

Cooperative Caching Technique for Multimedia Streaming Service in Mobile Ad-hoc Networks

Backhyun Kim¹ and Kyeongmo Park²⁺

¹ Faculty of Liberal Education, Incheon National University,
119 Academy-ro, Yeonsu-gu, Incheon, Korea

²⁺ Department of Computer Science and Information Engineering,
The Catholic University of Korea, 43 Jibong-ro, Bucheon-si, Gyeonggi-do, Korea

Abstract. Mobile ad-hoc networks are a network consisting of only free-moving nodes. The node itself performs all tasks related to communication. This requires efficient communication techniques such as transmission path management, service and transmission quality management, and security. The caching technique reduces the number of transmission paths and thus enables efficient usage of node energy and transmission bandwidth. In MANET, nodes are more often selected to neighboring nodes than path nodes. The active caching technique that unconditionally stores the received data increases the load of the node as the request rate increases, resulting in a decrease in the available time of the node. In this paper, we propose a method to cache and provide the received content irrespective of whether the content is requested or not. The caching node is selected based on the number of hops from the server among nodes having a direct link with the transmission path. The simulation compares and analyzes the cache hit ratio in terms of the maximum movement speed of the node and the various content request rates. From the simulation results, the proposed technique can obtain a similar cache hit ratio using a relatively small number of caching nodes.

Keywords: MANETs, Cooperative Caching, Aggressive Caching, Multimedia, Streaming Service

1. Introduction

Mobile ad-hoc networks (MANETs) do not need to establish communication infrastructure unlike a mobile communication network, and are composed of only communication nodes [1]. In MANETs, a node must handle all the processes for providing communication services such as transmission service, quality control, path management, security, etc. The movement of nodes in MANETs shortens the validity period of the path information generated in the previous step. Therefore, the node needs to continuously exchange route information to keep it as the latest information. As the frequency of this process increases, the consumption of network bandwidth and node energy increases proportionally.

The connection to the server is provided by establishing a link with the neighbour node set as the parent node for the path to the server of the requesting node. The path between the requesting node and the server is established through the iterative connection (hop-by-hop) of the parent nodes of each node [2]. MANETs have a relatively low transmission bandwidth and limited energy availability of nodes, unlike networks using communication infrastructure. Repeated transmission of the same content is inefficient in terms of network and energy usage. Therefore, storing and providing the contents of the server to other nodes can reduce the phenomenon of service requests being concentrated on the server and the transmission traffic.

In this paper, we propose a caching scheme that proposes a request service by setting up a proxy node for a route node and a neighbour node that can establish a direct link with a route node. A node is determined as a path node and a neighbour node in the path setting process between the server and the request node.

⁺ Corresponding author. Tel.: +82-02-2164-4365; fax: +82-02-2164-4777
E-mail address: kpark@catholic.ac.kr

Therefore, there is no need for an additional process to coordinate the nodes to set up the proxy. A node selected as a proxy stores this received content irrespective of its own request. Transport nodes or neighbour nodes provide services on behalf of the server if the requested content is stored in their cache. Therefore, when the cache hit occurs, the proposed caching scheme can reduce the node energy consumption and the transmission bandwidth due to the decrease of hop counts.

The composition of this paper is as follows. Section 2 describes related work. Section 3 introduces the caching scheme based on the number of hops proposed in this paper. Section 4 discusses the performance evaluation of the proposed scheme in terms of cache hit ratio with respect to request rate and node maximum movement speed. Finally, Section 5 describes the conclusion.

2. Related Work

The transmission path changes dynamically due to the continuous movement of the node in MANETs [2]. Proactive sets up a path between nodes at a fixed time interval so that the node uses a predetermined path without a route search process when requesting a service [3]. This scheme has the advantage that the latest path information is always maintained. However, this approach continuously consumes network bandwidth and node energy in the routing process even in the absence of a service request. Reactive is a method to set a path only when a service request occurs [4]. This performs the routing process only when necessary, and increases the network bandwidth and node energy usage efficiency compared to proactive method. However, whenever the service is requested, the route search is performed. As a result, the service delay occurs. As the number of request nodes increases, the load for management increases proportionally.

