ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Implementation of Provably Optimal Policies That Stabilize the Request Queues and Reduce Average Deficit to Zero at Small Cost and Ensures QOS

Mirza Abdul Azeem Baig
M.Tech Student,
Department of CSE,
Nawab Shah Alam Khan College of Engineering and Technology,
New Malakpet, Hyderabad.

Mr. Mohammed Khaleel Ahmed
Associate Professor,
Department of CSE,
Nawab Shah Alam Khan College of Engineering and Technology,
New Malakpet, Hyderabad.

Dr. Mohammed Waheed Uddin Hussain
HOD,
Department of CSE,
Nawab Shah Alam Khan College of Engineering and Technology,
New Malakpet, Hyderabad.

Abstract:

A content cache(a type of web cache) is an information technology for the temporary storage (caching) of web documents, such as HTML pages and images, to reduce bandwidth usage, server load, and perceived lag. A web cache system stores copies of documents passing through it; subsequent requests may be satisfied from the cache if certain conditions are met. A web cache system can refer either to an appliance, or to a computer program. In this paper we divide users depending on their requests for content. For some users delay may be ok, but for others delay is not acceptable. According to user requirements queues are formed. In the proposed system media vault is used inorder to store the content for catching. We employ algorithms for content distribution depending upon the users flexibility over delay.

Keywords:
Content, Wireless networks, Delay, Clusters, Media Vault.

Introduction:

Content Caching is an area of a computer's memory devoted to temporarily storing recently used information. The content, which includes HTML pages, images, files and Web objects, is stored on the local hard drive in order to make it faster for the user to access it, which helps improve the efficiency of the computer and its overall performance.Most caching occurs without the user knowing about it. For example, when a user returns to a Web page they have recently accessed, the browser can pull those files from the cache instead of the original server because it has stored the user's activity.

The storing of that information saves the user time by getting to it faster, and lessens the traffic on the network. Benefits of content caching:Content caching improves the performance of a web site by temporarily storing data that was recently accessed. While it's cached, requests for that data will be served by the load balancer instead of making another query to a web server behind it. The result is improved response times for those requests and less load on the web server.

• Users see faster load times for digital content, whether that means videos, images, compressed files, web pages or online games.
• Enterprises see higher customer satisfaction and engagement, avoiding the chance that distant users abandon the site for performance reasons.
• Additionally, enterprises see lower bandwidth costs since files are served from local caching servers, which typically have bulk data transfer rates.

File Types:

Content caching works well for files that don't change or that rarely change. Most images and static content are good candidates for content caching. You don't want to cache files that would change regularly or would be dynamically generated for different site visitors.

Related Work:

Several papers have addressed content caching and content replacement in wireless networks. In the following sections, we review the works that are most related to this paper of Caching in wireless networks.

File Types:

Content caching works well for files that don't change or that rarely change. Most images and static content are good candidates for content caching. You don't want to cache files that would change regularly or would be dynamically generated for different site visitors.

Related Work:

Several papers have addressed content caching and content replacement in wireless networks. In the following sections, we review the works that are most related to this paper of Caching in wireless networks.

Cooperative Caching:

In the paper [L. Yin and G. Cao, "Supporting cooperative caching in ad hoc networks," IEEE Trans. Mobile Comput., vol. 5, no. 1, pp. 77–89, Jan. 2006], distributed caching strategies for ad hoc networks are presented according to which nodes may cache highly popular content that passes by or record the data path and use it to redirect future requests. Among the schemes presented in [L. Yin and G. Cao, "Supporting cooperative caching in ad hoc networks," IEEE Trans. Mobile Comput., vol. 5, no. 1, pp. 77–89, Jan. 2006], the approach called HybridCache best matches the operation it as a benchmark for Hamlet in our comparative evaluation. In [N. Dimokas, D. Katsaros, and Y. Manolopoulos, "Cooperative caching in wireless multimedia sensor networks," ACM Mobile Netw. Appl., vol. 13, no. 3/4, pp. 337–356, Aug. 2008.], a cooperative caching technique is presented and shown to provide better performance than HybridCache.

However, the solution that was proposed is based on the formation of an over-lay network composed of "mediator" nodes, and it is only fitted to static connected networks with stable links among nodes.These assumptions, along with the significant communication overhead needed to elect "mediator" nodes, make this scheme unsuitable for the mobile environments that we address. The work in [Y. Du, S. K. S. Gupta, and G. Varsamopoulos, "Improving on-demand data access efficiency in MANETs with cooperative caching," Ad Hoc Netw.,vol. 7, no. 3, pp. 579–598, May 2009.] proposes a complete framework for information retrieval and caching in mobile ad hoc networks, and it is built on an underlying routing protocol and requires the manual setting of a network wide "cooperation zone" parameter.

