Adaptive Data Dissemination in Mobile ad-hoc Networks

Joos-Hendrik Böse, Frank Bregulla, Katharina Hahn, Manuel Scholz

Freie Universität Berlin, Institute of Computer Science, Takustr. 9, 14195 Berlin
{boese, bregulla, khahn, mscholz}@mi.fu-berlin.de

Abstract: In this paper we examine data dissemination in MANETs using various push and pull based protocols and a combination of both. We evaluate the protocols considering the load for each node and the resulting data freshness. Furthermore we introduce an adaptive pull protocol which enhances dissemination and saves up to 13% of the network load achieving the same freshness rate as other protocols.

1 Introduction

The advent of wireless communication technology embedded into small (handheld sized) devices enables the appearance of mobile ad-hoc networks (MANETs) which are expected to be deployed in scenarios where infrastructure based networks are not applicable. The most common scenarios are disaster and battlefield scenarios. A crucial demand of applications operating on shared data is the availability of up-to-date data which is built on data dissemination. Dissemination strategies used in structured networks are mostly based on knowledge about the topology which are not applicable in MANETs. In this paper we compare push and pull based data dissemination in MANETs as well as combinations of both. Data dissemination strategies are rated based on their costs in terms of network load within the MANET and overall freshness of data. The main challenge is to reach high freshness at minimal network load. In addition to the evaluation of simple push and pull based techniques we propose a novel dissemination approach, which adapts its query behavior to the frequency of object propagation.
The remainder of the paper is structured as follows: Section 2 gives a brief overview of related work and Section 3 describes the system model and application scenarios. In Section 4 different dissemination strategies are described and a new adaptive pull strategy is presented. In Section 5 results are discussed and Section 6 concludes the work and gives an outlook on future work.

2 Related Work

Research on data dissemination in MANETs often focuses on flooding techniques. In [WC02] various flooding protocols are classified and compared. Other approaches propose epidemic strategies, where node movement itself is used to spread data in the network

[DQA04]. A system also using pull based dissemination is presented in [LW04]. But this work only considers the hit rate of caches. A system for data dissemination and prefetching in mobile environments has been proposed in [PS00]. The main focus of [PS00] is the effect of mobility and query patterns on the dissemination. In [PS01] power conservation, wireless coverage and cooperation are taken into account but the amount of data that is needed to disseminate the information is not considered. In opposite to [WC02] we will focus on pull based protocols and a combination with simple flooding. In contrast to [PS00, PS01, LW04, DQA04] we take the network load into account and focus on general dissemination instead of selective dissemination. Moreover we introduce a new adaptive pull protocol enhancing the efficiency of dissemination by reducing the network load.

3 Scenario and System Model

Typical application scenarios in MANETs using data dissemination are news spreading systems. An example for such applications is the exchange of information of aides in a disaster area to coordinate relief actions. Every participating client is a rescue worker carrying a PDA like device using WLAN (IEEE 802.11) for communication. We assume that every object has a unique identifier. We also assume an isolated system i.e. we do not consider nodes leaving or entering the MANET nor are nodes turned off. Due to the characteristics of the used pull protocols this assumption does not have a strong impact on the results.

4 Dissemination Strategies

In this paper the main focus is on pull protocols described in this section. For push based dissemination we use a simple flooding strategy which is initiated each time when new objects have been created. [1]

Static Pull Protocols: The pull protocols used in the experiments of this paper are all based on the same basic synchronization protocol where the pulling client tries to synchronize itself with the pulled party. This is done by exchanging the IDs of owned objects. Static pulling mainly depends on its pull interval. When using static pulling all clients in the network use the same pull frequency which is predefined.