The proxy is located between client and server, acts as a relay, and stores data transmitted from the server [5][6]. The request generated by the client to the data stored in the proxy may be provided by the proxy. As compared with the case of using only a server, proxy technique can reduce the number of transmission paths, thereby reducing service delivery time and required transmission bandwidth. Service requests are not concentrated to the server and are distributed among the nodes, which can prevent bottlenecks phenomenon. The performance that can be achieved using a proxy depends on the location of the proxy in the network. If multiple proxies are located in a particular region or are located a greater distance from the server, the expected performance may not be achieved. By using location information such as GPS, a node in the proper location can be set as a proxy. If the proxy leaves the location, another node needs to be selected as a proxy. The previously selected proxy must transmit the stored content to the newly selected proxy to handle the service that occurs in the future.

Multimedia contents have a relatively large capacity compared with general data [7]. The link between the proxies should be maintained for a sufficient time to transmit all of the stored content. As this time decreases, the amount of content stored in the new proxy decreases proportionally, thus reducing the overall transmission efficiency. Therefore, it is not easy to guarantee the performance of MANETs by assigning a specific node as a proxy because of changes in the number of transmission paths and transmission bandwidth over time. When the request rate is high, the load of the network and the node is increased due to the packet flooding. It is possible to reduce the flooding by allowing the request packet to be delivered within a certain number of hops from the requesting node [8]. The available energy of a node in MANETs is limited. Dynamically selecting a proxy is an efficient way to increase the available time of a node. If the connection to the server is constantly guaranteed, the contents can be dynamically distributed among the nodes [9]. The nodes constituting the transmission path to deliver the contents transmitted from the server to the requesting node. In the CCN (Content Centric Network), the route nodes use In-network caching to store the transmitted content [10]. In MANETs, the transmission path is variable over time due to the movement of the node. The in-network caching scheme searches for the requested content only in the transmission path. Therefore, in MANETs, this approach is difficult to achieve a high cache hit ratio.

3. Hop-based Cooperative Caching Scheme

This paper proposes an aggressive caching scheme based on the number of transport hops. The following conditions are assumed for this purpose. The provision of seamless services is possible only in a fully connected state where there is a transmission path to another node. To do this, we initially place a sufficient

number of nodes in the network using the random scheme. The transmission path between the nodes is set using a proactive technique in which routing is performed by the content server every predetermined time (t_r). All nodes located in the network have the same transmission range R and are connected to other nodes under a hop-by-hop scheme. A mobile node or AP can be a content server and there is only one in the network. The content is composed of N frames, and each frame is encoded with a CBR (Constant Bit Rate). The content server initiates the routing process every t_r hours.

When routing is performed at time t_0 , the transmission path between the nodes is created as a tree structure with the content server as the root. Nodes consist of a parent node with less than 1, a sibling node with the same hop count, and child nodes with a hop count of 1 compared to its own hop count. At time $t_0 < t < t_0 + t_r$, each node has a transmission path consisting of a parent node and child nodes. The request packet includes a content server address A_c , a request node address A_r , a request content ID C_{id} , and a request content frame F_{id} . When the content server receives the request packet, the server transmits the requested content. This content is transferred to the requesting node through the reverse path over which the request packet was sent [14].

