ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Distributed Cache Model with Stability and Storage Management for MANET

S. Padmapriya[1], Mr. S. Sivaraj, ME[2]

II-M.E (CSE), Dept. of CSE, SSM College of Engineering, Komarapalayam, Tamilnadu, India[1]

Assistant Professor, Dept. of CSE, SSM College of Engineering, Komarapalayam, Tamilnadu, India[2]

Abstract: Pervasive Internet Access is supported under the Wireless ad hoc network. There is no support of any fixed infrastructure in a wireless ad hoc network environment. Wireless adhoc networks are constructed using smartphone, laptop, PALM-top, and sensor devices. Wireless ad hoc networks are resource constrained in terms of bandwidth and power. The network nodes are communicates with the multihop paths. A packet can be overheard by a wireless node due to the broadcast nature of wireless communication. Data caching is widely used to efficiently reduce data access cost. Overheard data values can be maintained in data caches for fast data delivery. Data overhearing techniques are used to provide the cooperative cache facility. Data request and data reply are overheard to optimize cache placement and cache discovery. Cache placement mechanism is used to assign data under cache nodes. Cache storages are provided in the overheard nodes. Cache discovery process is used to identify a requested data from the cached data collection. The overhearing aided cache management mechanism is designed to manage cache placement and cache discovery process. Cache placement algorithm is used to assign cache data values. Data access cost and transmission cost factors are used in the data request process. Data consistency feature is adapted in the overhearing aided cache management mechanism. Cache management mechanism is used to maintain the cache storage levels. Cache storages are maintained with node movement factors. Multiple cache copies are maintained with primary and secondary cache models.

I. INTRODUCTION

In the next generation of wireless communication systems, there will be a need for the rapid deployment of independent mobile users. Significant examples include establishing survivable, efficient, dynamic communication for emergency/rescue operations, disaster relief efforts, and military networks. Such network scenarios cannot rely on centralized and organized connectivity, and can be conceived as applications of Mobile Ad Hoc Networks. A MANET is an autonomous collection of mobile users that communicate over relatively bandwidth constrained wireless links. Since the nodes are mobile, the network topology may change rapidly and unpredictably over time. The network is decentralized, where all network activity including discovering the topology and delivering messages must be executed by the nodes themselves, i.e., routing functionality will be incorporated into mobile nodes.

The set of applications for MANETs is diverse, ranging from small, static networks that are constrained by power sources, to large-scale, mobile, highly dynamic networks. The design of network protocols for these networks is a complex issue. Regardless of the application, MANETs need efficient distributed algorithms to determine network organization, link scheduling, and routing. However, determining viable routing paths and delivering messages in a decentralized environment where network topology fluctuates is not a well-defined problem. While the shortest path from a source to a destination in a static network is usually the optimal route, this idea is not easily extended to MANETs. Factors such as variable wireless link quality, propagation path loss, fading, multiuser interference, power expended, and topological changes, become relevant issues. The network should be able to adaptively alter the routing paths to alleviate any of these effects. Moreover, in a military environment, preservation of security, latency, reliability, intentional jamming, and recovery from failure are significant concerns. Military networks are designed to maintain a low probability of intercept and/or a low probability of detection. Hence, nodes prefer to radiate as little power as necessary and transmit as infrequently as possible, thus decreasing the probability of detection or interception. A lapse in any of these requirements may degrade the performance and dependability of the network.

The concepts and operational requirements associated with the current idea of MANETs are discussed in the mobile computing and networking literature, notably documents and standards developed by the MANET Working Group of the Routing Area of the Internet Engineering Task Force (IETF).

II. RELATED WORK

Cache placement algorithms mainly differ in the metric to evaluate the significance of data items. In popularity-based algorithm, placement decision is made based on how "popular" the data item is. Popularity can be represented by access frequency, time interval from last access, etc. These algorithms differ in whether and how the access of a neighbor host is considered. The work of Yin and Cao focuses on information cached at a node. Depending on the access frequency and cache space, a node may cache the data itself or the path to the nearest cache node. On the other hand, in a benefit-based algorithm, data items are selected according to how much "benefit," e.g., the reduction in message cost, query delay, energy consumption, etc., can be obtained by caching the data items. Compared with popularity-based approach, the benefit based one is more efficient, as shown in [1]. This is because "benefit" can directly and precisely reflect the objective of data caching, i.e., the reduction in data access cost. Therefore, we also adopt benefit-like metric in our design.