Adaptive Pulling: The main idea of our adaptive pulling protocol is to use status information about the dissemination progress to initiate the pull request only when new data is available. This is done by exploiting the broadcast characteristic of radio signals. The protocol acts as follows: First object IDs are sent by the pulling client and answered by its neighbors by broadcasting. The primary goal of this action is to synchronize the pulling client with one of the pulled clients. In addition dissemination information is hereby spread to all clients in transmission range. Thus every involved client and also the neighbors of the pulled clients keep track of IDs they do not own and therefore know about

[1] To avoid continuous flooding every client only broadcasts the object when it is not known to him.

missing objects. This information is used to decide when to initiate the next pull request. A node initiates a pull request whenever its number of unknown objects exceeds a given threshold value x. Higher x values result in less frequent pull requests with more objects being transferred while smaller x values result in more frequent pull requests with fewer objects being transferred. To avoid pull storms and to enhance dissemination the adaptive pulling also uses a static pull interval in case of no adaptive pull request being initiated.

5 Experimental Studies

In this section we describe the experimental studies and their results. For the simulations ns2 is used. The simulated network utilizes an IEEE 802.11 conform MAC protocol. In all simulations the nodes move on a $1000m \times 1000m$ rectangular area using the random waypoint mobility model. The duration of all scenarios is one hour. We assume a transmission range of $100m$, a bandwidth of $11Mbit/s$ for all participating nodes, and an object size of $256byte$. We examine a total of 12 scenarios with different numbers of nodes (50, 100, 150), different numbers of updates (100, 400), and different update distributions (uniformly, gaussian over time) to observe the impact of these parameters on the dissemination. We compare the following protocols in each of the scenarios: 1. Static pull with intervals of 30 to 360 seconds. 2. Flooding combined with static pull with intervals of 60 to 720 seconds. 3. Adaptive pull with threshold values from 0 to 8. 4. Flooding combined with adaptive pull with threshold values from 0 to 8. To evaluate the protocols we consider the following values:

Load: The data load represents the average sum of all sent and received data of each node for the duration of the experiment.
Freshness: The freshness rate indicates how fast the updates are disseminated. The total freshness is computed by averaging over the values of all single nodes. The formula is based on the concept of Cho and Garcia-Molina shown in [CGM00].

In the following we will discuss the results of 3 scenarios in detail. The results of the experiments are shown in Figure 1.
Every protocol with particular parameter values is represented as a point in the coordinate system. The load is shown on the abscissa, the freshness on the ordinate. Therefore the efficiency of protocols increases from lower right to upper left of the coordinate system. Each diagram depicts four curves representing different protocols. Each curve is formed by connecting the differently parameterized results of the same protocol. The curves including adaptive pull always consist of five different results representing the five different threshold values (0,1,2,4,8). Expectedly the protocols are always ordered according to their threshold values from upper right (threshold value 0) to lower left (threshold value 8). The single results using static pull represent the different time intervals of the protocol. Results of each protocol form a logarithmic-like curve. This is due to the fact that the synchronization process of the pull protocols produces load whether objects are exchanged or not. Thus, more frequent pull requests result in more useless synchronization attempts which produce higher load without enhancing freshness.

Comparing static pull protocols with the combination of static pull and flooding you can see that static pull protocols are more efficient considering lower freshness values. The combination of flooding and static pull performs better considering higher freshness values. This is due to the fact, that flooding reaches nodes in the same partition very efficiently without synchronization overhead produced by pull protocols. In scenarios with a higher node density (150 nodes) you can see that the difference of freshness values is greater. This is because of the stronger influence of the flooding protocol because it benefits from a higher connectivity (i.e. larger partitions).

