Social-relation Aware Routing Protocol in Mobile Ad hoc Network

Jisun An, Yangwoo Ko, and Dongman Lee
Information and Communications University
{ jisun, newcat, dmlee } @ icu.ac.kr

In this paper, we consider mobile ad hoc network routing protocols with respect to content sharing applications. We show that by utilizing social relations among participant, routing protocols can help such applications be more efficient in terms of access time and network overhead.

Introduction

Mobile social software (or MoSoSo hereinafter) finds social networks among proximate users based on their interests and assists users to create and maintain these social relations through spontaneous interactions such as content sharing. Until now most MoSoSo content sharing applications running in a mobile ad hoc network (or MANET hereinafter) assume that all devices are within 1-hop distance [1][2][3][4]. This restriction on communication range depreciates the value of the applications generally because it reduces the probability of actually finding matches [5]. Moreover, this could limit the participation of users since some people are embarrassed if they should confront strangers face to face to exchange personal information. Thus multi-hop communication is necessary to increase the feasibility of these applications.

As the coverage of a MoSoSo application gets bigger, there are increasingly diverse users. They are grouped according to shared interests and contents are mostly exchanged within groups. Often there are popular contents in a group that are requested by most of users in the group. These application level characteristics could be exploited in routing to achieve enhanced efficacy. There have been several approaches to reflect application level requirements in routing protocols. For example, [6] proposes a QOS aware routing protocol that considers the quality of services of intermediate nodes as a key parameter for establishing paths. However, except routings for delay tolerant networks or opportunistic networks [7], to our best knowledge, there is no work considering social relations as a parameter for MANET routing.

In MoSoSo scenarios, a user's content will be consumed by a group of users interested in them and these consumers may share common interests with the content owner. Social relation[1] can be a good metric to predict whose contents will be requested by whom. Based on this observation, we propose a social relation aware routing protocol for content sharing MoSoSo applications in a MANET environment. The following sections describe our scheme, which has three main parts; utilizing neighbor nodes as cache, constructing a social relation aware routing table, and providing a social relation aware request/reply service.

Social-relation aware Routing Protocol

In our scheme, social relation that calculated based on shared interest is utilized in various ways. In social-relation aware routing table, social relation is exploited by utility value. High utility value means having more chance to consume content thus a path that has higher utility value will be chosen. In social-relation aware caching, cache hit ratio for content can be calculated by social relation between user and content. Both cache hit ratios of user and neighbors will be considered for determining whether cache content or not. Request/Reply service is performed using these two methods.

Assumptions

A user's interhest is described in a user profile stored in a user's portable devices. User profile such as [8] is distributed to other user and used for finding socially related user. A piece of content is conducted with its meta information, which is used for interest-based match with user profiles. A list of contents a user owns is exchanged between users. A way to match a user profile with meta information contained in a content item is assumed to be given and resulted in a scalar value, for example, 0 for no match and 1 for perfect match. In addition, a method to get social relation is important in social science but it is out of our research scope thus we assume that social relation can be calculated by some formula.

[1] Though there exist several factors constituting social relation between a pair of users, this paper only considers shared interests in contents and the inclusion of additional factors, especially those related with node mobility as a group, are left as future work.

Utilizing neighbor node as a Cache

A node caches a piece of content that might be consumed in the future. When content is passed by a node, a node decides whether take it or not according to cache hit ratio of that content, called caching condition. For instance, it can be calculated by an equation below where c is received content, i is a value of interest of a user.

$$F(c) = I(c, i) = 1 - |c - i| \quad (1)$$

Social relation aware request/reply service

When an end user requests a piece of content, in order to reduce access time, firstly a node checks owned cache. If the content is not contained, node requests to a source node which owns that content originally. A request message will be forwarded through intermediate nodes along a routing path built by the proposed scheme, and a node which contains requested content (intermediate node or source node) will reply with the content.

Social relation aware Routing table Construction

As we mentioned before, social relation can be a metric for predicting whose contents will be requested by whom. Thus for a specific content, we can determine by using social relation a path that has a more chance to consume in the future. In the proposed scheme, all nodes simply broadcast its routing table and based on advertised routing information, each node updates its own routing table based on the update algorithm described in Table1. We measure a utility value which is cache efficiency of a node compared with cost. A utility function, U(d), is defined by an equation below where d is a destination that contains requested content.

$$U(d) = \frac{\text{Sum of cache hit ratio values of nodes along a path to d}}{\text{Total number of hops to d}} \quad (2)$$

In this equation, cache hit ratio can be calculated by SR(intermediate node, d). A path that has the highest value of U(d) means that intermediate nodes in a path have a higher chance to cache that content. Because U(d) is additive, a routing table entry is updated simply.