Since we consider the same caching scenario, we use the distributed algorithm in [1] as a comparison point of our work. In [1], the placement metric is defined as the reduction of data access cost by caching a data item. Each node calculates the data access frequency individually by observing the data request messages passing-by. Then, when a new data copy arrives, a node needs to decide whether or not this data should be cached. Based on the historic data access frequency of different data items and the distance to the nearest cache copy, the node can calculate the overall cost of access all the data items needed with or without the new data cache copy, respectively. If caching the data copy can reduce overall data access cost, the data copy is added to the cache space. Of course, replacement is also considered in the access cost calculation if there is not enough free space. Fan et al. [5] proposed the only caching algorithm that has considered the openness feature of wireless links. However, their work focuses on the contention status of a link in cache node selection. More precisely, the delay of the links is evaluated with respect to contention caused by concurrent requests and heuristics to achieve short access delay is designed. For the cache discovery problem in ad hoc networks, there have been few efforts made. In active cache discovery [1], a cache node proactively disseminates the cache information to other nodes so that each node can maintain a nearest cache table.

Hybrid approaches can be found in [8], [10]. The solution in [8] imposes a cluster-based hierarchy on a flat ad hoc network. It focuses on how to construct the hierarchy and how to discover the nearest cache node, inside and outside of the clusters. In [9], a node tries to find a cache by flooding within a predefined zone before sending the request to remote nodes. Dimokas et al. studied the selection of the sensor nodes which will take special roles in running the caching and request forwarding decisions. The works in [11], target at wireless mesh networks, a special type of ad hoc network. Since the network itself is hierarchical by nature, those works focus on how to make use of mesh routers in cache placement and discovery as cluster heads. To obtain such advantages, we design the overhearing-aided cooperative caching algorithm. We design benefit-like placement metric and adopt active cache discovery approach due to their advantage against popularity and passive discovery as shown in [1]. However, our design focuses on how to utilize overhearing on calculating data access cost and disseminating cache request/reply.

III. DATA CACHING SCHEMES

In recent years, wireless ad hoc network, as a promising technology for pervasive Internet access, has received a lot of attention from both academia and industry communities. In a wireless ad hoc network, there is no support of any

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

fixed infrastructure (e.g., the base stations in 3G networks or access points in wireless LANs), and network nodes communicates with each other through multihop paths. A network node can be any computing device, ranging from a mobile computer, e.g., smartphone, laptop, to a backbone network device, e.g., mesh routers, or even an embedded small sensor node [2]. Due to the advantages in flexible deployment, low cost and easy maintenance, wireless ad hoc networks are especially suitable for the scenarios where the deployment of network infrastructure is too costly or even impossible, e.g., outdoor assemblies, disaster recovery and battlefield. On the other hand, wireless ad hoc networks are resource constrained in terms of bandwidth, power, etc., so data access cost is a major concern. Data caching has been widely used to reduce data access cost in traditional computer networks. It is much more desirable and effective in wireless ad hoc networks. When data are delivered through multihop paths, caching the data at intermediate nodes can significantly reduce the message cost and consequently save various resources, from network bandwidth to battery power. Accessing data at cache node can also help reduce data access delay (AD).

Quite a lot of work has been conducted for data caching in wireless ad hoc networks, including cache placement, cache discovery [8] and cache consistency [7], [3]. Cache placement refers to determining where and what to cache; cache discovery refers to the mechanism to find and obtain a cached data item; and cache consistency means to ensure that the data value in cache copies is consistent with the source copy at the data server. The first two problems are so closely related that they are usually studied together.

The cache placement problem in ad hoc networks has been proved to be NP-hard, even if only one data item is considered. Existing works on cache placement mainly focus on how to make use of the data access frequency information and network topology information in selecting cache nodes [4]. For cache discovery, recent research has been focused on combining passive and active query approaches [10]. Although there have been quite a number of algorithms for ad hoc networks, the openness of wireless link has not been considered in designing cooperative caching system. Due to the openness of wireless links, a network node transmits data in a broadcast way by nature, so a packet can be received by any node within the transmission range even if the node is not the intended target of this transmission. In this case, we say that the node "overhears" the packet. Correspondingly, we say that the intended target node "hears" the packet. Here, the target node means the node specified as the receiver of the packet in MAC address. It can be an intermediate node in the routing path or the destination node of the massage.