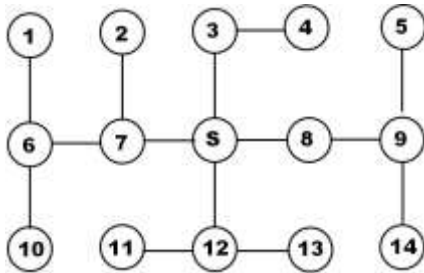


Fig. 1: An example of network topology at time t_0

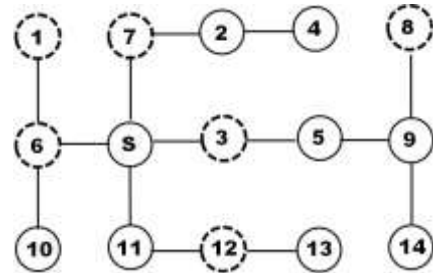


Fig. 2: An example of network topology at time $t_0 + t_r$

Fig. 1 shows an example of a network topology consisting of one content server S and 14 mobile nodes at time t_0 . The solid line represents the link between the nodes. Assume that node 1 requests content C_{id} at time $t_0 + \Delta t$. Node 1 first searches its own cache to find if the C_{id} is stored. If C_{id} is already stored, the content C_{id} is immediately reproduced. If not, the request packet is transmitted to the parent node, node 6. Upon receiving the request packet, the node 6 searches its own cache and confirms whether or not the C_{id} is stored. If so, it transmits the requested content to node 1 and discards the request packet instead of sending it to its parent node, node 7. If not, node 6 forwards the request packet to node 7. When the node 7 receives the request packet, it performs the same process as the node 6. When the content server receives the request packet from the node 1, it means that the corresponding content is not stored in the request node and the parent nodes. The C_{id} is transmitted along the reverse path $\{S, 7, 6, 1\}$ to node 1 by the content server.

In this paper, the node determines whether to cache received data based on the number of hops H_c from the server. The hop value of the server is 1. Therefore, if H_c is 1, adjacent nodes not designated as the receiving node do not store the received data. The H_c value of the child nodes of the server is 2. In the above description, node $\{3, 8, 12\}$ stores the received C_{id} without discarding if H_c is 2. The final destination of the C_{id} is node 1. Node $\{3, 8, 12\}$ does not have node 1 as a child node or a descendant node. Therefore, the received contents C_{id} are not transmitted to their child nodes. The node 7 delivers the received content C_{id} to the node 6, and the child node 2 also receives it. H_c of node 2 is 3. Node 2 discards the received content when H_c is set to 2 and stores the received content when H_c is 3 or higher. Nodes $\{6, 7\}$ are parent nodes constituting the transmission path between node 1 and content server S . In MANETs, content is delivered in a store-and-forward manner among the nodes constructing the path. Due to this transmission characteristic, the content transmitted to node 1 can be stored in node $\{6, 7\}$. When H_c is 2, the content C_{id} is stored in the node $\{3, 7, 8, 12\}$. If H_c is 3, the content C_{id} is stored in the node $\{2, 3, 6, 7, 8, 12\}$. When H_c is 4, the content C_{id} is stored in the nodes $\{1, 2, 3, 6, 7, 8, 10, 12\}$.

Suppose that the routing interval t_r and the sizes of the frame are the same. The first frame is transmitted up to $t_0 + t_r$ time. The second frame is transmitted from time $t_0 + t_r$ to $t_0 + 2t_r$. Assume that the transmission path changes at time $t_0 + t_r$ as shown in Fig. 2 due to the movement of the node. In Fig. 2, the dotted circle represents nodes storing the first frame of the content C_{id} . If $H_c \geq 2$, the second frame of C_{id} is stored in

node {1, 3, 6, 7, 10, 11}. Assume that at time $t_0 + tr \leq t < t_0 + 2tr$, node {3, 6, 7, 8, 12} requests the Cid that node 1 requested at time t_0 . In this case, since the first frame of the content is stored in their own caches, a local cache hit occurs. Therefore, they do not transmit the request packet to the content server but reproduce it using the stored content. When node {2, 4} requests content Cid, node 7, which is the parent node located on the path to the content server, stores the first frame of Cid. The requested service is provided by node 7, not from the content server, and is referred to as a path cache hit. Similarly, the path cache hits are issued by node 6 for node 10 and node 12 for node 13. Unlike the previous cases, nodes {9, 14} are serviced by node {8}. Node 8 is a child node that is not a parent node located on the transmission path to the content server. When node 9 sends a request packet to node 5, node 8 receives it and sends it to node 9 because it is storing the content. This case is referred to as an adjacent cache hit. When the node 9 receives a response packet indicating that the content is stored from the node 8, the node 9 transmits a packet cancelling the request packet to the node 5. To this end, the node composing the transmission path waits for twait time when receiving the request packet and transmits it to the parent node if the cancel packet is not received.

