RESEARCH ARTICLE OPEN ACCESS

Data Caching and Prefetching Using Collaborative Caching and Prefetching Algorithm in Wireless ADHOC Networks

[1] A. Bathsheba Parimala, [2] B. Jefferson, [3] K. Appasamy
Assistant professor, Department of BCA & M.SC(NT&IT) , St . Johns College , Palayamkottai.

Abstract:

Prefetching the information is a prominent strategy that enhances information availability in wired or remote systems. However, in wireless ad hoc networks, improvement in access latency and cache hit ratio may diminish because of the mobility and limited cache space of hosts. The proposed scheme uses a Collaborative Caching with Data Prefetching (CCDP) which is a method that prefetches the image based on association among data items. The scheme prefetches highly related data items and considers confidence of association rules. Whenever a node issues a request, the cache request processing module first logs this request into record and checks whether the desired data item is available in local cache of that node or in any of the node in the cluster. If it is a cache hit, the cache manager still needs to validate the consistency of the cached item with the copy at the original server. To validate the cached item, the cache manager checks the validation of data item from its TTL value. If the data item is verified as being upto date, it is returned to the node immediately. If it is a cache hit, but the value is obsolete, the cache manager sends an uplink requests to the server and waits for the data broadcast. When the requested data item appears, the cache manager returns it to the requester and retains a copy in the cache. The simulations are done using both MATLAB and Network Simulator 2 tool and the performance metrics like throughput, packet loss and delay will be compared with previous schemes.

Keywords **— Prefetching, Caching, ad-hoc networks**

I. INTRODUCTION

A wireless ad hoc network is a continuously auto-configuring, infrastructure-less network of mobile nodes connected wirelessly. Each client in a WANETs (Wireless Adhoc Networks) is free to move individually in any direction and will therefore change its links to other nodes frequently. Each must forward traffic unrelated to its own use and therefore be a router. Cache prefetching is a technique used by computer processors to boost execution performance by fetching instructions or data from their original storage in slower memory to a faster local memory before it is actually needed. Most modern computer processors have fast and local cache memory in which prefetched data is held till it is required. As mobile nodes in adhoc networks may have similar tasks and share common interest, cooperative caching, which allows sharing and coordination of cached data among multiple nodes can be used to reduce the bandwidth and power consumption. Since caching and prefetching are both well recognized for improving client perceived response time, the integration of both strategies may be exploited to improve the system performance. In mobile ad hoc networks cache misses are not isolated events and a cache miss may followed by a series of cache misses. Therefore data mining association rules may be used to find relationship among data items and hence perform the prefetching. In the prefetching, access to remote data is anticipated and the data is fetched before it is required.

In this paper a Collaborative Caching with Data Prefetching" (CCDP) that prefetches the image data based on association among image data items. The proposed scheme prefetches highly related image data items. To enhance the caching performance in GCCIP, the generated caching rules are used to prefetch the image data item(s). Simulation is performed using both MATLAB and Network simulator2 for evaluating the performance of this algorithm under several circumstances. Based on caching rules, prefetching is performed and confidence value along with other caching parameters is used during prefetching.

The remainder of this paper is organized as follows. In Section II we briefly review the related studies on cache replacement and prefetching in mobile ad-hoc networks and mobile environment. Section III gives description of proposed system. Section IV describes the results. Section V concludes the paper.