In this paper, we propose a novel cooperative caching algorithm for wireless ad hoc networks by taking overhearing into consideration. Our algorithm makes use of the overhearing property to significantly improve the performance of data caching in several aspects. First, by overhearing, a requesting node can obtain data copies from an intermediate node forwarding the data copy so as to reduce data access cost. Second, overhearing helps collect more data access information, e.g., access frequency, which is necessary to make decision on cache placement.

However, making use of overhearing in data caching is not a trivial task. Several issues need to be addressed in order to make use of overhearing. First of all, what requests can be met by overhearing is a key issue. The cache discovery mechanism must be delicately designed so as to capture as many overhearing data copies as possible. Second, how to define and evaluate the access cost should be considered carefully for cache placement. In existing work, the access cost is defined by the distance between a requesting node and the nearest cache node. With overhearing, however, this definition is not applicable any longer because a request may be replied by an intermediate node that overhears the data from passing by replies for other requests. Although overhearing has been studied and used in cooperative communication achieve transmit diversity in signal propagation level, and routing protocols help construct route more efficiently, our work makes use of the application level information via overhearing and the focus is how to help the overhearing node rather than the destination node.

Based on the above observations and considerations, we have designed an overhearing-aided data caching algorithm that jointly considers the cache placement and cache discovery issues. To evaluate data access cost with overhearing, we define function to calculate the access cost with the distance between the requestor and the replier rather the cache node. To capture data copies through overhearing, we find out the sufficient condition for predictable data reply overhearing. In our design, mobility is not the primary concern, so we basically assume a static ad hoc network, i.e., the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

nodes do not move. However, we still propose enhancing mechanisms to handle node mobility to make our algorithm applicable to various environments. To our knowledge, this is the first overhearing-aided data caching algorithm.

To evaluate the performance of our proposed algorithm, we have conducted extensive simulations. We consider various scenarios with different parameter settings and compare our algorithm with the one proposed in [1], which is the best one among existing distributed caching algorithms for ad hoc networks. To show the advantage of our placement metric, we also simulate a naive overhearing algorithm as the baseline of overhearing caching. The simulation results show that our proposed algorithm performs much better than the other two algorithms in terms of both message cost and data access delay.

IV. PROBLEM STATEMENT

Cooperative caching algorithm is used to manage data transmission by overhearing data from intermediate nodes. Data request and data reply are overheard to optimize cache placement and cache discovery. Cache placement mechanism is used to assign data under cache nodes. Cache storages are provided in the overheard nodes. Cache discovery process is used to identify a requested data from the cached data collection. The overhearing aided cache management mechanism is designed to manage cache placement and cache discovery process. Cache placement algorithm is used to assign cache data values. Data access cost and transmission cost factors are used in the data request process. The following drawbacks are identified in the existing system. Node mobility is not handled. Cache consistency is not maintained under overhear environments. Cache storage is not managed. Cache duplication process is not controlled.

V. DISTRIBUTED CACHE MODEL WITH STABILITY AND STORAGE MANAGEMENT

The MANET cache management system is designed to manage cache storage and consistency levels. Cache request frequency is used for the cache storage space management process. Node mobility is managed with cache storage consideration. The system is divided into five major modules. They are MANET data provider, request reply process, cache update process, consistency management and cache storage management. The data provider module is designed to manage the shared data values. Request reply module is designed to handle the data access operations. Cache update process module is designed to store cached data in overheard nodes. Cache storage and their current level are maintained under the consistency management module. Cache storage spaces are maintained under the cache storage management module.

5.1. MANET Data Provider

Data provider shares the data files to the mobile ad hoc network nodes. Data files are transferred with reference to the requests. The users can collect the shared data file list from the providers. Response values are transferred through the intermediate nodes. We consider a wireless ad hoc network that consists of n nodes. The network nodes communicate by sending and receiving messages through wireless links. Such a network can be represented as an undirected graph G(V ,E) where the set of vertices V represents the nodes in the network, and E is the set of weighted edges in the graph. Two nodes that can communicate directly with each other are connected by an edge in the graph. For user applications or other purposes, there are multiple data items to be accessed by network nodes. Each data item is served by its source node. One source node may serve multiple data items. The data source is known to all the nodes. The knowledge of data source is obtained by some way outside the caching system. For example, when a user wants to browse some webpage, the IP address of the server of a target webpage can be resolved by DNS. To reduce data access cost, each network node caches multiple data items subject to its memory capacity limitation. The node with a cache copy of some data item is called a cache node of that data item. For simplicity of presentation, the data source is also viewed as a cache node.

