google, Microsoft, and Yahoo! send and receive traffic that exceeds a petabyte per day. Accordingly, they bear huge costs to transport data.

While cost is clearly of concern, performance of traffic is critical as well because revenue relies directly on it. Even small increments in user-experienced delay (e.g., page load time) can lead to significant loss in revenue through a reduction in purchases, search queries, or advertisement click-through rates [20]. Because application protocols involve multiple round trips, small increments in path latency can lead to large increments in user-experienced delay. The richness of OSP networks makes it difficult to optimize the cost and performance of traffic. There are numerous destination prefixes and numerous choices for mapping users to data centers and for selecting ISPs. Each choice has different different cost and performance characteristics. For instance, while some ISPs are free, some are exorbitantly expensive. Making matters worse, cost and performance must be optimized jointly because the trade-off between the two factors can be complex. We show that optimizing for cost alone leads to severe performance degradation and optimizing for performance alone leads to significant cost [10].

2. Related Work

Several recent studies have shown that Internet routing often yields sub-optimal user performance, *e.g.*, [4, 27, 32, 33]. There are a number of contributing factors, including routing hierarchy, policy routing, and slow reaction (if any) to transient network congestion or failures. BGP routing instabilities can further exacerbate the problem. These observations have generated considerable research interest in offering end-users more control in route selection. For instance, the authors in [4, 27] propose using overlay routing to improve user performance. Achieving a large scale deployment with this approach is challenging, as cooperation among multiple organizations is not easy to arrange in practice. Multihoming is an alternative way to enable users to control routes. Many large enterprises, stub ISPs and even small businesses already use multihoming as a way to connect to the Internet.

Much of the previous work on multihoming focuses on how to design protocols to implement multihoming, *e.g.*, [5, 7, 11, 30]. For example, the authors of [5, 7, 12, 24, 30] use BGP peering as an implementation technique. Another technique is through DNS or NAT, which is used in [9, 21]. Our work differs from the above in that we do not focus on the implementations, but instead on designing algorithms to determine when and how much traffic a user should assign to different ISPs to optimize both performance and cost. Consequently, our work is complementary to the above. There are several papers that evaluate the benefits of smart routing, including [8, 28, 29]. More recently, Akella *et al.* [1] quantify the potential performance benefits of multihoming using real Internet traces. Their results show that smart routing has the potential to achieve an average performance improvement of 25% or more for a 2-multihomed user in most cases, and that most of the benefit can be achieved using 4 providers. Motivated by these results, we seek to develop routing schemes to achieve such benefits in practice. In addition, we study the effects of un-coordinated route optimization by multiple mutually interfering smart routing users.

Finally, there are a few research studies on designing algorithms for smart routing, *e.g.*, [1, 15, 17]. For example, Orda and Rom [17] investigate where to place multihomed users and show that the problem is NP-hard. Cao *et al.* [6] propose using hash functions to achieve load balancing among multiple links. In [11], the authors compare several route selection schemes in a local area network and show that hashing can achieve performance comparable to load-sensitive route selection. Our work differs from these studies in that we use both cost and network performance as metrics of interest. We also study the interactions between multiple smart routing users, and between smart routing and single-homed users.

3. Comic Framework

We give an overview of our proposed COMIC framework to optimize the sum of electricity costs for data centers and usage costs for CDNs through content multihoming. The entities in the COMIC framework mainly include four parts: user group, data, CDN and data center.

1) **User group:** A user group is the set of users who are represented by the one and same identity in the COMIC framework. For example, the users physically close to each other and served by the same regional ISP may be aggregated into a user group. In the following section, we do not distinguish between user and user group.
2) **Data:** Data are the collection of content objects that the users request through the Internet. A content provider can have several types of content objects, such as a video in an online video website or an email or a message. Note that there are both static content (e.g., images) and dynamic content (e.g., PHP) in a request. However, the static content is usually cached in the replica server of CDNs, we only consider the dynamic content in data center.
3) **CDN:** a CDN is a large distributed system of servers, which replicates contents originated from data centers, and delivers the locally stored contents to end-users through network. In small geographical regions, such as a city, the pricing function of a CDN may be the same. However, different geographical regions in a CDN or the same geographical region in different CDNs may charge differently. We use charging region to refer to the geographical region which has the same pricing function within a certain CDN.
4) **Data center:** a data center is a collection of servers in a certain location to serve the data requests from users. Note that the electricity cost of a data center depends on

not only the size of arrival requests, but also the local real-time electricity price in the modern power grid.