II. BACKGROUND WORKS

Dr M.Madheswaran and Mrs. K.Shanmugavadivu [1] have proposed a new group data caching scheme for improving data access efficiency in MANETs called Neighbor Group Data Caching. By collaborating local resources of mobile nodes, data availability and access efficiency is improved. Cooperative caching has two problems namely Cache resolution and cache management. However, it does not investigate the integration of broadcasting and cooperative caching. David S. L. Wei, Chao-Chin Chou, Jay Kuo and Kshirasagar Naik [2] has proposed anonymous communication protocol called MANET Anonymous Peer-to-peer Communication Protocol for P2P applications over wireless ad-hoc networks. Even under selective attacks, this anonymous communication protocol maintains high packet delivery ratio and also it is designed to be a middleware protocol sitting in between applications and network layer routing protocols. Nisar Hundewale, Sunsook Jung, , Alex Zelikovsky [3] has introduced approach to constrain route request broadcast which is based on node caching. Intuition behind node caching is that the nodes involved in recent data packet forwarding have more reliable information about its neighbors and have better locations (e.g., on the intersection of several data routs) than other nodes. Liangzhong Yin, Guohong Cao, and Chita R. Das [4] has proposed a cooperative cache-based data access framework lets mobile nodes cache the data or the path to the data to reduce query delays and improve data accessibility. Dan Hirsch and Sanjay Madria [5] have proposed novel scheme that seeks to distribute the storage, bandwidth and energy burden through a resource efficient adaptive caching scheme for mobile ad-hoc networks. Our performance results show that our scheme reduces both response time and bandwidth utilization by, 36%, through a reduction in hop count, as well as both a 79% increase in energy efficiency and a 53% reduction in storage utilization. Bhat [6] have proposed An Efficient Cache Management using Adaptive Buffer Mechanism in MANET. In a mobile environment, as a mobile node moves from one point of attachment to another during an ongoing application it is subjected to packet loss due to network and storage capacity [7]. Nayyar [8] have proposed a Cross-Layer System for Cluster Based Data Access in MANET'S. A cross-layer design approach is utilized to improve the performance of combined cooperative caching and prefetching schemes [9-12]. For future research there is a need to find out an efficient prefetching technique which further improves the data accessibility and reduce query delay to compliment the cooperative caching scheme [7]. Waleed et al., [13] have proposed A Survey of Web Caching and Prefetching. Web caching and prefetching are the most popular techniques that play a key role in improving the Web performance by keeping web objects that are likely to be visited in the near future closer to the client. Web caching can work independently or integrated with the web prefetching [12, 14].

III. COLLABORATIVE CACHING AND DATA PREFETCHING

Wireless Ad Hoc Networks (WANETs) consist of autonomous mobile nodes, these nodes cooperate with each other to exchange data by multiple-hop communication. Although each node has limited transmitting range, some nodes behave as routers and forward data (e.g. requests) from other nodes. Cache can be deployed either on each node or on some selected nodes to leverage cooperation. Figure 1 is a typical illustration of caching framework based on WANETs, in which caching nodes cooperate to cache contents and retrieve requested contents in a multiple-hop fashion. A collaborative caching with data prefetching framework for mobile nodes is proposed in which the buffer storage of each mobile node is allocated for prefetching the data from other nodes. Each cache node stores its most frequently accessed image data items in their buffer.

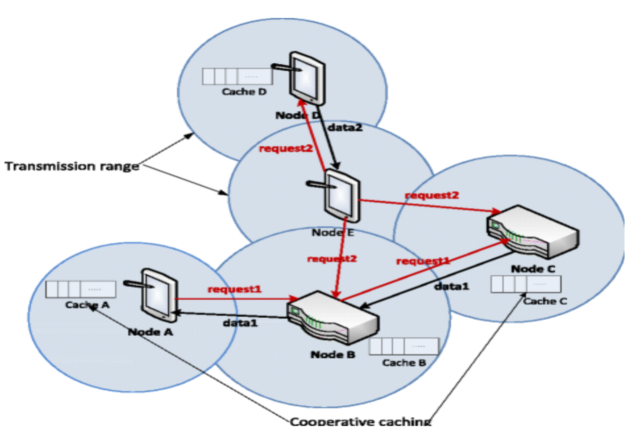


Figure 1 Caching Framework in WANET

There are some problems occurred during Caching and Prefetching in WANETs. They are as follows. Degradation of data access or latency in query processing is due to the following reasons with respect to caching and prefetching problems:

1. Cache Access: Slowdown in accessing the cache memory due to network congestion.

2. Memory constraint to store data: Memory is fixed, memory to be utilized efficiently to manage the data communication

3. Data overflow in prefetch memory: Data overflow from prefetch memory due to prefetch memory limit.

4. Cache Overflow: The contents not stored in the cache memory due to cache memory limit.

5. Cache Data Recovery: Data loss from the cache due to vulnerable attacks to be recovered.

6. Data consistency in cache and prefetch memory: Due to huge data traffic data consistency to be maintained between the cache and the prefetch memory

7. Data time stamp record: In prefetch memory data will be discarded if the data is not utilized for certain period of time.

The collaborative caching and data prefetching algorithm prefetches the data whenever possible to upgrade the performance of the networks. First the image is not processed in their format like .jpeg or .bmp, .png directly in network simulator. In network simulator tool the images are processed like hexadecimal value only. For processing image prefetching in the simulation, the image must be converted into hexadecimal value using MATLAB R2012a. The hexadecimal value will be used in network simulator tool to perform image prefetching process. The flowchart of the collaborative caching with image prefetching is shown in Figure 2.

Whenever a client issues a request for data to control server, the control server processing the request module and search the image data into its record and checks whether the desired image data item is available in local cache of mobile client or in any of the mobile client in the cluster. If the searched image data is present means, the cache manager still needs to validate the consistency of the cached image with the copy at the original server. To validate the cached image data, the cache manager checks the validation of data item from its time to live value. If the image data is verified as being upto date, it is returned to the mobile client immediately. If searched image data is present but the value is obsolete, the cache manager sends an uplink request to the server and waits for the image data broadcast. When the requested image data appears, the cache manager returns it to the requester and retains a copy in the cache. In the case that a searched image data miss occurs, the client cache manager checks the caching rule depository to derive the prefetching rules corresponding to the requested item. If this request triggers some prefetching rules, the ids of the item implied by these prefetching rules will also be piggybacked to the server along with id of missed cache item