5.2. Request Reply Process

Data request is submitted by the nodes. Request values are redirected to the provider through the intermediate nodes. Data request and reply information are updated in the overheard nodes. Data reply is prepared by the data provider nodes. The objective of cache placement is to minimize the total access cost of all the nodes in ad hoc networks by

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

appropriately placing the cache copies. The cache placement problem is known to be NP-hard. In the problem formulation below, for clarity of presentation, we assume all the data items have a uniform size of one memory unit. A node $_i$ has a memory with size m_i for data caching. We use a_{ij} to denote the access frequency for a node $_i$ to request the data item D_j, and d_{il} to denote the weighted distance between two nodes $_i$ and l. The source node of a data item D_j is denoted by S_j, for simplicity of presentation. The difference between the definitions of r(G,M) in (1) and (2) lies in the distance that a data request travels along the path between the requester and the nearest cache node. Without overhearing, the requested data item is obtained from a cache node. With overhearing, if the data are overheard by some intermediate node from passing by replies for other requests, the current request can be replied by that immediate node rather than the cache node specified as the destination of the request.

5.3. Cache Update Process

The overheard node collect data request and reply in the same coverage. Request and reply values are maintained with the cached data files. Data requests are verified with request history values. Cache data update is initiated for the new data request only. The key parameters are a_{ij} and op_{kj}. To collect and calculate aij and opkj, our algorithm can simply follow existing methods in other cooperative caching algorithms. A simple approach is letting a node calculate the parameter values based on the historical data request information observed by the node itself. That is, a node does the calculation based on the data access request/reply information heard and overheard. The access frequency can be calculated based on data request messages. To calculate the overhearing probability, an additional flag needs to be added to each reply message, which indicates whether the data copy included in the reply is obtained by overhearing. This method is simple and efficient since it does not need additional message exchange for collecting data access information. To increase the accuracy of calculation, the nodes can exchange the request information and overhearing probability values with each other. Such information can be piggybacked on normal data request and reply messages to save communication cost. How to make such calculation accurate is not the focus of our work. Based on the benefit of caching the data item D_j, node I can determine whether D_j should be cached. If the benefit of caching D_j exceeds some specific threshold value TH, node i will cache. TH can be specified by the user. If D_j is cached in some empty space, node i broadcasts an ADD(i, j) message to inform other nodes about the new cache copy. Otherwise, a cache copy of some data item D_t must have been replaced by D_j. Node i first needs to get the cache-list Mt for D_t from the source node St and then broadcasts an ADDEL(i, j, t, M_t) message.

5.4. Consistency Management

Cached data values are periodically verified with the data providers. Overheard node verifies the content level with actual data values. Data update time and cache verified time are checked in intervals. Counter values are used for the data consistency verification process. Cache discovery concerns about how a node can find the nearest copy of a data item requested. It includes two parts, i.e., request and reply. Different algorithms may have different methods to maintain the cache information, based on which the nearest cache node is discovered. We basically adopt the proactive approach where each node maintains a nearest cache list, which indicates the nearest node for each data item. Then, the request can be routed to the nearest cache node after looking up the list. The major challenge in our design of overhearing aided discovery is to find that, in what cases, overhearing is useful and utilizable for data caching. Overhearing may occur in numerous scenarios, but not all of them can be utilized. Intuitively, if a node knows that a data reply for some pending request will be heard or overheard, the later requests for the same data item can be certainly served without sending the request to the cache node. We call such an overhearing of data reply a "predictable" reply overhearing.