The requested content is stored in the request node, the route node and the adjacent node. Let P_R , P_P , and P_A be the probability of being a request node, the probability of being selected as a transmission path, and the probability of being adjacent to a transmission path, respectively. The probability that the node stores the content is $P_S = P_R + P_P + P_A$. In the proposed scheme, the cache hit ratio P_C of the requested content can be obtained as the sum of the local cache hit ratio P_{LC} , the path cache hit probability P_{PC} , and the adjacent cache hit probability P_{AC} .

$$P_C = P_{LC} + P_{PC} + P_{AC} = P_S + \sum_{i=1}^{H-2} P_S + \bigcup_{i=1}^{Hc-1} \sum_{i=1}^{N_A} P_S \quad (1)$$

In equation (1), H is hop counts from the server to the requesting node, and N_A is the number of adjacent nodes per transmitting node. Let the content have a size of N seconds and the transmission rate per second is S_{CBR} . The requested content may be transmitted by a server, a route node, or a neighbouring node. The content having N seconds is transmitted N_S times from the server, N_P times from the route node, N_A times from the adjacent node, and N_L is not transmitted when stored in the request node, and $N = N_S + N_P + N_A + N_L$. Therefore, the bandwidth B_T required to transmit the requested content is equal to Equation (2).

$$B_T = \{ \sum_{i=1}^{N_S} (H_{S,i} - 1) + \sum_{i=1}^{N_P} (H_{P,i} - 1) + \sum_{i=1}^{N_A} H_{P,i} \} S_{CBR} \quad (2)$$

H_S and H_P represent the number of hops from the request node to the server every second and the number of hops to the route node where the cache hit occurred. When the proposed caching scheme is used, the bandwidth that is reduced to transmit the content is given by Equation (3).

$$B_{nc} - B_T = \{ \sum_{i=1}^{N_P} (H_{S,i} - H_{P,i}) + \sum_{i=1}^{N_A} (H_{S,i} - H_{P,i} - 1) + \sum_{i=1}^{N_L} (H_{S,i} - 1) \} S_{CBR} \quad (3)$$

4. Simulation and Analysis

This section shows the simulation results for analysing the characteristics of the proposed caching technique. Only one content server exists and provides 100 contents having the same size. Each content has a size of 30 seconds and the playback rate $SCBR$ is 1 Mbps. The node cache size is $300SCBR$, which can store 10 contents, and it uses Least Recently Used (LRU) cache management algorithm [11]. The simulation network has a size of $100\text{ m} \times 100\text{ m}$ and 250 nodes. The transmission range of each node is selected from 10m to 20m using the Gaussian distribution and the mean value is 15m. Zipf's like distribution is used to make the content have different popularity, and the skew factor α is set to 0.85 [12]. Based on the popularity, the content provided by the server is requested at $\lambda = \{3, 6, 9, 12\}$ requests / minute, i.e. 3, 6, 9, or 12 requests per minute. The node mobility is defined by using a random waypoint. The maximum speed is 6, 8, 10 km / h and the pause time is 5 seconds [13]. The routing protocol is DSDV and the routing interval is 1 second [2][3]. The simulation was performed for 1000 seconds and the average of 10 results was taken.

In the initial state, the nodes are located at arbitrary coordinates of the network. The node uses a multi-hop method to guarantee connection with other nodes. Fig. 3 shows the cache hit ratios when the number of hops (H_c) to select a proxy node to perform caching is 0, 2, 3, and 4. On the X axis, 6, 8, and 10 are the maximum movement speed of the node. The request rate λ is 3. The cache hit rate is calculated as Equation (1) by the local cache hit, the route cache hit, and the adjacent cache hit ratio. If the value of H_c is 0, all neighbour nodes perform caching. If H_c is 2, the nodes located within the transmission range of the content

server, all nodes capable of direct link establishment with content server, are selected as the proxy. When H_c is 3, the proxy nodes are selected as the neighbour nodes of the server and the neighbour nodes of the child nodes of the server toward the request node. If H_c is 4, the server, the child nodes of the server, and the neighbour nodes of the grandchild nodes of the server perform caching. The simulation results show that cache hit ratio is not dependent on H_c . Simulation results show that cache hit ratio is not affected by H_c . Therefore, it is most efficient that the role of storing the transmitted contents is performed only on the nodes that can establish a direct link with the content server.