Note that assuming the presence of a routing protocol can prevent the adoption of the scheme in [Y. Du, S. K. S. Gupta, and G. Varsamopoulos, "Improving on-demand data access efficiency in MANETs with cooperative caching," Ad Hoc Netw.,vol. 7, no. 3, pp. 579–598, May 2009.] in highly mobile networks, where maintaining network connectivity is either impossible or more communication expensive than the querying/caching process. Furthermore, the need of a manual calibration of the "cooperation zone" makes the scheme hard to configure, because different environments are considered. Conversely, Hamlet is self contained and is designed to self adapt to network environments with different mobility and connectivity features.

Content Diversity:

Similar to Hamlet, in [G. Cao, L. Yin, and C. R. Das, "Cooperative cachebased access in ad hoc networks," Computer, vol. 37, no. 2, pp. 32–39, Feb. 2004], mobile nodes cache data items other than their neighbors to improve data accessibility. In particular, the solution in [G. Cao, L. Yin, and C. R. Das, "Cooperative cachebased access in ad hoc networks," Computer, vol. 37, no. 2, pp. 32–39, Feb. 2004] aims at caching copies of the same content farther than a given number of hops. Such a scheme, however, requires the maintenance of a consistent state among nodes and is unsuitable for mobile network topologies. The concept of caching different content within a neighborhood is also exploited in [C.-Y. Chow, H. V. Leong, and A. T. S. Chan, "GroCoca: Group-based peer-to-peer cooperative caching in mobile environment," IEEE J. Sel. Areas Commun., vol. 25, no. 1, pp. 179–191, Jan. 2007.], where nodes with similar interests and mobility patterns are grouped together to improve the cache hit rate, and in [T. Hara, "Cooperative caching by mobile clients in push-based informa- tion systems," in Proc. CIKM, 2002, pp. 186–193.], where neighboring mobile nodes implement a cooperative cache replacement strategy. In both works, the caching management is based on instantaneous feedback from the neighboring nodes, which requires additional messages. The estimation of the content presence that we propose, instead, avoids such communication overhead.

Existing System :

The problem of caching, and content scheduling has earlier been studied for online Web caching and distributed storage systems.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

A commonly used metric is a competitive ratio of misses,assuming an adversarial model. Load balancing and placement with linear communication costs is examined. Here, the objective is to use distributed and centralized integer programming approaches to minimize the costs. However, this work does not take account for network capacity constraints, delay-sensitive traffic, or wireless aspects.

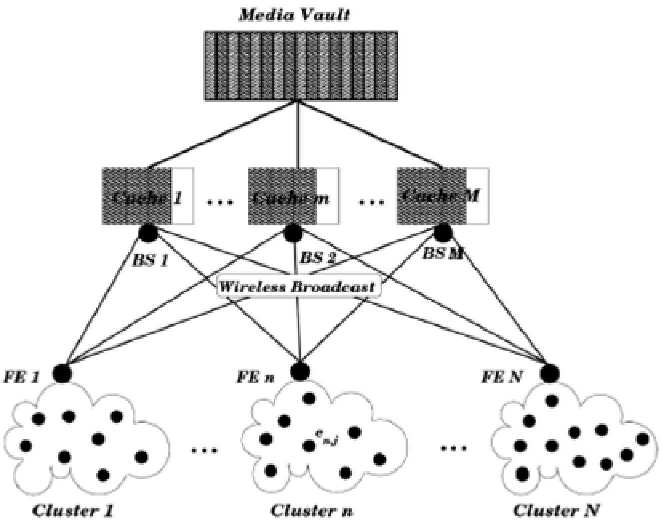
Disadvantages :

• Do not consider content distribution with its attendant question of content placement.
• Only considers elastic traffic and has no results on the value of prediction.

Proposed System :

In this paper, we are interested in solving the joint content placement and scheduling problem for both elastic and inelastic traffic in wireless networks. In doing so, we will also determine the value of predicting the demand for different types of content and what impact it has on the design of caching.

System architecture:



We use a request queue to implicitly determine the popularity of elastic content. Similarly, the deficit queue determines the necessary service for inelastic requests. Content may be refreshed periodically at caches. We study two different kinds of cost models, each of which is appropriate for a different content distribution scenario. The first is the case of file distribution (elastic) along with streaming of stored content (inelastic), where we model cost in terms of the frequency with which caches are refreshed. The second is the case of streaming of content that is generated in real-time, where content expires after a certain time, and the cost of placement of each packet in the cache is considered.

Client Module:

This is any entity that requests data from a data server. When a client submits a request, besides the data it requests, it may also include some content service requirements, arising from device limitations and data format limitations

Intermediate Server Module:

This is any entity that is allowed by a data server to provide content services in response to requests by clients. Intermediaries include caching proxies and transforming proxies. Totally we are using four proxy's. each proxy allotted for specific function. Based on user request the allotted proxy will process the request. if any one proxy busy with some function then the request delegated to the sub proxy that is assigned for the particular requested function.

