The Use of Scalable Source Routing for Networked Sensors

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

Abstract

Routing in sensor networks is a demanding task. Neither do the nodes have sufficient processing and memory resources to perform sophisticated routing algorithms like e.g. Dijkstra; nor do they have sufficient bandwidth and energy to revert to simple routing approaches like e.g. flooding. Moreover, the network topologies lack well-crafted structures that might simplify routing e.g. by the introduction of hierarchies.

In this paper, we briefly present a novel routing algorithm, scalable source routing *(SSR), which is capable of memory and message efficient routing in networks with 'random topology'. This algorithm enables sensor networks to use recent peer-to-peer mechanisms from the field of overlay networks, like e.g. distributed hash tables and indirection infrastructures. Unlike other proposals along that direction, SSR integrates all necessary routing tasks into one simple, highly efficient routing protocol.*

Simulations demonstrate that in a small-world network with more than 100 000 nodes, SSR requires each node to only store routing data for 255 other nodes to establish routes between arbitrary pairs of nodes. These routes are on average only about 20-30% longer than the globally optimal path between these nodes.

1. Introduction

Routing is a well-studied topic in networking. It is the task of picking a sequence of nodes between the source that sends a packet into the network and the packet's destination. Optimally, this sequence should denote a shortest path, but a (slight) deterioration is tolerable for most applications. (Note that in the Internet the service providers' policies often lead to suboptimal paths.)

Many routing algorithms are known and widely applied. Each has its particular strengths and weaknesses. In infrastructure networks, Dijkstra and Bellman-Ford yield globally optimal paths at the cost of having each node store a routing table for the entire network. (Only in networks with hierarchical structure, like e.g. the Internet, the routing table can be aggregated efficiently.) In ad-hoc networks, *flooding* yields globally optimal paths at the cost of creating a large message overhead. More message efficient approaches like e.g. *geographic routing* are typically limited to 2-dimensional unit disk like scenarios.

In this paper, we advocate a novel routing approach *scalable source routing* (SSR). It is motivated by the observation that sensor networks create the need for message and memory efficient routing in large networks that are neither purely unit-disk nor hierarchical. Such networks emerge e.g. when both, wired and wireless sensors and actuators are deployed in an uncoordinated manner. A guiding example is the community of digital homes, where laypersons install equipment that is enhanced with communication features. Many such devices will incorporate sensors whose data might be beneficial for other devices, too.

> Example 1: Blinds, marquees, etc. must be retracted when a storm approaches. Today, each house has its own sensors. With SSR, all the sensor readings from the neighborhood can be combined to yield more reliability for the entire community.

Note, that in such an environment, typically, wireless links are mixed with wired links. Individual sensors might use (different) RF-technologies, but there will also be e.g. wired wall-mounted switches, power lines to drive the blinds' motors, etc. These wires might also be used for communication. Moreover, using wires (where applicable) to bridge larger distances reduces both, the energy required by the sensors and the electromagnetic interference problems. Thereby, communication relations can efficiently be stretched across the whole network.

> Example 2: Sensors in the greenhouse at the other end of the garden need not directly report to the associated display in the living

Proc. 2nd IEEE Workshop on Embedded Networked Sensors, Sydney, Australia 2005. pp. 163-165

> room, kitchen, etc. Any infrastructure (both wired and wireless) can relay the data. If required, the data can be redirected to other locations on demand, e.g. the person who takes care of the greenhouse when the owners are on vacation.

One key design property of SSR is the separation of routing from data aggregation and dissemination. SSR provides routing between arbitrary nodes within the network. Mechanisms from the field of proximity aware routing overlays build there upon to provide *efficient* aggregation and dissemination of data. (Note that we do not discuss security and privacy here. Both will have to be provided by additional means. They are, among other aspects of SSR, subject of ongoing work in our group.)

2. Scalable source routing

To overcome the limitations of the classical infrastructure and ad-hoc network routing algorithms, SSR employs ideas from the field of peer-to-peer overlay networks. In essence, SSR is built upon the following principle:

> Instead of equipping each node with a full view of the network, nodes store routing data (source routes) for a limited part of the network, only. The view of each node is determined by its logical address, not its physical location in the network, spreading the knowledge redundantly across the entire network.

More detailedly, SSR is based on the following core ideas:

(i) All nodes bear logical addresses, independent of the physical network topology. Addresses can e.g. be manufacturer assigned, hashes of cryptographic keys, or random numbers. Since the addresses need not relate to any property of the network, SSR works in all network topologies.

(ii) Protocolwise, the nodes' addresses are viewed as to form a sparsely populated ring. Upon booting, each node acquires a source route to its successor in this virtual ring. It can be proven that this can be done very message efficiently. (See [2] and references therein.) Since inconsistencies are automatically detected (namely by two nodes claiming a node to be their successor), nodes can join the system at any time.

(iii) Messages are routed along the virtual ring using source routes from the caches maintained by each node. Here, a source route that spans a shorter physical distance is preferred over a source route that spans a larger virtual distance. (Note that an n-hop source route is viewed as n source routes, i. e. all intermediate hops in the cache are considered, too.) Thus, already within the first few hops, a message will advance far within the virtual ring. Moreover, this leads to nodes specializing to cache source routes for the ring segment of nodes with close addresses.

(iv) When forwarding a message by appending a source route, potential loops are eliminated. Here, special care is taken to always be able to provide a source route back to the node that appended the most recent route. This is necessary to update the route cache in case a subsequent link is found to be broken and hence allows the network to cope with (moderate) node churn.

This is only a very brief protocol description. All details are deferred to an accompanying technical report [1]. Nevertheless, we want to mention that the full routing protocol, including loop elimination, header compression, and provisioning of redundant links, can be implemented with a low memory footprint, both with respect code and state.