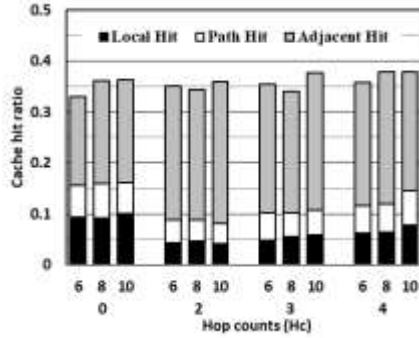


Fig. 3: Cache hit ratio according to the number of hops selected by the proxy

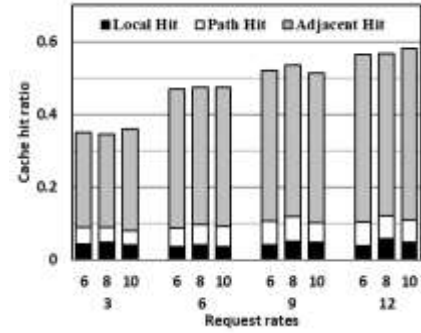


Fig. 4: Cache hit ratio when the request rates are 3, 6, 9 and 12, respectively (requests/minute)

Fig. 4 shows the cache hit ratios of the proposed technique with the maximum movement speed when the request rates λ are 3, 6, 9, and 12. The H_c value is 2. Simulation results show that the maximum movement speed of the node does not affect the cache hit ratio as shown in Fig. 3. Since the number of transmission streams from the server increases as the request rate increases, the probabilities that a node becomes a request node, route node, or neighbour node are also increased. The probability that high popularity content will be cached is increased. Thus, the cache hit ratio has a proportional relationship with the request rate.

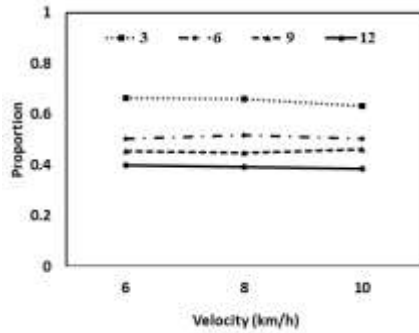


Fig. 5: Actual transfers / transfers when not using caching

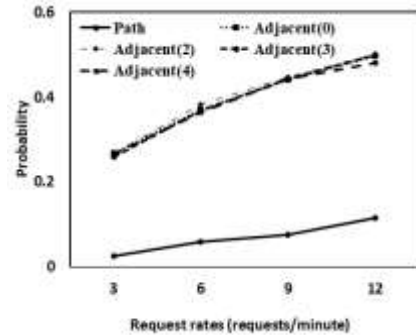


Fig. 6: Probability of being selected as a path node and an adjacent node in the request rates

Fig. 5 shows the proportion of the number of transmission paths when caching is not used and the number of transmission paths when using the proposed caching technique when the request rates λ are 3, 6, 9, and 12. The value of H_c is 2. The results show that the decrease in the number of transmission paths increases as the request rate increases. The reduced amount of transmission path can be obtained from Equation (2). The proportion of the number of transmission paths depends on the request rate rather than the moving speed, similar to the case of Fig. 4. Fig. 6 shows the probability of selecting a path node or a neighbour node when the request rates λ are 3, 6, 9, and 12. Numbers in parentheses indicate H_c values. The maximum travel speed of the node is 6 km / h. As the H_c value increases, the probability of being selected as an adjacent node increases. The proposed caching scheme has little effect on the maximum movement speed of the node and improves the performance as the request rate increases.