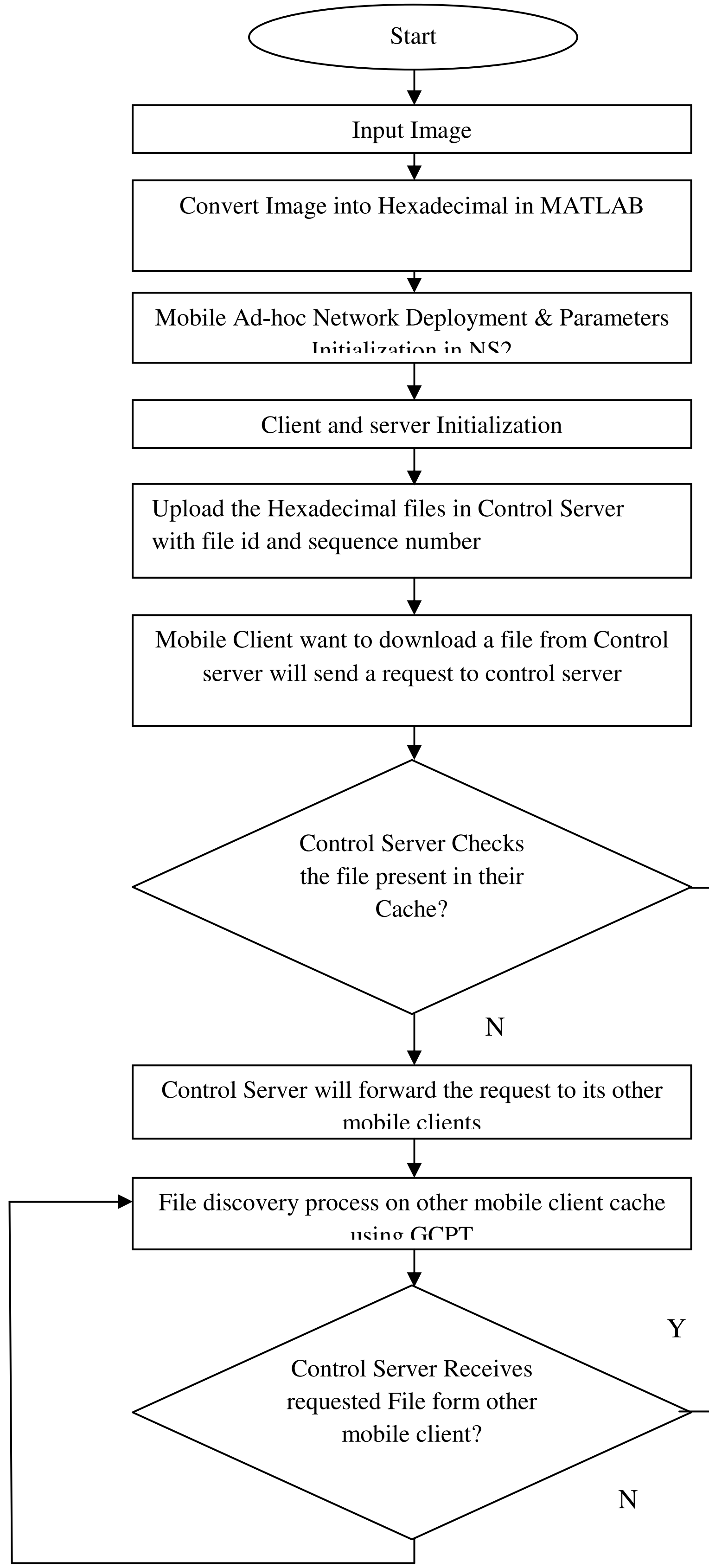


Figure 2 Flowchart of image prefetching

Algorithm:
Collaborative Caching with Data Prefetching

Step 1: Cache processing the request from mobile node. Cache will check the requested data is available in local or within cluster.

Step 2: If the requested data is available then check the confidence (originality of the data such as validity) and send it to requested mobile node

Else if the requested data is available but the validity is not satisfying the threshold level means cache manager sends the uplink request to server. After getting the data from server it will return the data to requested mobile node and keep that copy of data

Else

Step 3: the data is missing in cache manager checks the caching rule to derive prefetching rules using association rules for requested item. Assume that the client has a cache miss to item i_n. The proposed scheme finds out the rules whose antecedent is i_n. Consequent of each prefetch rule along with the confidence value in a linked list of such tuple is stored. The tuples of this list are then sorted in descending order of their confidence values. If cache miss happens, the prefetch set for the cached miss item is generated.

Step4: Instead of only requesting the cache miss item, the client also requests the items indicated in the prefetch set. When a cooperating node or server receives the request, it transfers the requested items over the wireless channel. The client downloads the items and stores them in its cache. By prefetching the items, the client can save future requests and reduce the query latency.