3. Simulation results

We implemented SSR in C++ and simulated random networks with several topologies and sizes. In these simulations we specifically focused on two aspects: (1) How quick and message efficient does the system converge into its globally consistent state? (2) What is the achieved path length of the source routes as compared to shortest paths?

Summarizing the results from [2], we can state that both small world topologies and unit disk graphs are handled well by SSR. Routing stretches (=ratio of the length of SSR's paths and shortest paths) are between 10% and 30%. (Figure 1 shows the CDFs for the achieved routing stretches.) With small-world networks a route cache of only 255 nodes is sufficient to handle networks of more than 100 000 nodes. Moreover, the stretch rises only slightly with increasing network size. With unit disks graphs the route cache needs to be larger (511 nodes) since routes are generally much longer there.

In a static network, SSR converges after the exchange of about 10-20 messages into its globally consistent state. In the 8000 node small-world network each node sent on average 5.5 messages to establish its successor relation. The unit disk scenario additionally requires some nodes to flood the network to achieve global consistency. (Statistically, only 3.7 nodes flood the network in the 8000 nodes scenario.) This is a significantly lower overhead than with classical ad-hoc routing protocols.

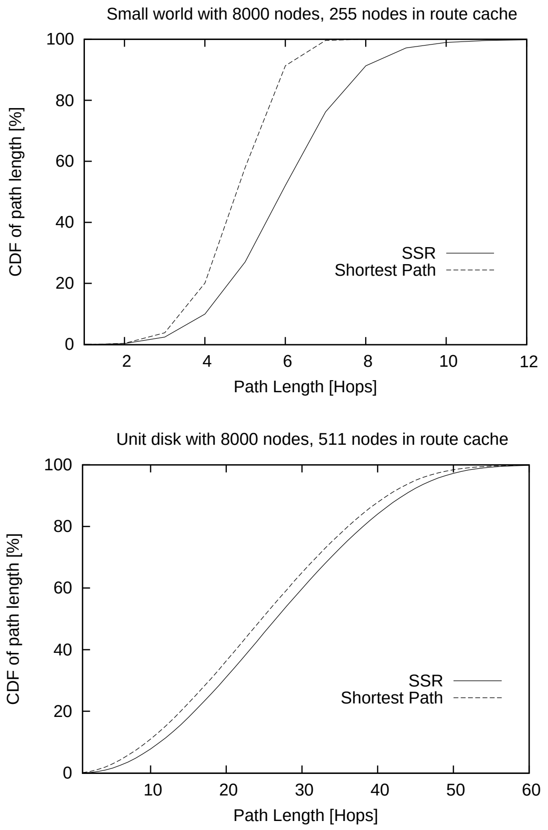

Figure 1. Achieved routing stretches

The loop elimination process cuts and joins paths at 'prominent' nodes, i. e. nodes that are contained in many paths and are thus known to many other nodes in the network. One might suspect that this leads to a self-amplifying effect that could eventually overload these nodes. We verified that this is *not* the case. Node frequencies in the paths are stable and correspond to the nodes' respective degrees (=number of physical links).

4. Data aggregation and look-up

So far, we have argued that SSR is a routing protocol that is well suited for networks where classical routing protocols are not applicable, either because they would require each node to store a large routing table, or because nodes would frequently flood the network. Clearly, SSR's memory and message efficiency renders it especially suitable for sensor networks where both node resource and bisection bandwidth are scarce. But beyond these advantages, SSR's particular routing approach makes it even more suited for sensor networks.

As described above, SSR is based on a virtual ring of logical addresses. These addresses need not be correlated to the physical network topology. This simplifies look-up problems that are difficult to solve in fully distributed systems. Moreover, even messages that are destined to a non-existing address are routed in a well-defined way: Namely, they will end up on the node with the closest address. As is known from approaches like the Internet Indirection Infrastructure (i3), this allows the easy construction of mapping services.

Taking up the storm detection example from above, sensor readings for a particular class of devices (e.g. wind speed) can be sent to the hash of a unique ID (e.g. vendor key). Thereby, a tree is formed, along which the values can be aggregated and disseminated backwards to interested actuators. With high probability aggregation will reflect physical network proximity. The realization of this idea is among the ongoing work in our group.

5. Conclusion and Outlook

In this paper, we have briefly described a novel routing approach, *scalable source routing* that is able to provide full connectivity in large networks of limited capability nodes. Unlike classical infrastructure routing protocols, SSR neither assumes the network to bear a hierarchical structure nor does it require the nodes to store large routing tables. Unlike classical ad-hoc routing protocols, SSR almost entirely avoids flooding. Nevertheless, SSR can efficiently benefit from short-cuts through e.g. a wired network infrastructure.

Due to the very tight space limitation for this paper, we could only try to make the case for the use of SSR for networked sensors and actuators. We kindly refer the interested reader to two technical reports that fill the gaps in the argumentation made here: [1] describes the SSR protocol in detail. [2] reports on extensive simulations that evaluate SSR in various scenarios. The latter also reviews related approaches, both from classical routing and from recent publications in the area of peer-to-peer computing and sensor networks. It contains also the references to the literature our work builds upon.

References

[1] C. Cramer, T. Fuhrmann, and K. Kutzner, "Scalable source routing - protocol specification, version 1.0", Tech. Rep. 2005-4, Faculty of Informatics, University of Karlsruhe, Germany, 2005.

[2] T. Fuhrmann, "Scalable source routing – protocol evaluation", Tech. Rep. 2005-13, Faculty for Informatics, University of Karlsruhe, 2005.