Computer Networks 56 (2012) 2080–2095

Contents lists available at SciVerse ScienceDirect

Computer Networks

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

A bandwidth and effective hit optimal cache scheme for wireless data access networks with client injected updates

Mursalin Akon, Mohammad Towhidul Islam, Xuemin (Sherman) Shen *, Ajit Singh

Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1

ARTICLE INFO

Article history:
Received 28 October 2010
Received in revised form 18 October 2011
Accepted 17 February 2012
Available online 2 March 2012

Keywords:
Wireless data access
Cache
Replacement policy
Access scheme
Data update
Proactive and reactive data access

ABSTRACT

In this paper, we propose an optimal cache replacement policy for data access applications in wireless networks where data updates are injected from all the clients. The goal of the policy is to increase effective hits in the client caches and in turn, make efficient use of the network bandwidth in wireless environment. To serve the applications with the most updated data, we also propose two enhanced cache access policies making copies of data objects strongly consistent. We analytically prove that a cache system, with a combination of our cache access and replacement policy, guarantees the optimal number of effective cache hits and optimal cost (in terms of network bandwidth) per data object access. Results from both analysis and extensive simulations demonstrate that the proposed policies outperform the popular Least Frequently Used (LFU) scheme in terms of both effective hits and bandwidth consumption. Our flexible system model makes the proposed policies equally applicable to applications for the existing 3G, as well as upcoming LTE, LTE Advanced and WiMAX wireless data access networks.

1. Introduction

In recent years, we have witnessed extraordinary improvements in computing electronics and wireless communications. These inventions are promising flexibility to our daily lives. Traditionally, cellular devices were used for voice communication. In contrast, modern mobile communication devices, such as smart phones, personal digital assistants (PDAs) and other hand-held computers are powerful general purpose computing devices with communication capabilities. These devices create the platform for computing and data access from any where and at any time by using existing high bandwidth 3G, under deployment LTE, and expected LTE Advanced and IEEE 802.16 m data access networks. Due to these emerging technologies, many necessary and entertaining mobile Internet applications such as Mobile IP telephony, mobile TV, video-on-demand (VOD), video conference, tele-medicine, mobile online banking, stock market tracking, instant messaging, on the road adaptive navigation, multi-player games have become a reality. Live Internet applications, such as online social networking (i.e., Facebook [1], Qzone [2], MySpace [3], Twitter [4]), document storage and sharing (i.e., Skydrive [5], Flickr [6], Picasa [7], Photobucket [8], Dropbox [9]), video sharing (i.e., Youtube [10], Youku [11]) have changed the way people communicate with each other, and store and share their multimedia contents and documents. The benefits and features of these Internet applications are made available at the expense of huge bandwidth consumptions, burdening the communication infrastructure. Mobile devices with faster processor, variety of sensors and powerful operating systems are becoming more affordable. As more users subscribe for wireless data access services, the problem of bandwidth is simply going to get worse. Modernizing wireless communication infrastructure towards higher capacity is extraordinary expensive. Thus, the service provides have no choice but to look for

* Corresponding author.
E-mail addresses: mmakon@bbcr.uwaterloo.ca (M. Akon), mtislam@bbcr.uwaterloo.ca (M.T. Islam), xshen@bbcr.uwaterloo.ca (X. (Sherman) Shen), asingh@uwaterloo.ca (A. Singh).

1389-1286/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2012.02.015

alternate solutions and user incentive mechanisms [12–14].

In a mobile information retrieval system, databases and files are hosted at a remote server. Conventionally these servers are directly connected to the wired networks to make the data access process faster and feasible. When a wireless user accesses data objects from the server, all communications have to pass through the channels of the wireless network. In spite of many improvements, wireless channel bandwidth is the scarcest resources, making data access in wireless networks very expensive. To reduce the data access cost, a client has to be very economic about the amount of data access. Additionally, when data is accessed a client must make the best utilization of the wireless channels. Many efforts have been put towards the best utilization of the available wireless channels. In such efforts, some applications behave adaptively depending on the state of the wireless channel, available bandwidth and other resources. For example, a mobile image retrieval system may retrieve images whose quality is adjusted according to the available bandwidth. A mobile device Internet browser retrieving compressed hypertext documents from the server and decompressing before displaying is another example of environment aware applications. However, developing such network aware applications is not trivial, particularly because the application logic as well as the development process become exceptionally complex [15]. In wireless data access applications, caching recently accessed objects is a very practical approach to reduce the amount of data access, and hence, cost of data access. Notice that cache oriented solutions do not contradict with the idea of developing network aware applications, rather, in many cases, cache can be deployed irrespective of network awareness of the applications.