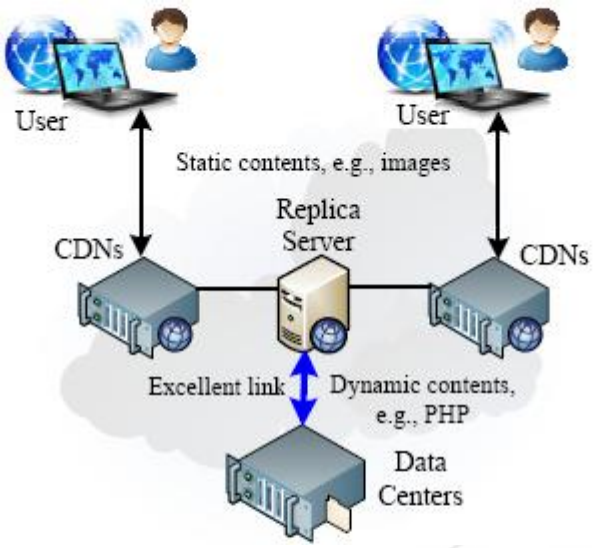

Figure 1: Procedures to Serve End-user‟s Requests in a CDN

As shown in Fig. 1, a CDN is used to deliver both static and dynamic content over a vast network infrastructure with excellent links. When an end-user requests a content service, the request will be redirected from the originating site‟s server to a server in one of CDNs. When a request comes into an edge server in a selected CDN, the static content is served directly from it. On the other hand, the dynamic content is not in the cache, then the edge server in the CDN makes a request to an origin server in one of data centers. We can note that there are two selections in the operation of CDN for handling dynamic contents: one is to select a CDN for delivering content service; the other is to select a data center for requesting dynamic content. Normally, the closest CDN is selected for delivering the content and the arrival workload is equally distributed among data centers. However, the simple strategies are usually not cost-efficient.

In this paper, the COMIC framework is proposed which runs a cost-aware optimization algorithm to choose which data center for content generation and which CDN for content delivery. Note that the optimization algorithm is a centralized solution which depends on all the electricity prices of data center and all the CDN usages prices at the same time. To this end, the usage cost functions of CDNs are provided by CDN providers and the real-time electricity prices are automatically reported by smart meters installed in electricity grids.

4. Algorithm

a) Two Time Scale Control algorithm

The main goal is to provide a unifying framework that allows one to exploit power cost reduction opportunities across all these levels. Moreover, the *non-work-conserving* nature of our framework allows us to take advantage of the temporal volatility of power prices while offering an *explicit* trade off between power cost and delay.

At different time instances, workload arrives at the front end proxy servers which have the flexibility to distribute this workload to different back end clusters. The back end clusters receive the workload from front end servers and have the flexibility to choose when to serve that workload by managing the number of activated servers and the service rate of each server. The problem then is to make the following three decisions, with the objective of reducing power cost: (i) how to distribute the workload from the front end servers to the back end clusters, (ii) how many servers to activate at each back end cluster at any given time, and (iii) how to set the service rates (or CPU power levels) of the back end servers.

This algorithm exploits temporal and spatial variations in the workload arrival process (at the front end servers) and the power prices (at the back end clusters) to reduce power cost. It also facilitates a *cost vs. delay* trade-off which allows data center operators to reduce power cost at the expense of increased service delay. Hence, our work is suited for *delay tolerant workloads* such as massively parallel and data intensive MapReduce jobs. Today, MapReduce programming based applications are used to build a wide array of web services – e.g., search, data analytics, social networking, etc. Hence, even though our proposed solution is more effective for delay tolerant workloads it is still relevant to many current and future cloud computing scenarios.

b) Stochastic Subgradient based trough filling algorithm

Intelligent trough filling needs to accommodate the following issues. First, the overall capacity of a datacenter is likely to be random, e.g., due to server failure. Second, capacity demand of delay-sensitive jobs (DSJs),, such as Internet requests, varies due to dynamic load. Given the higher priority of DSJs, available capacity for delay tolerant jobs (DTJs) is random and hard to predict or learn in statistics. Meanwhile, the demand of DTJs is also likely to be dynamic.

Further, in order to consider a set of geographically distributed Internet-scale datacenters (IDC), there are additional constraints. First, load shifting is constrained by the bandwidth available between IDCs. In our setting, similar to capacity, bandwidth is prioritized for shifting DSJs, and thus results in a random „residual bandwidth‟ for DTJs. Second, electricity prices diversity and dynamics bring challenges as well as opportunities, e.g., in price-aware load shifting, in the context of trough filling. Third, due to heterogenous service agility, different classes of DTJs may require different sets of IDCs. Moreover, different IDCs maybe heterogenous in service rates and energy consumption for each type of DTJs. We consider these issues and address the above challenges in this paper. The goal is to design intelligent trough filling mechanisms, that achieve both energy efficiency and good delay performance. We design joint dynamic speed scaling and load shifting schemes.

A stochastic subgradient based trough filling scheme is proposed, named SSTF, with the objective of minimizing energy and shifting cost while stabilizing the DTJ queues. The proposed algorithm does not need underlying probability of system states, which is usually difficult to estimate.

