Exploiting Reinforcement Learning for Multiple Sink Routing in WSNs

Anna Egorova-Förster
University of Lugano, Switzerland
anna.egorova.foerster@lu.unisi.ch

Amy L. Murphy
FBK-IRST, Italy
murphy@itc.it

Abstract

Efficiently moving sensor data from its collection to use points is both the fundamental and the most difficult challenge in wireless sensor networks, as any data movement incurs cost. In this work, we focus on routing data to multiple, possibly mobile sinks. To deal with the dynamics of the environment arising from mobility and failures, we choose a reinforcement learning approach where neighboring nodes exchange small amounts of information allowing them to learn the next, best hop to reach all sinks. Preliminary evaluation demonstrates that our technique results in low cost routes with low overhead for the learning process.

1 Introduction

[1]Most wireless sensor network (WSN) routing algorithms efficiently collect data from multiple sources at a single sink. Our work reverses this scenario, focusing on delivering data from one (or more) sources to multiple, possibly mobile sinks *within* the network. Such a scenario arises when rescue workers with portable devices use information from their sensor-enhanced environment to decide where to go and what actions to take.

Technically, the problem we face is similar to building a multicast tree, however previous solutions for the similar MANET environment require geographical information [5] which is not always available in WSNs or incur large communication overheads to construct a multicast tree using additional control packets [7]. In contrast, our approach uses only localized, easily obtainable information and uses the data packets themselves to identify the best routes, lowering the overhead.

To cope with the dynamics inherent in the system arising from failures and mobility, we choose a *reinforcement learning* solution in which each node incrementally learns its best next-hop on the route to all destinations. Learning occurs as neighbors share knowledge by exploiting the inherent broadcast nature of wireless communication. We provide details of this feedback mechanism in Section 2.

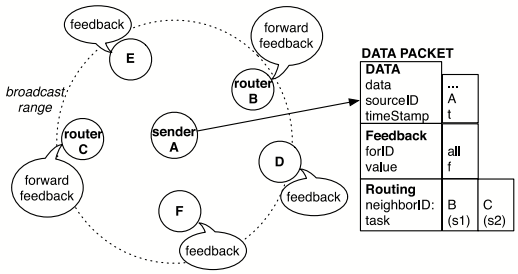


Figure 1. Sample feedback scenario. Node A sends a packet for routing to nodes B and C, and feedback to all neighbors.

Related efforts have applied reinforcement learning to routing in WSNs [1, 2]. In comparison, our work applies learning to the multiple sink problem and defines a precise protocol, outlined in Section 3. Following this, in Section 4, we numerically demonstrate the potential benefits of our approach and discuss its ability to cope with changes in the network due to failure and mobility.

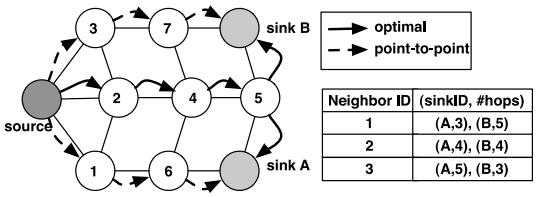
2 The FR FRAMEWORK

Our reinforcement learning approach requires nodes to receive data from neighbors in order to learn. This data is usually small, such as residual node energy, available routes to sinks, route costs to specific sinks, application role assigned to the node, link quality, etc.

To facilitate easy, low-cost exchange of information, we devised the FR FRAMEWORK, which takes advantage of the broadcast nature of the wireless medium to exchange small pieces of information within a neighborhood. In a nutshell, it piggybacks information on all data packets, allowing all nodes in range to receive it.

In a broadcast environment, every node must process at least the beginning of every packet to know if it is an intended recipient. With the FR FRAMEWORK, while the

[1]The work described in this paper is supported by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under grant number 5005-67322.



Neighbor ID	(sinkID, #hops)
1	(A,3), (B,5)
2	(A,4), (B,4)
3	(A,5), (B,3)

Figure 2. Sample network showing lower costs by sharing links. The Neighbor Table of the source is also provided.

intended data recipients process the data, other nodes extract the feedback information from the message body. To this end, we define a simple packet format, shown in use in Figure 1, which specifies both the packet destination(s) (in the *Routing* section), as well as feedback information, itself destined to some neighbors. In theory feedback can take any form, but in general it must be size limited, to avoid unnecessarily increasing packet size. An implementation of FR FRAMEWORK for the Omnet++ network simulator is available at WWW.INF.UNISI.CH/MICS.

