Computer Networks 131 (2018) 1–14

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Joint cache resource allocation and request routing for in-network caching services

Weibo Chu[a,*], Mostafa Dehghan[b], John C.S. Lui[c], Don Towsley[d], Zhi-Li Zhang[e]

[a] Northwestern Polytechnical University, Xi'an, China
[b] Google Inc., Cambridge, USA
[c] The Chinese University of Hong Kong, Hong Kong
[d] University of Massachusetts, Amherst, USA
[e] University of Minnesota, Minneapolis, USA

ARTICLE INFO

Article history:
Received 6 July 2017
Revised 23 October 2017
Accepted 28 November 2017
Available online 2 December 2017

Keywords:
cache resource allocation
cache partitioning
request routing
optimization
distributed algorithms

ABSTRACT

In-network caching is recognized as an effective solution to offload content servers and the network. A cache service provider (SP) always has incentives to better utilize its cache resources by taking into account diverse roles that content providers (CPs) play, e.g., their business models, traffic characteristics, preferences. In this paper, we study the cache resource allocation problem in a Multi-Cache Multi-CP environment. We propose a cache partitioning approach, where each cache can be partitioned into slices with each slice dedicated to a content provider. We propose a content-oblivious request routing algorithm, to be used by individual caches, that optimizes the routing strategy for each CP. We associate with each content provider a utility that is a function of its content delivery performance, and formulate an optimization problem with the objective to maximize the sum of utilities over all content providers. We establish the biconvexity of the problem, and develop decentralized (online) algorithms based on convexity of the subproblem. The proposed model is further extended to bandwidth-constrained and minimum-delay scenarios, for which we prove fundamental properties, and develop efficient algorithms. Finally, we present numerical results to show the efficacy of our mechanism and the convergence of our algorithms.

1. Introduction

In recent years, we have witnessed a dramatic increase in traffic over the Internet. It was reported that global IP traffic has grown 10 times from 2007 to 2015, and it will continue to increase threefold by 2020 [1]. Among the various types of traffic generated by different applications, traffic from wireless and mobile devices accounts for a significant portion, i.e., according to [2] global mobile and wireless data traffic in 2016 amounted to 47 exabytes per month, that is 49% of the total IP traffic.

Current Internet faces significant challenges in serving this "Big Data" traffic. The host-to-host communication paradigm makes it rather inefficient to deliver content to geographically distributed users due to repeated transmissions of content, which results in unnecessary bandwidth wastage and prolonged user-perceived delays. The connection-oriented communication model also provides little or poor support for user mobility – an important feature of future networks.

The overwhelming data traffic and limitations of the current Internet has led to a call for content-oriented networking solutions. Examples include CDNs (Content Delivery Networks) and ICNs [18] (Information-Centric Networks). Both advocate caching (either at network edge or network-wide) as part of network infrastructure, where content can be opportunistically cached so as to bring significant benefits such as bandwidth saving, short delays, server offloading. Due to its fundamental role in global content delivery, and the fact that cache storages are always scarce as compared to the amount of content transmitted over the Internet, how to efficiently utilize cache resources becomes a significant research topic. A furry of recent studies focus on this area, such as modeling and characterizing caching dynamics [24,33], design and performance evaluation of caching mechanisms [4,14], to name a few.

In this paper, we envision that besides maximizing cache performance (measured in hit rate or miss probability) as most previous work concentrated, we also study how cache resources in network should be utilized in a way that better supports general

* Corresponding author.
E-mail addresses: wbchu@nwpu.edu.cn (W. Chu), cslui@cse.cuhk.edu.hk (J.C.S. Lui), towsley@cs.umass.edu (D. Towsley), zhzhang@cs.umn.edu (Z.-L. Zhang).

https://doi.org/10.1016/j.comnet.2017.11.009
1389-1286/

management purposes (e.g., QoS, fairness). Particularly, since content providers (CPs) have business relations with cache providers, a cache provider always has incentives to utilize its cache resources fully by taking into account diverse roles that CPs play in the market, e.g., their heterogeneous traffic characteristics, business models, QoS requirements. Baring this in mind, in this paper we study the problem of allocating cache resources among multiple content providers. We consider the problem in a "Multi-CP Multi-Cache" environment, where there are multiple cache resources distributed at different network locations serving user requests from multiple content providers. This is exactly the same setting for a variety of networking applications, such as CDNs, wireless/femtocell networks, web-cache design, and most recently, ICNs. Since there are multiple paths between each content provider and its end-users through caches, it naturally leads to a problem of jointly optimizing cache resource allocation and request routing. However, achieving system optimum by this joint optimization with the objectives of, e.g., maximizing network utility, poses a significant challenge since the problem is inherently combinatorial and NP-hard [7,17,32], and thus some optimization algorithms are needed to solve these problems efficiently (with low complexity) and practically (in a decentralized manner).