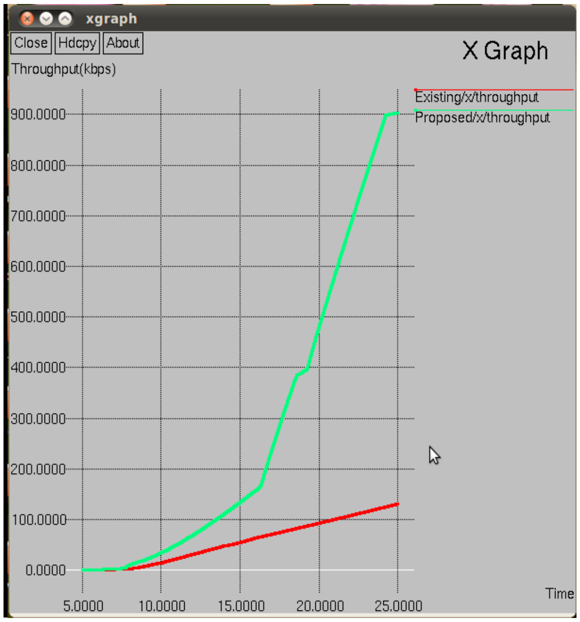
IV. RESULTS

The simulations are done using built-in random generator in Network Simulator 2 (version NS-2.34). The protocol evaluations are based on the simulation of 50 wireless mobile nodes forming an ad hoc network, moving about over a 5m/s. In our experiment, we have set the communication range of mobile node to 300m. The media access control layer we used in simulation is 802.11 MAC protocol. We generated all the movement scenarios using setdest command in NS2. First the images are converted into hexadecimal values using MATLAB, because in NS2 the images are processed only in hexadecimal value format. After the conversion of image to hexadecimal value that will be used to prefetching the data to mobile node. First if a mobile node requests image data to control server. After receiving request the control server first check its cache and if the requested image data is not present in the cache means it will forward the request to nearby mobile nodes. After getting the image data it will transfer the image data to the requested mobile node and keeps the copy of that image data.

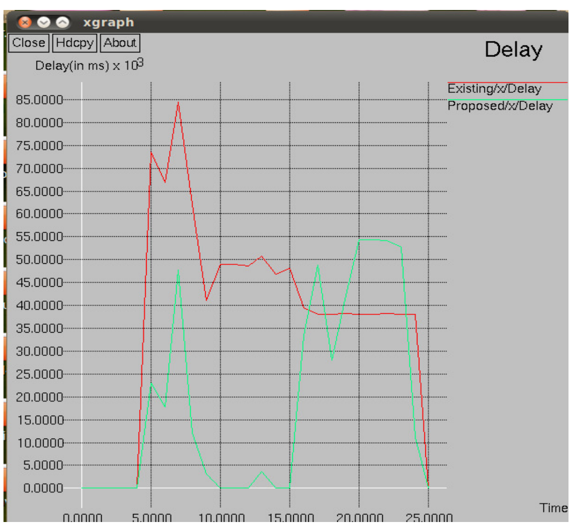
Table 1 : Simulation Parameters

Parameter	Value
Channel Type	Wireless
Routing Protocol	AODV
Queue Length	50 Packets
Number of Nodes in Topography	50,75.100
Node Placement	Random
Simulation End Time	50 sec
MAC Protocol	IEEE 802.11
Packet Size	512 bytes
Traffic Type	CBR
Path Loss Model	Free Space
Energy	500J
Transmission Power	0.75
Receiving Power	0.5
Mobility	5m/s
Frequency	912mhz

Communication Range	300m



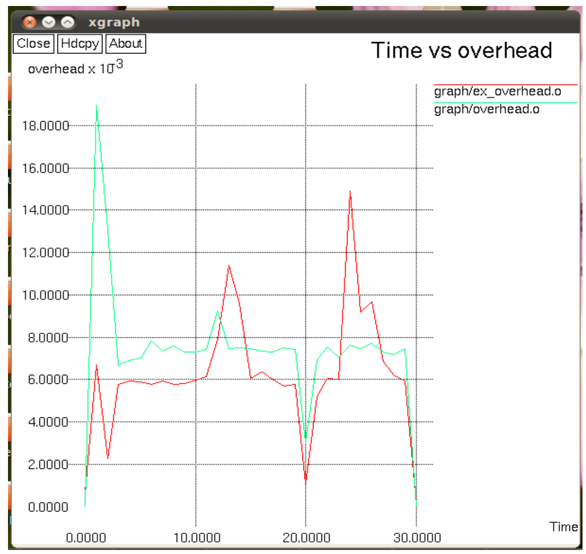
(a)



(b)

Figure 3 (a) Throughput (b) Delay

Figure 3(a) and (b) shows that the screenshot of throughput and delay. Throughput is defined as the amount of the bits received at receiver side after the data transmission. It will be measured in bits per second. The throughput of the proposed prefetching increases gradually and it will decrease because of the searching of image data. From this the throughput will increases than existing system. Delay is the amount of time taken for image data transfer between the client and server node. The proposed image prefetching technique requires less time to transfer the data than previous cooperative caching scheme.