Integrated Cache-Routing:

• Nearest-caching tables can be used in conjunction with any underlying routing protocol to reach the nearest cache node, as long as the distances to other nodes are maintained by the routing protocol.
• However, note that maintaining cache-routing tables instead of nearest-cache tables and routing tables doesn't offer any clear advantage in terms of number of messages transmissions.
•We could maintain the integrated cache-routing tables in the similar vein as routing tables are maintained in mobile ad hoc networks. Alternatively, we could have the servers periodically broadcast the latest cache lists. In our simulations, we adopted the latter strategy, since it precludes the need to broadcast Add Cache and Delete Cache messages to some extent.

Localized Caching Policy:

The caching policy of DCA is as follows. Each node computes benefit of data items based on its "local traffic" observed for a sufficiently long time.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The local traffic of a node i includes its own local data requests, nonlocal data requests to data items cached at i, and the traffic that the node i is forwarding to other nodes in the network.

•A node decides to cache the most beneficial (in terms of local benefit per unit size of data item) data items that can fit in its local memory. When the local cache memory of a node is full, the following cache replacement policy is used.
•In particular, a data item is newly cached only if its local benefit is higher than the benefit threshold, and a data item replaces a set of cached data items only if the difference in their local benefits is greater than the benefit threshold.

Data Server Module:

This is an entity that originally stores the data requested by a client. The data server module contains the Intermediary profile table. The table contains the detail about the proxy and specific function about particular proxy and their keys are all stored in that table. The server uses all these details when the client sends the request. And it contain function and corresponding privileges table also. This table contains the detail about what are the data present in the server and there privileges present in the table. The client request processed based on the privileges. It has two updates. Read and update.

Advantages :

•The elastic request queues have a finite mean, while inelastic deficit values are zero on average.
•The max-weight scheduling algorithm that we propose to use for joint content placement and scheduling. It satisfies the feasibility constraints.
•A joint content placement and scheduling algorithm that minimizes the average expected cost while stabilizing the deficit queues.

Conclusion:

Mobile video is the main reason for speedy growth in cellular data traffic. Conventional methods for network capacity increase are very costly and do not exploit the unique features of video, especially asynchronous content reuse.In this paper we examined and implemented a content caching mechanism using Media vaults. Users of the Wireless network are also categorized and serviced according to their needs and requirements.

This was utilizing the existing brandwidth, we will be able to satisfy multiple customers simultaneously. A significant area of future work is that of predicting user requirements. The efficiency of caching schemes depends not only on the quantity of content reuse, but also on our capability to recognize and predict request behavior across clusters of users.

References:

[1] Navid Abedini and Srinivas Shakkottai, Member, IEEE, Content Caching and Scheduling in Wireless Networks With Elastic and Inelastic Traffic, IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 22, NO. 3, JUNE 2014.
[2] N. Abedini and S. Shakkottai, “Content caching and scheduling in wireless broadcast networks with elastic and inelastic traffic,” in Proc. IEEE WiOpt, 2011, pp. 125–132.
[3] B.-J. Ko and D. Rubenstein, “Distributed selfstabilizing placement of replicated resources in emerging networks,” IEEE/ACM Trans. Netw., vol. 13, no. 3, pp. 476–487, Jun. 2005.
[4] G. Cao, L. Yin, and C. R. Das, “Cooperative cachebased access in ad hoc networks,” Computer, vol. 37, no. 2, pp. 32–39, Feb. 2004.
[5] C.-Y. Chow, H. V. Leong, and A. T. S. Chan, “Gro-Coca: Group-based peer-to-peer cooperative caching in mobile environment,” IEEE J. Sel.Areas Commun., vol. 25, no. 1, pp. 179–191, Jan.2007.
[6] T. Hara, “Cooperative caching by mobile clients in push-based informa- tion systems,” in Proc. CIKM, 2002, pp. 186–193.
[7] L. Yin and G. Cao, “Supporting cooperative caching in ad hoc networks,” IEEE Trans. Mobile Comput., vol. 5, no. 1, pp. 77–89, Jan. 2006.
[8] N. Dimokas, D. Katsaros, and Y. Manolopoulos, “Cooperative caching in wireless multimedia sensor networks,” ACM Mobile Netw. Appl., vol. 13, no. 3/4, pp. 337–356, Aug. 2008.
[9] Y. Du, S. K. S. Gupta, and G. Varsamopoulos, “Improving on-demand data access efficiency in MANETs with cooperative caching,” Ad Hoc Netw.,vol. 7, no. 3, pp. 579–598, May 2009
[10] W. Li, E. Chan, and D. Chen, “Energy-efficient cache replacement policies for cooperative caching in mobile ad hoc network,” in Proc. IEEE WCNC, Kowloon, Hong Kong, Mar. 2007, pp. 3347–3352.