On the Impact of Network Dynamics on a Discovery Protocol for Ad-Hoc Networks

Fei Liu, University of Twente, the Netherlands
Geert Heijenk, University of Twente, the Netherlands

ABSTRACT

A very promising approach to discovering services and context information in ad-hoc networks is based on the use of Attenuated Bloom filters. In this paper we analyze the impact of changes in the connectivity of an ad-hoc network on this approach. We evaluate the performance of the discovery protocol while nodes appear, disappear, and move, through analytical and simulative analysis. The analytical results are shown to be accurate when node density is high. We show that an almost linear relation exists between the density of the network and the number of update messages to be exchanged. Further, in case of nodes moving, the number of messages exchanged does not increase with the speed of movement. [Article copies are available for purchase from InfoSci-on-Demand.com]

Keywords: Ad-Hoc Networks; Mobility; Networking; Performance Evaluation; Wireless Technologies

INTRODUCTION

Context-aware ad-hoc networks adapt their behavior based on the context in which they operate. For this purpose, nodes use information from context sources. To discover these sources, a context discovery protocol is needed. Such a protocol disseminates information on context information that can be provided by nodes to nodes that might want to use the information. Ad-hoc networks are severely limited in resources, such as communication bandwidth, energy usage, and processing power. To save communication resources, we have proposed to perform context discovery using attenuated Bloom filters (ABFs) (Liu & Heijenk, 2007). We have proven that using ABFs our discovery protocol can provide discovery, while exchanging far less information than conventional approaches.

Another important feature of ad-hoc networks is dynamics in connectivity. In this paper, we present an investigation of the impact of network dynamics on our ABF-based discovery protocol through an analytical ap-
A DISCOVERY PROTOCOL FOR AD-HOC NETWORKS

Attenuated Bloom Filters (ABF)

Bloom filters (Bloom, 1970) have been proposed in the 1970s to represent a set of information in a simple and efficient way. They use \( b \) independent hash functions to code the information. The hash results are over a range \( \{1...w\} \), where \( w \) denotes the width of the filter. In the filter, which has a length of \( w \) bits, every bit is set to 0 by default. Only the bit positions associated with the hash results will be set to 1. The resulting Bloom filter can be used to query the existence of certain information. If all the bit positions related to the hash results of the queried information are 1 in the filter, the information exists with small chance of false positive.

Attenuated Bloom filters (ABFs) are layers of basic Bloom filters. We use ABFs to represent information regarding the presence of context sources on a hop-distance basis (Liu & Heijenk, 2007). The \( i \)th layer of an ABF (\( 0 \leq i < d - 1 \)) aggregates all information about context sources \( i \) hops away. The depth of the ABF, \( d \), also stands for the total propagation range of the information. Note that context sources reachable in \( i \) hops may also be reachable via longer paths. As a result, hash results at layer \( i \) will often be repeated in lower layer \( j \) (\( j > i \)).

Figure 1 exemplifies the context aggregation operation for a node with two neighbors. In this example, each node has an ABF with 8 bits width (\( w=8 \)) and a depth of 3 (\( d=3 \)). The node uses two hash functions (\( b=2 \)) to encode its local context sources “temperature” and “humidity” into \( \{2,8\} \) and \( \{2,5\} \) respectively. If we set the corresponding bit positions, we can obtain \( filter\_local \) as shown in Figure 1. When the node receives the incoming filters \( filter\_in[1,..] \) and \( filter\_in[2,..] \) from its neighbors, it shifts the received filters one layer down and discards the last layer. Thus, \( filter\_in[1,..]' \) and \( filter\_in[2,..]' \) are obtained. We perform a logical OR operation on each set of corresponding bits of \( filter\_local, filter\_in[1,..]' \), and
Related Content

Modeling Communication in Multi–Processor Systems–on–Chip Using Modular Connectors
[www.irma-international.org/article/modeling-communication-multi-processor-systems/42984/](www.irma-international.org/article/modeling-communication-multi-processor-systems/42984/)

Interference Suppression Capabilities of Smart Cognitive-Femto Networks (SCFN)
[www.irma-international.org/chapter/interference-suppression-capabilities-smart-cognitive/74423/](www.irma-international.org/chapter/interference-suppression-capabilities-smart-cognitive/74423/)

Towards Sustainable Development of Energy-Aware Systems
[www.irma-international.org/chapter/towards-sustainable-development-energy-aware/65597/](www.irma-international.org/chapter/towards-sustainable-development-energy-aware/65597/)

Cognitive Aeronautical Communication System
[www.irma-international.org/article/cognitive-aeronautical-communication-system/52982/](www.irma-international.org/article/cognitive-aeronautical-communication-system/52982/)

Predictive Analytics of Hyper-Connected Collaborative Network
[www.irma-international.org/article/predictive-analytics-of-hyper-connected-collaborative-network/216429/](www.irma-international.org/article/predictive-analytics-of-hyper-connected-collaborative-network/216429/)