Middle-East Journal of Scientific Research 23 (Sensing, Signal Processing and Security): 381-386, 2015
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2015.23.ssps.208

Caching Techniques and Content Management in Mobile Adhoc Network

[1]Tharani Vimal and [2]Sahaya Arul Mary

[1]Department of CSE, Manonmaniam Sundaranar University, Tirunelveli, 627012, Tamilnadu, India
[2]Department of CSE, Jayaram College of Engineering and Technology, Trichy-621014, Tamilnadu, India

Abstract: Caching techniques are used in Mobile Adhoc Network's to improve the data access efficiency and also reduce the bandwidth and delay. The management of this cache content is required to efficiently utilize the cache memory so that it improves the cache hit ratio and accordingly the data access efficiency. A number of caching techniques and cache management methods have been proposed. This paper consolidates the various caching techniques and cache management issues in MANETs like cache consistency, cache invalidation and cache replication. The data accessibility efficiency can be greatly enhanced on implementation of these techniques.

Key words: Cache · Cache consistency · Cache invalidation and cache replication

INTRODUCTION

Wireless networks have become increasingly popular these days. There are two types of mobile wireless networks. They are infrastructure based networks with fixed and wired gateways and infrastructure less mobile network. The first type is also known as "one-hop" wireless network that communicates through the access point to the other networks like wireless Lan. The second type is commonly known as the MANET and their network topology keeps changing continuously due to mobility of nodes.

They are self-organizing, self-configuring and "multi-hop" network which does not require any fixed infrastructure. Each node in MANET acts as a router keeping route information to reach other nodes. The nodes communicate among themselves using radio signals.

MANET's were initially used for military purpose which was later extended to industrial and commercial purpose. They were also used for disaster recovery, battlefield communications and rescue operations where it is difficult to set up a wired network. The following are some of the reasons that leads to performance degradation in MANETs.

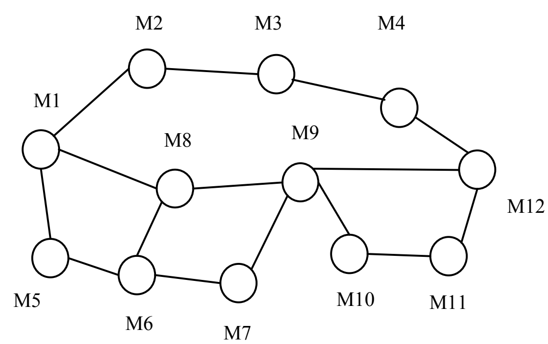

Fig. 1: Mobile Adhoc Network

- The nodes in a MANET generally access the data item from the data provider or the server. This results in heavy load on the server, as a result of which there is a delay in response time from the server.
- Frequent mobility of the nodes is another issue which leads to network partition and change of network topology which also degrades the system performance as the data is not available when required.
- The multi-hop communication in MANET degrades the performance since the data request has to go through multiple hops.

Corresponding Author: Tharani Vimal, Department of CSE,
Manonmaniam Sundaranar University, Tirunelveli, 627012, Tamilnadu, India.

So to improve the data access and availability of data, the technique of caching the data item in the nodes is used. Cache management in MANET is also a challenging issue due to frequent data updates, mobility, limited client resource and insufficient bandwidth.

Caching: Caching is a performance enhancing technique in MANETS used for storing frequently used information in the local memory so that when the same data is required next time, it can be fetched from the cache instead of fetching it from the data provider or server. This reduces the congestion in network and increases the performance of the system by reducing the communication cost, bandwidth requirements and latency in information retrieval from server. Caching is particularly suitable for mobile environments that are weakly connected or disconnected frequently from the network due to mobility. The data in the cache is not available if its lifetime expires or if the memory is released or if the caching does not happen due to some other reason. The items in cache are accessed using an indexing mechanism.

Some of the challenges to be resolved in a MANET for data caching are multihop routing, frequent disconnection of mobile hosts, dynamically changing topologies, varying link and node capabilities.

Some of the caching techniques currently used in MANET's are neighbor caching, group caching, cooperative caching and so on.

The cache data management in MANET's are broadly categorized as cache replacement, cache consistency and cache invalidation techniques.

In this paper, we have discussed on various types of caching techniques and issues of cache data management in MANETs.

Caching Techniques: In MANET's it is very important to prevent the deterioration of data accessibility at the point of network partition. So it is beneficial to cache frequently accessed data to reduce the average query latency and also save wireless bandwidth. There are different types of caching in MANET which are discussed as follows.

The concept of n*eighbor caching* (NC) in [1] utilizes the cache space of inactive neighbors for caching tasks. When a node wants to save the data it has fetched, in its caching space, due to cache limitations they are sometimes forced to remove the least used data. Such data's are saved in the cache space of idle neighboring nodes. In future if the node needs the data again, it requests the data not from the far remote source node but from the near neighbor that keeps the copy of data. The NC scheme utilizes the available cache space of neighbor to improve the caching performance.