3 FROMS

In this section we show one instantiation of our FR FRAMEWORK, FROMS (Feedback Routing for Optimizing Multiple Sinks), where shared routes to multiple sinks are discovered through a simple learning mechanism. The benefits of finding shared routes to multiple sinks is intuitively shown in Figure 2. Considering broadcast transmission costs, the best route from the source to both sinks goes through nodes 2, 4, and 5 with a total cost of 4. Note that in this case and in general, the resulting shared route does not overlap with any of the single best source-to-sink routes.

Next we provide the intuition of how FROMS uses Q-learning to identify the best routes to multiple sinks. Additional detail is available in [3].

Overview In Q-learning, each *learning agent* takes actions, receives a reward, updates local information (Q-values) with input from the environment, and repeats the process. Ideally the local information informs the agent about the goodness of the available actions, allowing it to learn the best actions. In FROMS, each node is a learning agent, the available actions are different routing options at each node to reach all sinks in the network, and the Q-values are the estimated route costs. Rewards are exchanged among neighbors (the environment) using our FR FRAMEWORK and include the best Q-value known at the sending node. These rewards are used upon receipt to update local Q-values. The following gives additional detail on the individual parts of the protocol.

Routing options and Q-values To track the available routing options to all known sinks, each node maintains a special data structure, which we call the Path Sharing Tree (PST). Continuing our example from Figure 2, the source has different routing options to reach both sinks: using only neighbor 1, only neighbor 2, only neighbor 3, or using different neighbors for different sinks. We use $\{N_1(A, B)\}$ as shorthand for the first option. Q-values are assigned to each routing option and represent its current cost estimation.

Q-values must be initialized. This could be done randomly, but in our scenario, we can use information from the network to set *smarter* initial values that will speed up the learning process and minimize network costs. Specifically, we assume that when a sink announces its interest in receiving data, each node caches the number of hops between itself and that sink. This hop count is used as an initial upper bound cost estimate of the shared route. In our example, for option $\{N_1(A, B)\}$ we estimate a cost of 7: 3 hops to sink A, 5 hops to sink B and minus 1 because of the first hop broadcast.

Rewards When packets begin to flow through a node it selects among its available routing options, *exploring* them. When sending the packet to the next hop, it includes also its *reward* as feedback for the previous hop. In our case the reward is the lowest hop count to the destined sinks. Thus, the actual routing costs propagate through the network while exploiting only the neighborhood communication of our FR FRAMEWORK. Eventually the best routes will be identified, and the Q-values no longer change. This is referred to as *convergence*. In our example, after convergence, the Q-value at the source for $\{N_1(A, B)\}$ is 5, the actual path cost.

Exploration/exploitation An important part of the Q-learning approach is its exploration strategy. This refers to how the routing options are chosen in each round. Using only the best available ones (exploitation) may lead to a routing solution which is a local minimum. Thus, some exploration of non-optimal routes is needed. Section 4 presents an evaluation of FROMS with two different exploration strategies: GREEDYEXPLORE, a pure exploitation strategy and UNIFORMEXPLORE, which selects uniformly between all available routes, giving preference to less-explored routes.

4 Discussion

Next we present a sample of our numerical results from simulating FROMS and a short discussion of its most important properties.

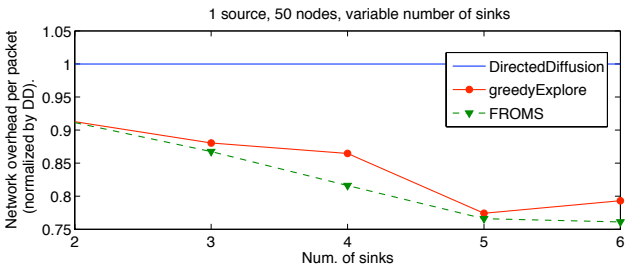


Figure 3. Routing cost for FROMS and GREEDYEXPLORE w.r.t. DIRECTEDDIFFUSION.