Each node has a routing table that contains records for recognizable destinations. Each record consists of a destination address, the number of hops to the destination, a next hop to forward, a cumulative value of cache hit ratio reaching the destination, the destination's interest value, and time stamp. Thus a utility value can be calculated using routing table entries based on an equation below.

$$U(d) = \frac{\text{Cache hit ratio}_{\text{advertised}} + \text{Cache hit ratio}_{\text{mine}}}{1 + \text{hop count}_{\text{advertised}}} \quad (3)$$

When a node receives a routing table from its neighbor, it updates its own routing table according to the following update algorithm.

Table 1. Routing Table Update Algorithm

1. Add one hop to the hop count for each advertised destination
2. Repeat the following steps for each advertised destination:
   - Calculate a cache hit ratio using interest value
   - Calculate a utility value for destination using equation (3)
   1. If (destination not in the routing table)
      - add the advertised information to the routing table
   2. Else
      1. If (advertised info. has a bigger sequence number)
         - Replace entry in routing table with the advertised one
      2. Else if (advertised info. has a same sequence number)
         1. If(advertised info. has a smaller number of hops)
            - Replace entry in routing table with the advertised one
         2. Else if (advertised info. has a same number of hops)
            1. If (calculated utility value greater than utility value in the routing table)
               - Replace entry in routing table with the advertised one

Based on this update algorithm, a path that confirms highest probability of cache hit will be selected even though it is not a shortest path.

Evaluation

We initially simulate a request/reply of content to compare our scheme to proactive shortest path algorithm by using NS2 simulator. Simulation environment is revealed in Table 2.

Table 2. Simulation environment (settings)

Number of nodes	40
1 hop comm. range	250m
Area	1000x1000 (m^2)
Number of request	160
Size of content	1000 bytes
Simulation time	550s
Routing table exchange time	15s

10 nodes are requests 16 pieces of content based on their interest. We evaluates a proactive shortest path algorithm and social-relation aware routing algorithm(SRRT) by number of received content, average access time, and network overhead. In addition, for precise analyzing we count a number of content from owned cache. The results are presented in the followings.

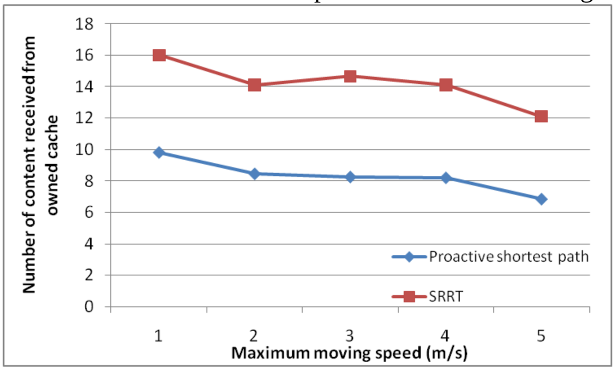

Figure 1. Number of received content

Based on the result in the figure 1, we can know that social relation can be a metric to predict whose content will be requested by whom in the future since SRRT user can get more content from own cache which means SRRT provides good metric for caching content that might be requested by user. Moreover, this result implicates that possibility of utilization of social relation in other kinds of application. For example, not for an application requires simultaneous request/reply interaction but for an application needs proactive caching for later uses, social relation can be used for reduce a huge amount of information flow.