5.5. Cache Storage Management

Cache storage values are maintained with redundant data copies. Initial copy of the data is maintained under the primary data storage environment. Duplicate data values are maintained under the secondary data environment. Request frequency levels are considered in the cache delete process. The request message needs to be forwarded by the intermediate nodes between i and N_j. If an intermediate node k has a cache copy of D_j, it replies the requestor by sending a REP(i;D_j) message. Otherwise, it can still serve the request for D_j if its O_j is "true," i.e., another request for D_j is still pending and the reply for this request can be overheard, therequest will be put in the pending list. Node k will then stop propagating the request and wait for the reply for the pending request. If neither of the above two cases is true, node k will continually forward the request toward N_j and set O_j to be "true" so that it can serve other requests for the same data item when it gets the reply corresponding to the current request. Besides the intermediate nodes in the path between node i and N_j, some other nodes may also get the request for D_j by overhearing. If a node l can overhear a request forwarded by some intermediate node, it sets its O_j to be "true" so as to serve future requests for the same data item. Upon overhearing or hearing a reply message REP(i, D_j), a node resets O_j to be "false" and forwards the reply message to all pending requestors for D_j. Please notice that a request can only be pended and replied by intermediate nodes in the path between the requestor and destination cache node, so no duplicate reply copies will occur. To evaluate the performance of our design, we have conducted extensive simulations. For comparison purpose, we have also simulated two other algorithms. The other one is an algorithm as the baseline of overhearing. Since there is no existing cooperative caching algorithm with overhearing considered, we extended a naive frequency-based algorithm by adopting packet overheard. Simulation results show that our proposed algorithm performs much better than the other two, in terms of both message cost and access delay.

VI. CONCLUSION

Wireless data communications are carried out under the infrastructure less environment in adhoc networks. Wireless ad-hoc networks are used to share Internet under wireless nodes. Cache storages are used to improve the data delivery process. Overhear aided cache placement algorithm is used to maintain cache storages. Cache consistency and storage management features are integrated with the system. The system reduces the message cost for node communication. Access delay is minimized in the data caching scheme. Traffic overhead is controlled by the overhearing aided data caching scheme. The system achieves limited energy consumption in the MANET nodes. Storage management is used to remove unused data elements from the cached nodes.

REFERENCES

[1] B. Tang, H. Gupta, and S. Das, "Benefit-Based Data Caching in Ad Hoc Networks," IEEE Trans. Mobile Computing, vol. 7, no. 3, pp. 289-304, Mar. 2008.
[2] N. Dimokas, D. Katsaros, L. Tassiulas, Y. Manolopoulos, "High Performance, Low Complexity Cooperative Caching for Wireless Sensor Networks," Wireless Networks, vol. 4, no. 3, Apr. 2011.
[3] Y. Huang, J. Cao, Z. Wang, B. Jin, and Y. Feng, "Achieving Flexible Cache Consistency for Pervasive Internet Access," IEEE Trans. Parallel and Distributed Systems, vol. 21, no. 8, 2010.
[4] J. Zhao, P. Zhang, G. Cao, and C.R. Das, "Cooperative Caching in Wireless P2P Networks: Design, Implementation, and Evaluation," IEEE Trans. Parallel and Distributed Systems, Feb. 2010.
[5] X. Fan, J. Cao, and W. Wu, "Contention-Aware Data Caching in Wireless Multi-Hop Ad Hoc Networks," J. Parallel and Distributed Computing, vol. 71, no. 4, pp. 603-614, Apr. 2011.
[6] J. Cao, Y. Zhang, L. Xie, and G. Cao, "Data Consistency for Cooperative Caching in Mobile Environments," Computer, vol. 40, no. 4, pp. 60-66, Apr. 2007.
[7] R. Gunasekaran, V.P. Divya, and V. Rhymend Uthariaraj, "Mitigating Channel Usage in Cooperative Caching for Mobile Ad Hoc Networks," Int'l J. Ad Hoc and Ubiquitous Computing, pp. 87-99, 2011.
[8] N. Chauhan, L.K. Awasthi, N. Chand, R.C. Joshi, and M. Mishra, "A Cooperative Caching Strategy in Mobile Ad Hoc Networks Based on Clusters," Proc. Int'l Conf. Comm., Comp. and Security, Feb. 2011.
[9] G. Chiu and C. Young, "Exploiting In-Zone Broadcasts for Cache Sharing in Mobile Ad Hoc Networks," IEEE Trans. Mobile Computing, vol. 8, no. 3, pp. 384-396, Mar. 2009.
[10] Y. Du, K.S. Gupta, and G. Varsamopoulos, "Improving on- Demand Data Access Efficiency in MANETs with Cooperative Caching," Ad Hoc Networks, vol. 7, no. 3, pp. 579-598, 2009.
[11] M.K. Denko, T. Nkwe, and M.S. Obaidat, "Efficient Cooperative Caching with Improved Performance in Wireless Mesh Networks," Proc. IEEE Int'l Conf. Comm. (ICC), 2010.