Simulation results We have extensively simulated and evaluated FROMS with Omnet++ and its Mobility Framework (WWW.OMNETPP.ORG). Figure 3 demonstrates the behavior of FROMS, compared to a one-phase-pull Directed Diffusion[6] implementation. The evaluation presents averages over 50 random, connected 50-node topologies on a field of 1500x1500 m, each with 5 random seeds. This evens out the effects of the stochastic nature of FROMS. Data is sent every second from the sources to the sinks. The nodes have a maximum transmission radius of 400 m and use the standard simulation models (physical layer with a bit-error function, non-persistent CSMA, Mica-2 battery model). The figure clearly shows a gain in network cost per packet between 10 and 25% for varying numbers of sinks. The additional gain from exploration is also evident, compared to GREEDYEXPLORE.

Recovery after failure Keeping additional routes in the PST not only allows us to explore many routing options to find the best one, but also gives us back-up routes in case of node failure. When a node detects a neighbor failure, it can easily switch to the next best available routes. If the total cost changes, this will be disseminated in the relevant part of the network through feedback at no additional cost.

Sink Mobility Sink mobility introduces another degree of changes to the network topology that a routing protocol must adapt to. From the perspective of FROMS, sink mobility is a two-step process: link failure and insertion. The first step, link failure, is basically the same as node failure and consequently can be directly handled by FROMS recovery. The second step, link insertion, requires the sink to broadcast its announcement at regular intervals to its *one-hop neighbors*, ensuring that the new links are detected. Both steps trigger route cost updates, which are quickly disseminated through rewards to the relevant parts of the network.

5 Conclusion and future work

This paper presents both FROMS, our reinforcement learning approach for multiple sinks routing in WSNs and our FR FRAMEWORK for exchanging neighborhood information at minimal cost. Our evaluation clearly shows that the additional expense of learning, combined with the negligible overhead to piggyback reward information significantly improves network lifetime. The observation that FROMS innately supports node failure and sink mobility further increases its applicability.

Our immediate plans include further study of FROMS in various mobility and failure environments. We also intend to develop and deploy a real implementation to more accurately assess performance outside the inherent boundaries available in simulation. Finally, we are currently working on application of FROMS to support non-uniform data dissemination and clustering [4]. It assumes sinks require different aggregation ratios from different parts of the network and thus requires the network to be partitioned in a smart way into clusters of varying sizes. We intend to use a learning approach to identify the clusters and the cluster heads and FROMS to route the data from the cluster heads to multiple, mobile sinks.

References

[1] P. Beyens, M. Peeters, K. Steenhaut, and A. Nowe. Routing with Compression in Wireless Sensor Networks: A Q-Learning approach. In *Proceedings of the 5th European Workshop on Adaptive Agents and Multi-Agent Systems (AAMAS)*, 2005.

[2] J. A. Boyan and M. L. Littman. Packet routing in dynamically changing networks: A reinforcement learning approach. *Advances in Neural Information Processing Systems*, 6, 1994.

[3] A. Egorova-Förster and A. L. Murphy. A Feedback Enhanced Learning Approach for Routing in WSN. In *Proceedings of the 4th Workshop on Mobile Ad-Hoc Networks (WMAN)*, Bern, Switzerland, 2007. Springer-Verlag.

[4] A. Egorova-Förster and A. L. Murphy. Exploring Non Uniform Quality of Service for Extending WSN Lifetime. In *Proceedings of the 3rd International Workshop on Sensor Networks and Systems for Pervasive Computing (PerSens)*, White Plains, NY, USA, 2007. IEEE Computer Society.

[5] J. A. Sanchez, P. M. Ruiz, and I. Stojmnenovic. GMR: Geographic multicast routing for wireless sensor networks. In *Proceedings of the 3rd Annual IEEE Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, volume 1, pages 20–29, 2006.

[6] F. Silva, J. Heidemann, R. Govindan, and D. Estrin. *Frontiers in Distributed Sensor Networks*, chapter Directed Diffusion. CRC Press, Inc., 2003.

[7] R. Sun, S. Tatsumi, and G. Zhao. Q-map: A novel multicast routing method in wireless ad hoc networks with multiagent reinforcement learning. In *Proceedings of the 2002 IEEE Conference on Computers, Communications, Control and Power Engineering (TENCON)*, volume 1, pages 667–670 vol.1, 2002.