Link Cache Validation Mechanism for Dynamic Source Routing (DSR) in Ad hoc networks

Dimitri Marandin, Jose Gonzalez
Chair for Telecommunications,
Technische Universität Dresden, Germany
Contact: marandin@ifn.et.tu-dresden.de

.

Abstract

An *ad hoc network* is a collection of mobile hosts with wireless network interfaces that may form a temporary network without the aid of any established infrastructure or centralized administration. In such environment, the nodes operate both as hosts as well as routers. Due to mobility, the topology of the network may change randomly, rapidly and unexpectedly. Because of these aspects and the fact that the resources are limited in mobile nodes, efficient routing in ad hoc networks is a crucial and challenging problem.

Dynamic Source Routing (DSR) is one of efficient on-demand routing protocols. The main idea of on-demand routing is to find and maintain *only needed routes*. The obvious advantage of discovering routes on-demand is to avoid overhead costs of maintaining routes that are not used. A route to some destination is tried to be discovered only when a sending node generates a data packet addressed to that node. In order to avoid the need for such a route discovery to be performed before each data packet is sent, such routing protocols must cache previously discovered routes. The entire route is included in each packet, so that spatial information about the network topology can be extracted from packets and used to form a cache.

In DSR, the route returned in each Route Reply that is received by the initiator of a Route Discovery represents a complete path (a sequence of links). By caching each of these paths separately, a path cache can be formed. Alternatively, a link cache could be created, in which each individual link in the routes returned in Route Reply packet is added to a unified graph data structure of node's current view of the network topology. A link cache can effectively utilize all of the potential information, but to find a route, a node must use a much more complex graph search algorithm.

Due to mobility, cached routes easily become stale. A mechanism to delete invalid links from the cache is necessary. Prior work in DSR used heuristics with ad hoc parameters to predict the lifetime of a link. However, because of topology changes, heuristics can not accurately predict timeouts. The goal of our work is to implement a mechanism that will adapt to topology changes, without using ad hoc parameters, which depend on the topology. In our approach all reachable nodes that have cached a broken link are notified proactively when a link failure occurs. In this case, the cache will be updated as soon as a link failure is detected, and not after timeout as in usual approach, so the usage of broken links and stale routes will be avoided.

To achieve this goal, it is necessary to keep which node has cached which link, in a distributed manner. To do this each node, forwarding either a route reply or a data packet, will store the links of the route and the neigboring nodes, which learnt these links from it. Using this local information, each node can determinate which neighbors have to be informed when a link failure occurs. In case of failure, the information about the failure will rapidly propagated through the network. Every node, that cached that link, will be aware of the failure and delete the invalid information from its cache.

Our approach shown better performance againist timeout-based invalidation algorithms. Especially good results have been obtained for delivery ratio and throughput.