The zone caching (ZC) [20] scheme allows a group of one hop neighbors to form a zone. The cost of communication within the zone is considered to be low in terms of energy consumption and message exchange. When the required data item is not available in the nodes within its zone then the request is forwarded to others outside the zone.

In [2][3][4] g*roup caching*(GC) maintains localized caching status of 1-hop neighbors for efficient caching. Each MH and its 1-hop neighbors form a group by using the "Hello" message mechanism. In order to utilize the cache space of each MH in a group, the MHs periodically send their caching status to the group. Thus, when caching placement and replacement need to be performed, the MH selects the appropriate group member to execute the caching task in the group, this reduces redundancy of cached data objects.

The c*ooperative caching* method in [5] proposes three methods of caching. They are cache data, cache path and hybrid cache. In *cache data*, the intermediate nodes cache the data to serve future requests instead of fetching data from the data center. In *cache path*, the nodes cache the data path and use it to redirect future requests. As the nodes are mobile cache path may not be reliable. Thus the data path is cached only if the caching node is very close or it results in stale path. In hybrid caching, a node caches the data or path based on the data size and time-to-live (TTL). If data size or TTL is small, cache data is optimal as data item occupies less space or the data will become invalid soon. If TTL or data is large, cache path is used. But the cache path can be invalid as the nodes are mobile. Hence the *hybrid cache* uses the advantages of both cache data and cache path.

A *cooperative and adaptive data caching scheme* (COACS) is proposed in [6].Here the nodes act as caching node(CN), request node(RN), query directory(QD). Queries are indexed and they improve the overall hit ratio in fetching the required information. This approach also reduces network traffic, minimizes the query delay and occupies less space compared to original data storage. The problem with this approach is the consistency maintenance as the data in cache nodes may be stale and may not be up to date as the consistency of data is not checked here. The updating process of node's mobility is based on the internal process of the routing protocol.

A dynamic caching technique [7] is proposed to cope up with the dynamic mobility pattern of MANET. The repeated data item and data path in MANET for a particular period of time are cached. Dynamic caching increases the data reusability rate.

The mobility of MHs has greater impact on data availability in MANET [8]. The data becomes unavailable when a particular node moves apart from a network. Two techniques are proposed to ensure data availability in MANET even in the greater mobility namely data replication and data diffusion. Both these protocols define global and stability metrics that ensure data availability. The proposed solution is tested on various mobility models such as Random Walk, Random Way Point, Mobility Management, Reverse Path Graph Model etc. It concludes that high node quantity has negative impact on data cache capacity. Moreover, high mobility affects the data diffusion.

Caching is carried out in higher level i.e. database level [9]. A large amount of data is stored in data center or database based on the query and its response. The nodes in database caching can take up either of these roles: query directory (QD), caching nodes (CN) and service manager (SM). It overcomes the database related caching issues. It mainly focuses on improving the query response time.

Cache Management: The cache content management mechanism addresses on how to manage the individual cache to improve the overall efficiency of the system. It is broadly categorized into cache consistency maintenance, cache invalidation and cache replacement strategy.

Cache Consistency Maintenance: The cache consistency mechanism ensures the validity of the copies of the data item cached. The cache consistency maintenance is a challenging issue due to frequent disconnections and mobility of the nodes. Several cache consistency maintenance schemes are proposed. The objective of these schemes is to increase the availability of correct up to date information and minimize the overhead in maintaining the consistency.

There are several strategies in maintaining mobile cache consistency.

In [10][11] cache consistency is based on consistency level. There are three levels of consistency suggested in this method. They are strong, delta and weak consistency levels. In strong consistency, the cache node is always up to date with source node. In delta consistency, the cached data is not out of date by more than a specific time with source data. In weak consistency, the data item in the node is an old copy of the original data item at the server.

In [12] cache consistency is based on consistency control. The cached copy should be updated whenever the original copy in the server is changed. Based on the update level two models are suggested, weak and strong consistency model.

The weak consistency model may return a stale data to the user. Two mechanisms of this type are TTL (Time-To-Live) and client-polling. In the TTL method a cached copy of data is considered up-to-date if the TTL has not expired. There are chances that the cached data may be stale though the TTL is alive. In the client-polling method, the client periodically checks with the server, the validity of the data cached. But here also there are chances that the cached data may be stale. Weak consistency is thus not a satisfactory method in maintaining consistency.

The strong consistency model will always return the up-to-date information after a write to the server. There are two mechanism involved in this type. They are Broadcast Invalidation Report, Invalidation message and client polling. In the Broadcast Invalidation message, the server will broadcast an invalidation report if the cached copy is changed. In the Invalidation message, the server will always keep a list of all clients where a particular data is cached. Whenever a data changes it will send an invalidation message to all the clients where that data was cached. In the Polling method, every time a cached data is requested, the cache validates the cached copy with the server before sending it to the user.