In this work, we propose a joint cache partitioning and cache-level content-oblivious request routing scheme, where we allow a cache provider to partition its caches into slices with each slice dedicated to a content provider, and each content provider routes its requests to caches it connects so to maximize its own utility. Note that there are two advantages of the proposed scheme: 1) cache partitioning restricts content contention for cache space into partitions for each CP, and hence it decouples the interactions among them and also provides a natural means for the cache manager to tune the performance for each CP; 2) besides its simplicity due to content-obliviousness (less state), cache-level request routing provides a unified request pattern seen by caches, which leads to nice properties, i.e., the hit probability of each content is solely affected by allocated cache amount, and the hit rate of each CP is linear in traffic volume directed to caches. Overall, our scheme is easy-to-implement and is suitable for cache resource management.

To abstract business relations between content providers and a cache provider, we associate with each CP a utility that is a function of its content delivery performance. We formulate an optimization problem in which the objective is to maximize the weighted sum of utilities over all content providers through proper cache partitioning and request routing. We prove that the formulated problem has a biconvexity structure, and hence can be effectively solved by existing algorithms [15]. We further prove that, with our proposed routing scheme, the optimal solution to the formulated problem has a special request routing configuration, i.e., all requests of each CP are directed to one cache it connects. This property together with the convexity of the resource-allocation subproblem makes it possible to design decentralized (online) algorithms to achieve optimum.

To illustrate that our model actually provides a general framework for cache resource allocation, we extend it to bandwidth-constrained and delay optimization scenarios, where there are bandwidth limitations between caches and content providers, and where the goal is to optimize content delivery latency. We formulate optimization problems for the two scenarios, and establish the same biconvexity property. In addition, we discover interesting phenomena, i.e., under bandwidth limitation the optimal solution is the one such that each CP directs its requests to at most one cache at the volume less than the maximum volume, and it either does not direct or directs requests at the maximum volume to the other caches. Based on these fundamental properties, efficient algorithms can be devised.

In summary, we make the following contributions:

- We propose a joint cache partitioning and cache-level content-oblivious request routing scheme in the context of multiple content providers and multiple caches, and formulate a utility-based optimization framework for cache resource management.
- We prove fundamental properties of the formulated problem, obtain its optimal routing structure, and then develop decentralized algorithms.
- Using utility-based framework, we further consider bandwidth-constrained and delay optimization scenarios. We formulate optimization problems for the two extensions, show that they also have nice properties which lead to efficient algorithms design.
- We perform numerical studies to validate the efficacy of our mechanism, and demonstrate convergence of the proposed decentralized algorithms to optimal solution.

The remainder of this paper is organized as follows. We review related work in Section 2. Section 3 describes problem setting and basic model. In Section 4 we formulate the joint cache resource allocation and request routing problem, prove its fundamental properties by analyzing its problem structure. In Section 5 we develop decentralized (online) algorithm for implementing utility-maximizing cache allocation. Section 7 presents numerical results and Section 8 discusses future research directions. We conclude the paper in Section 9.

2. Related work

The issue of cache resource allocation and management has been extensively studied in the context of CPU and memory caches (i.e., see [20,27] and the references therein). Clearly, the characteristics of the cache workload and problem settings are quite different from the networking environment, so that the techniques and design choices developed therein cannot be readily applied to our problem.

In the context of web caching, Kelly et al. [19] proposed a biased replacement policy for web caches to implement differentiated quality-of-service (QoS) by prioritizing cache space to servers. Ko et al. [21] presented a scalable QoS architecture for a shared cache storage which guarantees hitrates to multiple competing classes. Lu et al. [23] implemented an architecture for supporting differentiated caching services and adopted a control-theoretical approach to manage cache resources. Feldman and Chuang [9] proposed a QoS caching scheme that achieves service differentiation through preferential storage allocation and objects transitions across priority queues. A general cache partitioning model that integrates both QoS classes, content priority and popularity is also presented in [10].

In recent years, a significant research effort has been dedicated to the cache resource management issue in information-centric networks. Rossi et al. [29] proposed to allocate content storages heterogeneously across the network by considering graph-related centrality metrics. Psaras et al. [26] proposed probabilistic caching scheme and their studies suggested to put more cache resources at the network edge. Similarly, Fayazbakhsh et al. [8] demonstrated through simulations that most of performance benefits can be achieved by edge caching. Wang et al. [30] studied the problem of optimal cache resource allocation to network nodes by formulating it as a content placement problem.

Cache resource allocation among content providers in network for management purposes (e.g., QoS, fairness) is a new research topic. Araldo et al. in [3] adopted content-oblivious cache partitioning approach to maximize the bandwidth savings provided by the ISP cache for handling content encryption. While they focused on single-cache allocation, the problem we study here is in a Multi-Cache (and possibly with multiple service providers) environment. Hoteit et al. in [16] proposed a game-theoretic cache allocation ap-