Scalable Routing in Sensor Actuator Networks with Churn

Thomas Fuhrmann
IBDS Systemarchitektur, Universität Karlsruhe (TH), Germany
thomas.fuhrmann@ira.uka.de

Abstract

Routing in wireless networks is inherently difficult since their network topologies are typically unstructured and unstable. Therefore, many routing protocols for ad-hoc networks and sensor networks revert to flooding to acquire routes to previously unknown destinations. However, such an approach does not scale to large networks, especially when nodes need to communicate with many different destinations.

This paper advocates a novel approach, the scalable source routing (SSR) protocol. It combines overlay-like routing in a virtual network structure with source routing in the physical network structure. As a consequence, SSR can efficiently provide the routing semantics of a structured routing overlay, making it an efficient basis for the scalable implementation of fully decentralized applications.

In [5] it has been demonstrated that SSR can almost entirely avoid flooding, thus leading to a both memory and message efficient routing mechanism for large unstructured networks. This paper extends SSR to unstable networks, i. e. networks with churn where nodes frequently join and leave, the latter potentially ungracefully.

1. Introduction

The study of mobile ad-hoc networks and sensor networks focuses on two distinct areas: (a) scenarios of small network clouds of typically less than 100 nodes that communicate e. g. via IEEE 802.11 wireless links, and (b) scenarios of many tiny devices with limited resources that aggregate data towards larger gateways. We aim at a related but different scenario: large unstructured networks where the nodes communicate using the semantics of structured routing overlays. The following example shall motivate why we believe that such a scenario will become more and more important in the near future.

1.1. Application scenario

Our guiding example is a community of digital homes where technically inexperienced people unstructuredly deploy a growing number and variety of networked devices. These devices contain tiny processors interfacing to sensors and/or actuators. They could, for example, be part of a lighting or air conditioning system, or control household appliances, potentially together with other kinds of devices such as PDAs acting as user interface towards these embedded controllers.

We assume that many of these devices are connected wirelessly using different technologies, e. g. Zigbee, Bluetooth, IEEE 802.11, etc. But we also assume that at least some devices also connect to a wired network, e. g. via Ethernet, powerline, or a field bus. Multi-protocol devices can provide connectivity across these different link layer technologies. This will result in a potentially large network that spans not only single houses, but local communities or larger municipal or industrial complexes. Thus, the network topology is neither that of a typical ad-hoc network nor that of a typical infrastructure network.

Today, we already see the onset of such a scenario. More and more devices communicate wirelessly; and more and more buildings provide a local wireline infrastructure; but the wireless links typically operate on a point-to-point (or multipoint) basis, only: A PDA attaches to an access point; a remote control attaches to the TV set or the air-conditioning. Neither of them is capable of using one of the other devices as a relay.

Our vision is that all these devices form a real network where messages are routed potentially across several intermediate devices. With such an approach devices do not need to use their radio to reach their respective destination, but can rely on a more powerful link. Any device that has enough resources should be able to act as relay. This saves overall energy, reduces interference, and allows the creation of truly distributed applications. The latter may be simplified considerably by providing the semantics of a so-called *structured rout-*

1-4244-0626-9/06/$ 20.00 (c) 2006 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE SECON 2006 proceedings.

ing overlay in the network, as detailed in the following section.

1.2. The key-based routing semantic

Overlay networks, including peer-to-peer networks, have been quite successful recently. One powerful concept is the concept of a *structured routing overlay* providing the *key-based routing semantic* (KBR) [2]. *Distributed hash tables* (DHT) typically use KBR to provide distributed storage of data items whose keys are hashed to an identifier space that is split across the participating nodes. Typical examples are CAN [17], Chord [20], and Pastry [18]. Furthermore, structured routing overlays can be used for various other purposes as described e. g. with the Internet Indirection Infrastructure (i3) [19].

In a scenario like ours, the KBR semantics allows the easy construction of e. g. a rendezvous mechanism for service discovery and other look-up problems. Thereby, systems can be broken up into sensors and actuators whose coupling is not fixed, but that can be flexibly mediated by the underlying network. KBR can also act as a basis for data aggregation and multicast (as done with i3), and of course for a distributed storage system. Moreover, KBR is a generalization of unicast routing, so that a KBR system also provides unicast routing.