In [13] the consistency management is taken care of using four different consistency primitives based on the application requirement like GC,LC,TC,PC. The consistency level required depends on the who is the user of the application like an individual, group or all.

Global Consistency provides a strong consistency of cached data item for every read operation of data item and it is very difficult to achieve in MANET's.

Local Consistency helps to maintain strict consistency only among a group and its members. Strict consistency for the entire network is very costly and not necessary for such applications.

Time Based Consistency is used for systems where the value of data items keeps changing continuously and slowly. They are used to form temporary subgroups to perform a subtask.

Peer Based Consistency maintains consistency within the peers in a network.

In Smart Server Update Mechanism for Maintaining Cache Consistency in Mobile Environments, SSUM[14] the consistency maintenance problem of COACS is taken care. The server is always aware of the nodes that cache a particular data. Whenever there is a change of data in server it sends update message to all cache nodes that cache that data.

Cache Invalidation: It is important to maintain the cache consistency and it is achieved using invalidation techniques.

In [15] the author proposes invalidation by absolute validity interval (AVI) method to invalidate the cache. For a data item, AVI is based on the real time characteristics such as update interval. User may validate the data items on comparing its own AVI and recent update time. A data item is said to be invalidated when it has greater recent update time.

Server based cache consistency scheme (SSUM) is proposed in [14]. Based on the popularity and data update rate server performs all the processes such as data caching, updating etc. In this system, a node can take up the role of either QD or CN. These nodes have the control over the update rate of each node. Query response time and bandwidth utilization are considered to be primary parameter metrics. This mechanism is deployed on top of the COACS model. SSUM analyses on cache update rate and its impact on the server performance.

A strong cache consistency in World Wide Web can be achieved using adaptive TTL, polling every time and invalidation techniques. [16] Compares all these techniques in the aspect of analysis, implementation and trace reply. These are weak cache consistency techniques and each of them has their own drawbacks. Invalidation and adaptive TTL techniques introduce a considerable amount of network traffic and increases server load. Polling every time technique increases query response time. Therefore, it suggests a technique to achieve strong cache consistency accompanied with the invalidation based protocol.

Cache Replacement: It checks if the cache space is sufficient enough to hold the data item to be cached. If not it helps to decide on which cached data item is to be removed to give place for the new data item. Many techniques are proposed to address this issue.

The traditional policies for cache replacement are Least Recently Used (LRU), Least Frequently Used(LFU), First In First Out (FIFO), Least Recently Used with minimum size(LRUMIN), Object Size based replacement and TTL based.

Then depending on the decision of how caching replacement is made, it can be divided into non-cooperative and cooperative cache replacement. In the non-cooperative caching, an individual node takes its own decision regarding caching. Only the cooperative caching schemes supports cooperative cache replacement that allows sharing of cached information among nodes. Therefore, several nodes caching the same data item can be avoided.

Cooperative Cache Replacement: One of the popular cache replacement techniques is update based cache replacement strategy [17]. This can be further divided into least access to update (LA2U) and least access to update difference (LAUD). LA2U and LAUD are based on update frequency and access frequency respectively. These techniques concludes that update based cache replacement technique improve the cache performance significantly. This technique is preferred at the situation when the network experience heavy updates.

To decrease the cache miss ratio, a special cache replacement strategy is designed called time and distance sensitive (TDS) [18]. As the name indicates, it is a function of both time and distance. Distance is a measure of number of hops. Time is also considered as the distance becomes obsolete. Based on both these parameters, it is classified into three various strategies such as TDS_D, TDS_T and TDS_N. The TDS_D and TDS_T perform cache replacement based on distance and time respectively. TDS_N considers the product of both these parameters.

Non-Cooperative Cache Replacement: Most of the existing caching techniques exploit least recently used (LRU) cache replacement policies. The main idea behind LRU is to evict the data items that are kept unused for a long time. It is a simple and commonly used cache replacement algorithm. This kind of cache stores only the most recently used data items. When a new data item enters the cache, the cache places it in the front of the list and evicts the data item at the back of the list.

Since, MANET is a resource constrained network, a resource efficient caching method is essential. [19] Proposes greedy dual least utility (GDLU) algorithm to assure data availability dynamically. Utility based approach is based on certain MANET characteristics such as connection disconnection, hand off, data renewal, query pattern etc. It also includes passive pre fetching and dynamic cache replacement strategy. Cache replacement is based on the data item that has least utility value (LUV). LUV is a function of access probability, data item size, TTL value, distance between the client and data center.

CONCLUSION

In this paper we have discussed the various caching techniques for mobile adhoc networks and the three main issues in cache content management like cache consistency, cache invalidation and cache replacement. The future work is directed towards the efficient access of data items from the nodes in a mobile adhoc network after having considered all these issues.