Incorporating cache in a network system may increase the performance of the system in three ways – (1) the average access time or latency is shortened. As the local cache hosts a number of objects (depending on some criteria) many data objects are delivered from the local cache without retrieving the objects from the remote server. Therefore, caching is heavily used in hardware systems, such as processor cache and disk cache; (2) if caching is not deployed, at an access, the requested object would have to be fetched from the data server. With cache, many data objects are served locally, reducing the amount the data transferred over the networks; and (3) without cache, all data accesses have to be handled by the server. With cache, many of the accesses are handled by the client, cutting down the server load. The latter two benefits also make the data access system more scalable – allowing a server to handle more clients without adding any additional computing or network resources. Use of cache in wireless environment results in another very interesting and crucial benefit to its users. Data communications from wireless devices consume a significant amount of power. Utilization of a cache reduces the amount of communication transmissions. Consequently, data access cost is reduced, power is saved, and battery life is prolonged.

The goal of deployment cache may vary. In wireless data access applications, a cache mechanism needs to address two crucial aspects – cache access and cache replacement policies. A cache access policy determines how a cache is accessed and how the client–server system utilizes the cache. Maintaining consistency between copies scattered over clients and the server is also a task of the cache access policy. Notice that ensuring consistency in a distributed environment is complicated, particularly, if updates are allowed to be injected from any of the clients.

Despite the complexity of the problem, many applications demands availability of the most recently updated information/objects, because updates render all copies of older versions of the updated objects obsolete for further computation. This kind of consistency is called the *latest value* consistency [16], and a cache, satisfying the latest value consistency, is said to be *strongly consistent* [16,17]. In this case, a client has to retrieve the data item from the server. Several strong cache consistency algorithms for mobile/wireless data access have been proposed, such as, *Invalidation Report (IR)* schemes [18,19,16,20–29], *Poll-Each-Read* [17] and *Call-Back* [30,17,31,28].

When a cache is full and a new object is introduced, a decision has to be made – whether caching the new object is beneficial, and if it is, which existing object should be removed to make space. A replacement policy makes this important decision. Most researchers consider *Least Recently Used (LRU)* [18,19,16,20–25,17,26–29] or *Least Frequently Used (LFU)* [32] replacement policies for wireless data access. In general, an access policy can be deployed with any of a set of replacement policies and vice versa. However, performance of different combinations of access and replacement policies varies depending on system characteristics. Hence, a system developer has to be prudent in choosing the appropriate replacement and access policies.

In this paper, we provide a strongly consistent and update-aware cache mechanism for wireless clients scattered over a network spanning multiple wireless cells, where data updates may originate from any client. We make three major contributions – firstly, we propose two strongly consistent cache access policies – *Proactive Access Policy* (PAP), and *Reactive Access Policy* (RAP). Secondly, we introduce an Update-oriented Replacement Policy (URP). Our access policies are designed keeping the replacement policy in mind. The access policies collect different access and update related information to facilitate working capability of the cache replacement policy. In turn, the replacement policy aims towards higher effective hits. Thirdly, we analytically prove that our replacement policy ensures the optimal performance. As a result, this research provides the upper boundary for the worst case performance of any caching scheme and a foundation for average case performance comparison. The design goals of the proposed cache mechanism are – (1) to increase the effective hit ratio and (2) to reduce transmission cost (i.e., bandwidth consumption) by the applications. Simulations are performed to validate our proposals and claims.

The remainder of the paper is organized as follows. We present our system model, related works and performance metrics in Section 2. The PAP and RAP cache access, and URP cache replacement policies are introduced in Section 3. Quantitative analysis is provided in Section 4. Performance