1.3. Contribution

Previous work of the author [5] has already described and evaluated the *scalable source routing* (SSR) protocol, an approach that provides scalable routing in large unstructured networks. However, so far, SSR has been limited to static networks. In this paper we extend SSR so that it remains stable in presence of node churn, i. e. in situations where nodes frequently join and leave the network, the latter potentially ungracefully.

Churn has been studied intensively with peer-to-peer networks. But it is also a typical phenomenon for wireless networks such as the scenario described above. Even when nodes have limited mobility, the link quality can quickly and unexpectedly change. Moreover, individual nodes might suddenly fail or be powered down. The mechanisms described in this paper serve to maintain SSR's routing service intact in spite of churn. Nevertheless, the design takes in to account that the protocol must be suitable for large networks of small embedded devices that can hold only very little state.

This paper is structured as follows: Section 2 gives a brief overview of related work. Section 3 describes the SSR protocol: Sections 3.1 – 3.5 review SSR for static networks, introducing a few improved protocol features, especially to speed up node joins (section 3.4). Sections 3.6 and 3.7 present new protocol mechanisms which ensure SSR's stability in case of node churn. Section 4 presents simulation results from an 8000 node network with varying churn; and section 5 concludes with an outlook to future work.

2. Related work

Ad-hoc routing has been studied for quite a while now [8]. A well-known protocol is *ad-hoc on-demand distance vector routing* (AODV) [15][1]. AODV floods the network with route request messages in order to enable the network to route to previously unknown destinations. The so acquired state is stored as routing table entries in the network, i. e. nodes know the direction to active sources and destinations.

Dynamic source routing (DSR) [11], too, uses flooding, but in a way that builds a source route spanning the entire way from the source to the destination. Both, source and destination node cache the source route in a route cache. But unlike AODV, all the other nodes do not need to maintain state. Several additions [13] like salvaging, gratuitous route repair, and promiscuous listening improve DSR's performance in the presence of churn and node mobility. Further optimizations [9] concern the way routes are kept in the caches.

SSR shares many of these ideas except for the principal differences that routes are not discovered by flooding route requests, but by appending routes stepwise while approaching the destination in the virtual ring. Moreover, unlike DSR, SSR's route cache is limited in order to meet the requirement of a constant memory footprint.

Geographic routing [12][14] relies on the physical position of the nodes for routing. Thereby, the nodes do not need to acquire routes but can calculate the respective next hop node from the positions of their neighbors and the position of the destination node. However, to work properly, geographic routing needs to employ perimeter routing which can only be used when the nodes are spread in a 2-dimensional space. Moreover, geographic routing requires the nodes to somehow determine and communicate their physical positions. Both limitations are not present with SSR.

Unlike SSR and the previously mentioned protocols, hierarchical routing mechanisms group the nodes into clusters or subsets. Landmark routing [6] selects one node in each subnet to act as landmark for the subnet. Only the routes to the landmarks are distributed globally. Messages are first routed towards one of these landmarks, and then distributed locally by another routing mechanism, e. g. flooding. Moreover, in order to be

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE SECON 2006 proceedings.

able to send a message in the first place, a source needs to know a landmark in the vicinity of the respective destination. To this end, the nodes must be organized into subnets and the routes to the subnets' landmarks need to be distributed globally. This is not true for SSR: node addresses do not need to adhere to a subnet structure, source routes can be bootstrapped in a scalable manner, and messages can be routed to any destination without a final flooding step.

Routing in sensor networks is mainly concerned with data aggregation and dissemination. Accordingly, flooding and gossiping are important mechanisms [10][3]. SSR, on the other hand, aims at providing KBR independent of a particular application. Although SSR can efficiently provide data aggregation and dissemination along the routing tree implicitly created by the key-based routing [19][7], it also provides unicast routing for arbitrary pairs of nodes. This is important when an application needs to link specific sensors and actuators, e. g. because they jointly form a spatially distributed apparatus.