REFERENCES

1. Joonho Cho, Seungtaek Oh, Jaemyoung Kim, Hyeong Ho Lee and Joonwon Lee, 2003. "Neighbor Caching in Multi-Hop Wireless Ad Hoc Networks", IEEE Communications Letters, 7(11), November 2003.
2. Yi-Wei Ting and Yeim-Kuan Chang, 2007. "A Novel Cooperative Caching Scheme for Wireless Ad Hoc Networks: Group Caching ", International Conference on Networking, Architecture and Storage (NAS 2007), 0-7695-2927-5/07 $25.00 © 2007 IEEE.
3. Chi-Yin Chow, Hong Va Leong and Alvin T.S. Chan, 2007. "GroCoca: Group-based Peer-to-Peer Cooperative Caching in Mobile Environment" IEEE Journal On Selected Areas In Communications, 25(1), January 2007.
4. Chi-Yin Chow, Hong Va Leong and Alvin T.S. Chan," 2004. Group-based Cooperative Cache Management for Mobile Clients in a Mobile Environment", Proceedings of the 2004 International Conference on Parallel Processing (ICPP'04).
5. Liangzhong Yin and Guohong Cao, 2006. "Supporting Cooperative Caching in Ad Hoc Networks", IEEE Transactions On Mobile Computing, 5(1), January 2006.
6. Hassan Artail, Haidar Safa, Khaleel Mershad and Zahy Abou-Atme, 2008. "COACS: A Cooperative and Adaptive Caching System for MANET" IEEE Transactions on mobile computing, 7(8).
7. Ying-Hong Wan, Jenhui Chen, Chih-Feng Chao and Tai-Hong Yueh, 2005. "A Dynamic Caching Mechanism for Mobile Ad Hoc Networks" 11th International Conference on Parallel and Distributed Systems (ICPADS).
8. Takahiro Hara, 2010. "Quantifying Impact of Mobility on Data Availability in Mobile Ad Hoc Networks" IEEE Transactions on Mobile Computing, 9(2).
9. Artail, H., H. Safa and S. Pierre, 2005. "Database Caching in MANETs Based on Separation of Queries and Responses" Proceedings of IEEE International Conference on Wireless and Mobile Computing, Networking and Communication, pp: 237-244.
10. Jiannong Cao, Yang Zhang, Guohong Cao and Li Xie, 2007. " Data Consistency for Cooperative Caching in Mobile Environments ", IEEE Computer Society 2007.
11. Doha Elsharief, Hamidah Ibrahim, Ali Mamat and Mohamed Othman, " A Framework for an application based mobile cache consistency method", Journal of Theoretical and Applied Information Technology.
12. Pei Cao and Chengjie Liu, 1998. "Maintaining Strong Cache Consistency in the World Wide Web", IEEE transactions on computers, April 1998.
13. Takahiro Hara and Senior Member, 2009. IEEE and Sanjay Kumar Madria, Senior member, IEEE, "Consistency Management Strategies for Data Replication in Mobile Ad Hoc Networks", IEEE Transactions on Mobile Computing, 8(7), JULY 2009.
14. Khaleel Mershad and Hassan Artail, June 2010. "SSUM: Smart Server Update Mechanism for Maintaining Cache Consistency in Mobile Environments".
15. Joe Chun-Hung Yuen, Edward Chan, Kam Yiu Lam and H.W. Leung, 2000. "An Adaptive AVI-based Cache Invalidation Scheme for Mobile Computing Systems" IEEE 11th international workshop on database and expert systems applications, pp: 155- 159.
16. Pei cao and chengjie liu, 1998. "Maintaining Strong Cache Consistency in the World Wide Web" IEEE transactions on computers, 47(4).
17. Hui Chen, Yang Xiao and Xuemin Shen, 2009. "Update-based Cache Replacement Policies in Wireless Data Access" IEEE 2nd International Conference on Broadband Networks, pp: 857-864.

18. Sunho Lim, Wang Chien Lee, Guohong Cao and Chita R. Das, 2003. “A Novel Caching Scheme for Internet based Mobile Ad Hoc Networks” ICCCN 12th international conference on computer communications and networks, pp: 38-43.
19. Huaping Shen, Mohan Kumar, Sajal K. Das and Zhijun Wang, 2004. “Energy-Efficient Caching and Prefetching with Data Consistency in Mobile Distributed Systems” Proceedings of the 18th International Parallel and Distributed Processing Symposium.
20. Chand, N. Joshi, R.C. and M. Misra, 2006. "Efficient Cooperative Caching in Ad Hoc Networks Communication System Software and Middleware," 2006. Comsware 2006. First International Conference on 08-12 Jan 2006 Page(s): 1-8.