5. Conclusion

This paper proposes a caching scheme to efficiently transmit multimedia contents in mobile ad-hoc networks. In the proposed scheme, the content to be transmitted is stored in the node composing the

transmission path and the adjacent node in the transmission path, and is provided to the user. The neighbouring node performing caching is selected based on the number of transport hops from the content server. The proposed caching scheme is evaluated from the viewpoint of the maximum movement speed of the node, the cache hit ratio using the content request rate, the reduced number of transmission paths and the probability of caching node selection. Simulation results are compared with the result of active caching technique that unconditionally stores received data. Simulation results show that the cache hit ratio is less related to the number of hops of neighbour nodes. Since the content is requested based on popularity, the request rate and the cache hit rate are in proportion. Therefore, it can be confirmed that only the nodes that can establish a direct link with the content server perform caching efficiently in terms of the load of the node and the use of the network bandwidth. The simulation results show that most cache hits occur in adjacent nodes. The proposed caching scheme can be an alternative technique to enable efficient use of transmission bandwidth and node energy in high node density environment.

6. Acknowledgements

This work was supported by The Catholic University of Korea Research Grant in 2018

7. References

- [1] M. Conti and S. Giordano. Mobile ad hoc networking: milestones, challenges, and new research directions. *IEEE Communications Magazine*, 2014, **52** (1), pp. 85-96.
- [2] Aggarwal, S. Gandhi, and N. Chaubey. Performance analysis of AODV, DSDV and DSR in MANETS. *International Journal of Distributed and Parallel Systems*, 2011, **2** (6), pp. 167-177.
- [3] S. Mitra, R. Islam, K. Mukherjee, A. Das, and S. Nandi. A Modified Algorithmic Approach of DSDV Routing Protocol for Wireless Ad Hoc Network, *IOSR Journal of Computer Engineering*, 2014, **14** (4), pp. 49-54.
- [4] N. Chaubey, A. Aggarwal, S. Gandhi, and K. A. Jani. Performance analysis of TSDRP and AODV routing protocol under black hole attacks in manets by varying network size. In *Advanced Computing and Communication Technologies (ACCT)*, IEEE, 2015, pp. 320-324.
- [5] P. Srivastava and R. Kumar. An efficient proxy adaptive gateway discovery algorithm based on quality of service parameters. *International Journal of Communication Networks and Distributed Systems*, 2016, **16** (3), pp. 261-280.
- [6] P. Cao, and S. Irani. Cost-aware www proxy caching algorithms. In *Usenix symposium on internet technologies and systems*, 1997, **12** (97), pp. 193-206.
- [7] G. Kioumourtzis, A. Gkamas, and C. Bouras. Mobile Ad hoc networks (MANETs) for multimedia transmission. In *Encyclopedia of Information Science and Technology*, 2015, pp. 6239-6248.
- [8] S. Lim, W. C. Lee, G. Cao, and C. R. Das. Performance comparison of cache invalidation strategies for internet-based mobile ad hoc networks. *IEEE international conference on Mobile Ad-hoc and Sensor Systems (MASS)*, 2004, pp. 104-113.
- [9] L. Zhou, T. Zhang, X. Xu, Z. Zeng, and Y. Liu. Broadcasting based neighborhood cooperative caching for content centric ad hoc networks. In *Communications in China (ICCC)*, 2015 IEEE/CIC International Conference on. IEEE, 2015, pp. 1-5.
- [10] R. A. Rehman and B. S. Kim. LOMCF: Forwarding and Caching in Named Data Networking Based MANETs. *IEEE Transactions on Vehicular Technology*, 2017, **66** (10), pp. 9350-9364.
- [11] N. Laoutaris, H. Che, and I. Stavrakakis. The LCD interconnection of LRU caches and its analysis. *Performance Evaluation*, 2006, **63** (7), pp. 609-634.
- [12] Y. Ma and A. Jamalipour. A cooperative cache-based content delivery framework for intermittently connected mobile ad hoc networks. *IEEE Transactions on Wireless Communications*, 2010, **9** (1), pp. 366-373.
- [13] P. Nayak and P. Sinha. Analysis of Random Way Point and Random Walk Mobility Model for Reactive Routing Protocols for MANET Using NetSim Simulator. In *Artificial Intelligence, Modelling and Simulation (AIMS)*, IEEE, 2015, pp. 427-432.