(c)

Figure 3 (c) Overhead

Figure 3 (c) shows that snapshot of overhead. Overhead means excess consumption of resources that are required to perform a specific task. The proposed image prefetching has less overhead than existing.

V. CONCLUSION

In WANETs Caching and Prefetching technique can be used to upgrade the system performance in wireless ad hoc networks. However caching and image prefetching also consumes a large amount of system resources such as computation power and energy. Thus it is very important to only prefetch the right data. In this paper, collaborative caching and data prefetching algorithm is proposed which upgrades the performance improvement due to caching in wireless ad hoc networks. The proposed techniques will lead to further research work in several areas related to cache management and security against vulnerable attacks in mobile computing environments.

VI. REFERENCES

[1] Mrs. K.Shanmugavadivu and Dr M.Madheswaran, "Caching Technique for Improving Data Retrieval Performance in Mobile Ad Hoc Networks", International Journal of Computer Science and Information Technologies, Vol. 1 (4), 2010, 249-255

[2] Chao-Chin Chou, David S. L. Wei, Jay Kuo and Kshirasagar Naik "An Efficient Anonymous Communication Protocol for Peer-to-Peer Applications over Mobile Ad-hoc Networks", IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 25, NO. 1, JANUARY 2007

[3] Sunsook Jung, Nisar Hundewale, Alex Zelikovsky, "Node Caching Enhancement of Reactive Ad Hoc Routing Protocols", 2009

[4] Guohong Cao, Liangzhong Yin and Chita R. Das, "Cooperative Cache-Based Data Access in Ad Hoc Networks", Published by the IEEE Computer Society, 2004

[5] Dan Hirsch and Sanjay Madria "A Resource-Efficient Adaptive Caching Scheme for Mobile Ad-Hoc Networks", IEEE International Symposium on Reliable Distributed Systems, 2009

6. Kassem F and Hassan A (2013) Distributed Cache Invalidation Method For Maintaining Cache Consistency In Wireless Mobile Networks. IEEE Transactions on Mobile Computing 12: 1536-1233.

7. Nayyar A (2012) Cross-Layer System for Cluster Based Data Access in MANET'S. International Journal of Computer Science & Informatics 2: 15-20.

8. Denko MK and Tian J (2008) Cross-layer design for cooperative caching in mobile ad hoc networks. Consumer Communications and Networking Conf 375-380.

9. Artail H, Safa H, Mershad K, Abou-Atme Z, Sulieman N (2008) COACS: A cooperative and adaptive caching system for MANETs IEEE Trans. Mobile Comput 7: 961-977.

10. Tian J and Denko MK (2007) Exploiting clustering and cross layer design approaches for data caching in MANETs. Wireless and Mobile Computing, Networking and Communications 52.

11. Marquez J, Domenech J, Gil JA, Pont A (2008) A web caching and prefetching simulator. Software, Telecommunications and Computer Networks 346- 350.

12. Waleed A, Shamsuddin SM, Ismail AS (2011) A Survey of Web Caching and Prefetching. International Journal of Soft Computing and Its Applications 3: 18- 44.

13. Angrisani L, Bertocco M, Fortin D, Sona A (2008) Experimental study of coexistence issues between IEEE 802.11b and IEEE 802.15.4 wireless networks. IEEE Trans Instrum Meas 57: 1514-1523.

14. Chungsoo L and Gregory TB (2008) Exploiting producer patterns and L2 cache for timely dependence-based prefetching. IEEE 1063-6404.