Table 1: Comparative Study

Technique	Table Column Head		
	Working principle	Advantages	Disadvantages
Two time scale control algorithm	Power cost reduction in geographically distributed data centre"s	Power cost is reduced at data center while handling delay tolerant workloads	Usage cost for CDN"s is not managed
Stochastic subgradient based trough filling algorithm	Solve a convex optimization problem for capacity allocation and load shifting in each slot	Optimal cost is achieved	Delay is very large

5. Conclusion

In this paper, we design a series of novel smart routing algorithms to optimize cost and performance for multihomed users. Using both analysis and extensive simulations based on realistic traces, we show that our algorithms are very effective in minimizing cost and improving performance. We further examine the global effects of smart routing using simulations based on realistic topologies and traffic. Our results show that under traffic equilibria smart routing can improve performance without hurting other traffic. There are several avenues for future work. In this paper, we focus on algorithmic design and evaluation through analysis and simulation. A natural next step is to implement the algorithms and conduct experiments in the Internet.

6. Acknowledgment

I would like to express my sincere thanks to my guide Prof. Madhuri Zawar & H.O.D. Prof. Pramod Gosavi for his motivation and useful suggestions which truly helped me in improving the quality of this paper and heartily thankful to IJAERD for giving me such a wonderful opportunity for publishing my paper.

References

[1] V. Adhikari, Y. Guo, F. Hao, V. Hilt, and Z.-L. Zhang, "A tale of three cdns:an active measurement study of hulu and its cdns," in Proceedings of IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS),, 2012, pp. 7–12.

[2] J. Koomey, "Growth in data center electricity use 2005 to 2010"" Analytics Press, aug. 2011

[3] L. Rao, X. Liu, L. Xie, and W. Liu, "Minimizing electricity cost: Optimization of distributed internet data centers in a multi-electricity market environment," in Proceedings of the 29th Annual Conference of the IEEE Communications Society (INFOCOM"10), San Diego, CA, USA, march 2010, pp. 1–9.

[4] Z. Liu, M. Lin, A. Wierman, S. Low, and L. Andrew, "Greening geographical load balancing," in Proceedings of the International Conference on Measurement and Modeling of Computer Systems (ACM Sigmetrics"11), San Jose, California, USA, june 2011, pp. 7– 11.

[5] E. Kayaaslan, B. B. Cambazoglu, R. Blanco, F. P. Junqueira, and C. Aykanat, "Energy-price-driven query processing in multi-center web search engines," in Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information (SIGIR"11), New York, NY, USA, 2011, pp. 983–992.

[6] Y. Yao, L. Huang, A. Sharma, L. Golubchik, and M. Neely, "Data centers power reduction: A two time scale approach for delay tolerant workloads," in Proceedings of the 31th Annual Conference of the IEEE Communications Society (INFOCOM"12), march 2012, pp. 1431 –1439.

[7] J. Luo, L. Rao, and X. Liu, "Temporal load balancing with service delay guarantees for data center energy cost optimization," IEEE Transactions on Parallel and Distributed Systems, vol. 99, no. PrePrints, 2013.

[8] D. K. Goldenberg, L. Qiuy, H. Xie, Y. R. Yang, and Y. Zhang, "Optimizing cost and performance for multihoming," in Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM"04), New York, NY, USA, 2004, pp. 79–92.

[9] H. H. Liu, Y. Wang, Y. R. Yang, H. Wang, and C. Tian, "Optimizing cost and performance for content multihoming," SIGCOMM Comput. Commun. Rev., vol. 42, no. 4, pp. 371–382, Aug. 2012.

[10] D. Meisner, B. T. Gold, and T. F. Wenisch, "Powernap: Eliminating server idle power," in Proceedings of the 14th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS"09), New York, NY, USA, 2009, pp. 205–216.

[11] L. Parolini, B. Sinopoli, B. Krogh, and Z. Wang, "A cyber-physical systems approach to data center modeling and control for energy efficiency," Proceedings of the IEEE, vol. 100, no. 1, pp. 254–268, 2012.

[12] M. Ilyas, S. Raza, C.-C. Chen, Z. Uzmi, and C.-N. Chuah, "Redbl: Energy solution for loading data centers," in Proceedings of the 31th Annual Conference of the IEEE Communications Society (INFOCOM"12), 2012, pp. 2866–2870.

[13] D. Xu and X. Liu, "Geographic trough filling for internet datacenters," in Proceedings of the 31th Annual Conference of the IEEE Communications Society (INFOCOM"12), 2012, pp. 2881–2885.

[14] Y. Yao, L. Huang, A. Sharma, L. Golubchik, and M. J. Neely, "Data centers power reduction: A two time scale approach for delay tolerant workloads," in Proceedings of the 31th Annual Conference of the IEEE Communications Society (INFOCOM"12), Orlando, Florida, USA, march 2012.

[15] H. Xu and B. Li, "A general and practical datacenter selection framework for cloud services," in Proceedings of the 5th International Conference on Cloud Computing (CLOUD"12), 2012, pp. 9–16.

[16] M. Adler, R. K. Sitaraman, and H. Venkataramani, "Algorithms for optimizing the bandwidth cost of

content delivery," Computer Networks, vol. 55, no. 18, pp. 4007–4020, Dec. 2011.

Author Profile

Rashmi Rajendra Kurkure 09,"Rajvatika", Ayodhya Nagar, MIDC, Jalgaon, Maharashtra, India, Pin- 425003

Madhuri Zawar is in GF"s foundation Godavari college of Engineering, Jalgaon