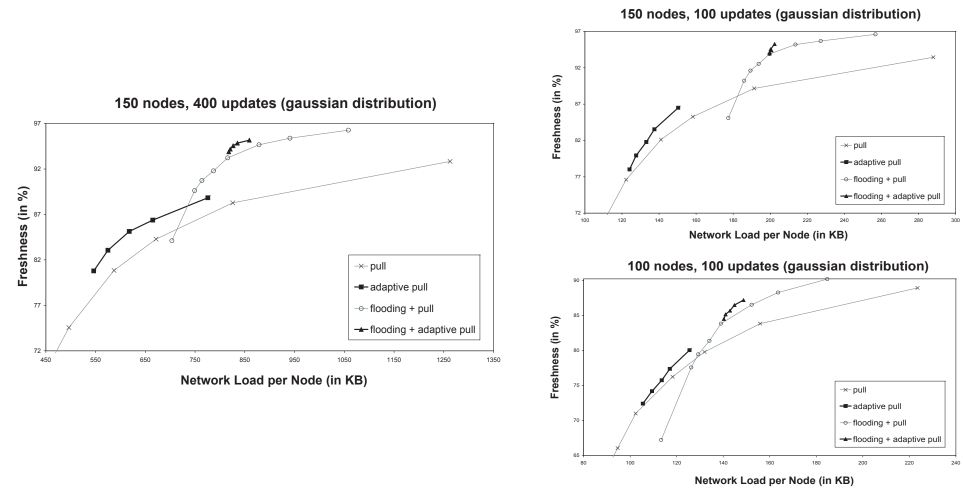

Abbildung 1: Simulation Results

The protocols using adaptive pull instead of static pull show the same characteristics, i.e. the combined protocols perform better considering a higher freshness level and the pull protocols perform better considering a lower freshness level. All adaptive protocols perform better than the protocols without adaptive pulling. Especially in scenarios with higher node density (150 nodes) the adaptive protocols perform well and need up to 13% less load to achieve the same freshness rate. This relies on the fact that adaptive pull protocols benefit from more nodes overhearing spread object IDs. Simulation results of scenarios with lower node densities (50 nodes) show that there is no significant benefit when using adaptive pull protocols. Adaptive protocols perform slightly better in scenarios with a higher update frequency (400 updates) because more information in the network produces more status information which results in more efficient adaptive pull requests.
The benefit of adaptive protocols in combination with flooding is lower (up to 7%) than for pull protocols. The reason is that the adaptive protocol can only enhance the dissemination for nodes not being in the partition of the initiating node. All other nodes already received

the object by flooding. That also explains the fact that the benefit of the adaptive protocols in combination with flooding is lower in scenarios with higher node densities (150 nodes). We also examined scenarios with objects distributed uniformly over time which matches the characteristic of the static protocols. This is the reason they perform better in these scenarios resulting in a lower benefit using adaptive pull protocols.

6 Conclusion and Future Work

In this paper we have tested various dissemination protocols and focused on their efficiency considering load vs. freshness. We have shown that pull protocols perform better considering a lower freshness level and that a combination of flooding and pull protocols performs better considering a higher freshness level. We also introduced an adaptive pull protocol which uses overheard status information and compared it to other protocols. We have shown that adaptive protocols perform better in all scenarios saving up to 13% of the load to reach the same freshness rate as other protocols. In the future we want to refine the adaptive protocol and take a look at scenarios with clusters and more heterogeneous node densities. Furthermore we want to examine the impact of different caching strategies and we want to take energy status information into account.

Bibliography

[CGM00] Cho, J. and Garcia-Molina, H.: Synchronizing a database to improve freshness. In: *Proceedings of the 2000 ACM SIGMOD international conference on management of data*. pp. 117–128. Dallas, Texas. 2000.

[DQA04] Datta, A., Quarteroni, S., and Aberer, K.: Autonomous Gossiping: A self-organizing epidemic algorithm for selective information dissemination in mobile ad-hoc networks. In: *International Conference on Semantics of a Networked World (IC-SNW'04)*. Paris. June 2004.

[LW04] Lindemann, C. and Waldhorst, O. P.: Exploiting epidemic data dissemination for consistent lookup operations in mobile applications. *SIGMOBILE Mob. Comput. Commun. Rev.* 8(3):44–56. 2004.

[PS00] Papadopouli, M. and Schulzrinne, H.: Seven degrees of separation in mobile ad hoc networks. In: *Proceedings of IEEE Global Telecommunications Conference(GLOBECOM)*. San Fransisco, California. 2000.

[PS01] Papadopouli, M. and Schulzrinne, H.: Effects of power conservation, wireless coverage and cooperation on data dissemination among mobile devices. In: *ACM International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc)*. Long Beach, California. October 2001.

[WC02] Williams, B. and Camp, T.: Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks. In: *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing (MOBIHOC '02)*. pp. 194–205. Lausanne, Switzerland. June 9-11 2002.