The combination of underlay and overlay approaches has been briefly discussed in the literature: Pucha et al [16] advocate using DSR to populate Pastry's routing table. But since the authors do not detail their approach, we cannot compare it to SSR. We can only speculate that DSR's flooding might severely affect the performance of this straightforward approach. Moreover, SSR contains elaborate route cache operations absent in Pucha's proposal.

In order to achieve scalability, several authors have proposed to use landmark routing. MADPastry [21] combines landmarks to obtain virtual addresses, Pastry to store and use these virtual addresses, and AODV to build up the according routing state for the actual physical network. [4] pursues a similar approach, but uses DSR to obtain the physical routes. In both cases, virtual addresses are used to create a hierarchical structure on top of the network that helps reducing the required amount of state. In contrast, SSR deeply intertwines virtual and physical structure to obtain a scalable routing not from a hierarchy but from SSR's emergent effects.

3. Scalable Source Routing

3.1. Overview

Scalable source routing (SSR) is a self-organizing routing protocol that focuses on large, unstructured networks where nodes have limited resources. Its design was guided by the following three assumptions that reflect the constraints and conditions of the scenario described above:

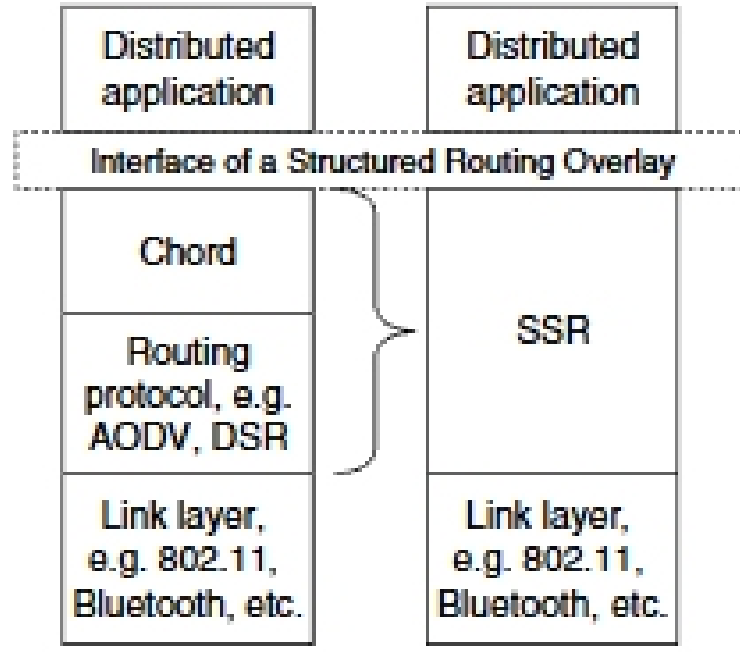

Figure 1. SSR's cross-layer approach

1. Nodes can afford only little memory to hold routing state. The memory is assigned statically, i. e. there is no way to acquire additional memory.

2. Nodes communicate with many different destinations using the structured overlay routing semantic.

3. The network consists of heterogeneous nodes that have largely differing communication capabilities. Most nodes have only very few links, but a few nodes are very well connected, i. e. the network has small world topology.

SSR was inspired by the Chord overlay, and it shares some basic ideas with DSR. Like DSR, SSR builds up source routes and maintains a cache holding the 'most useful' source routes. Unlike DSR, it does not need to flood the network to discover unknown source routes, but uses an additional routing rule to forward the packet. This routing rule corresponds to the Chord overlay routing rule. To this end, SSR supplements the physical network topology with an additional virtual ring structure. This combination of DSR-like source routes with a Chord-like virtual ring makes SSR scalable and provides the routing semantics of a structured routing overlay.

However, unlike Chord, SSR is a network layer protocol, i. e. it does not require another routing system underneath (cf. fig 1). This cross-layer approach reduces both, the implementation complexity and the resource requirements compared to the case where an overlay routing protocol like Chord would operate on top of an ad-hoc routing protocol like DSR.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE SECON 2006 proceedings.