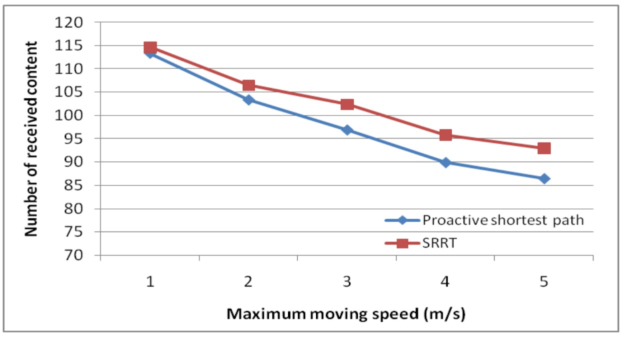

Figure 2. Number of received content

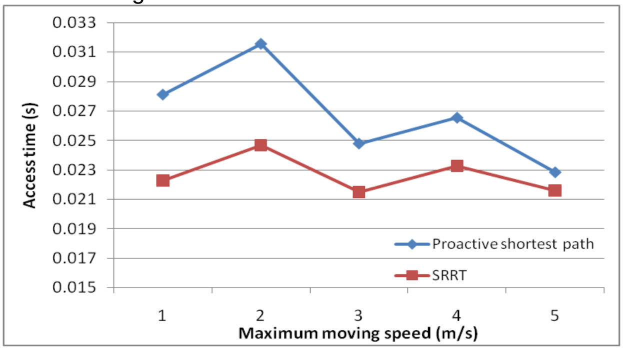

Figure 3. Average access time

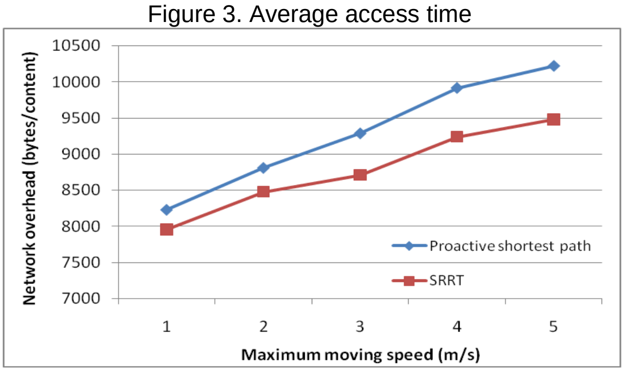

Figure 4. Network overhead

According to results in figure 2, 3, and 4, we can find out that our SRRT performs better in terms of delivery ratio, access time, and network overhead than proactive shortest path algorithm. As you know accuracy of routing table is depreciated by increasing node's maximum moving speed. Since SRRT got more content from owned cache, the degradation of performance in SRRT by high mobility is smaller than proactive shortest path's. The effect of social relation can be proved by this result, however, it seems insignificant. We expect that if the simulation were more realistic, for example, we assumes that content can be transmitted by using only one packet, the effect of social relation will be obvious.

Conclusion and Future works

In this paper, we propose social relation aware mobile ad hoc network routing protocols with respect to content sharing applications. It has three main parts; utilizing neighbor nodes as cache, constructing a social relation aware routing table, and providing a social relation aware request/reply service. Based on result of our simulation we can know that utilizing social relations among participant can help routing protocols to support such applications be more efficient in terms of reliability, access time and network overhead in highly mobile environment.

This paper revealed a possibility of adopting social relation to network layer. As we mentioned before, social relation can be utilized in an application that uses asynchronous interaction and requires proactive cache for later uses since SRRT gets more content from owned cache. In addition, the results show that our scheme performs better than proactive shortest path algorithm; however, it seems insignificant improvement. For holding our argument, we need to simulate based on more realistic environment to show effect of social relation clearly.

References

[1] N. Eagle and A. Pentland, "Social serendipity: Mobilizing social software," *IEEE Pervasive Computing*, vol. 4, no. 2, pp. 28–34, 2005.

[2] Younghee Jung, Jan Blom, Per Persson, "Scent field trial: understanding emerging social interaction," In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services MobileHCI '06*, Espoo, Finland.

[3] http://www.jambo.net

[4] Michimune Kohno, Jun Rekimoto, "Searching common experience: a social communication tool based on mobile ad-hoc networking," In *Proceedings of the 7th international conference on Human computer interaction with mobile devices & services MobileHCI '05*, Salzburg, Austria.

[5] Panayotis Antoniadis and Costas Courcoubetis, "The case of multi-hop peer-to-peer implementation of mobile social applications," In *International Conference on Systems and Networks Communication (ICSNC'06)*, Tahiti, French Polynesia, 2006

[6] S.-J. Lee and M. Gerla, "Dynamic Load-Aware Routing in Ad Hoc Networks," *Proc. ICC*, Helsinki, Finland, June 2001

[7] C. Boldrini, M. Conti, I. Iacopini, and A. Passarella, "HiBOp: a History Based Routing Protocol for Opportunistic Networks", In *Proc. IEEE WoWMoM 2007*, Helsinki, Finland, June 2007

[8] D. Bottazzi, R. Montanari, and A. Toninelli, "Context-Aware Middleware for Anytime, Anywhere Social Networks," *Intelligent Systems, IEEE*, vol. 22, 2007, pp. 23-32.