Virtual Localization for Robust Geographic Routing in Wireless Sensor Networks

Tony Grubman[1], Y. Ahmet Şekercioğlu[12⋆], and Nick Moore[1]

[1]Wireless Sensor and Robot Networks Laboratory, Monash University, Melbourne, Australia
[2]Université de Technologie de Compiègne, Compiègne, France

Abstract. Geographic routing protocols are well suited to wireless sensor networks because of their modest resource requirements. A major limiting factor in their implementation is the requirement of location information. The *virtual localization* algorithm provides the functionality of geographic routing without any knowledge of node locations by constructing a virtual coordinate system. It differs from similar algorithms by improving efficiency – greedy routing performs significantly better over virtual locations than over physical locations. The algorithm was tested and evaluated in a real network environment.

Keywords: wireless sensor networks, geographic routing, localization

1 Introduction

Efficiency of routing protocols is very important in wireless sensor networks [2], where nodes are cheap, resource-limited devices, and power consumption and use of the constrained wireless channel are key issues. The most *scalable* routing scheme is geographic routing [13], which requires the use of local information only (1-hop neighbourhood).

The simplest geographic routing protocol is greedy routing [13], where routing decisions are made to locally optimise the *progress* of a packet (usually measured as the distance to the destination). This generally finds efficient paths, especially in dense, uniform networks. Packet delivery, however, is not guaranteed, as packets can get stuck in local minima (called *voids*). The success rate of greedy routing is heavily dependent on the network's *topology* and *geometry*.

The main drawback of geographic routing is that it requires the knowledge of node locations. The most common solution is to equip each node with a GPS receiver, but this adds to the cost and power consumption of the nodes. Also, GPS signals may not always be available. Alternatively, some nodes may be *anchored*, with known locations, while others run a localization algorithm to find their coordinates relative to the anchored nodes. A comprehensive analysis of three such

⋆This work has been partially supported by the Labex MS2T, which is funded by the French Government, through the program "Investments for the Future", managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02).

algorithms (APS [8], Robust positioning [11], and N-hop multilateration [12]) is provided in [6]. Another localization algorithm is LASM [4].

For a completely self-organising wireless mesh network, there should be no requirement for any nodes to know their physical location. To achieve this, there are algorithms that construct *virtual* locations purely for routing purposes. These algorithms attempt to reproduce the functionality of geographic routing without using location information. This was first done in [9], where the algorithm relies on finding 'perimeter' nodes on the edges of the network. These nodes then exchange information to determine their virtual locations, after which they become anchor nodes (the other nodes perform a localization algorithm). The first two stages of the algorithm require many packets to be *flooded* through the network, and the resource requirements at the perimeter nodes are linear with respect to the network size.

In [5], a more scalable approach is used, where distances to some fixed anchors are used as the virtual coordinates directly (the anchors do not require physical locations). VCap [3] adopts a similar technique for constructing coordinates, but also defines a method to determine distances (in hops) to anchors. This involves packet flooding, but only to choose the anchors; the anchors do not need to flood messages to all other anchors. Discrete Ricci flows are used in [10] to construct virtual coordinates from a triangular mesh (which can be created without location information). The locations generated provide guaranteed delivery for greedy routing.

While the operation of these algorithms is more scalable, the performance of the routing algorithm itself (in terms of *reachability* and *path length*) is not considered in detail when constructing the virtual coordinate systems. The performance of greedy routing in [5] and [3] is comparable to (and sometimes worse than) using the physical coordinates. In [10], the reachability is always 100%, but the average path length is considerably higher than the case with physical coordinates. Thus all of these methods necessarily sacrifice performance to provide geographic routing capabilities to networks without location information.

Our virtual localization algorithm is explained in Section 2, an overview of the test network, we set up in Monash Wireless Sensor and Robot Networks Laboratory (WSRNLab) [14], for collecting the experimental data can be found in Section 3, and the results of the experiments are presented in Section 4. Finally, in Section 5, we offer our concluding remarks.

2 Virtual Localization

The virtual localization algorithm [7] constructs virtual coordinates using only local connectivity information (topology). Each node stores the virtual locations of its 2-hop neighbourhood and uses this information to calculate its own coordinates. Consistency is achieved with periodic broadcast packets containing the locations of the sender's 1-neighbours.

Nodes determine an optimal location to "place" themselves in by minimising an energy function. This corresponds to virtual forces acting on the node. The forces are based on a simple model: nodes are attracted to their 1-neighbours with a spring-like force, and experience a repulsive electrostatic-like force from their 2-neighbours. The energy can be minimised using any optimisation technique, but as the nodes are assumed to have limited computational capabilities, the stochastic hill climbing method is chosen for its simplicity. Fig. 1 summarises the operation of the algorithm.

As updated locations of neighbours are received, the energy function (and hence the optimum virtual location) changes. In fact, the energy minimisation algorithm can be continuously iterating as the neighbours' locations are being updated. This is especially useful in mobile ad-hoc networks, where the network topology changes constantly.

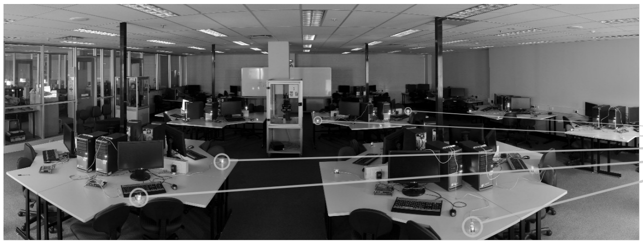
This algorithm is arbitrarily scalable because it uses only local information. All storage, computational, and network overhead requirements depend only on the node degree (i.e. network density), and not on the size of the network.

Require: Virtual locations of 1-neighbours, N
and 2-neighbours, M
Ensure: Own virtual location, ℓ
Parameters: Constants k_a, k_r, N_ITERATIONS

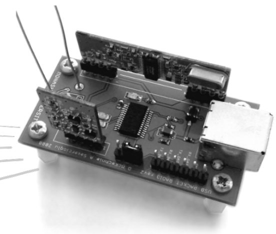
```
function ENERGY(a)                                  ▷ Calculates energy at location a
    U ← 0
    for b ∈ N do                                    ▷ Energy from 1-neighbours
        U ← U + k_a ||a − b||^2
    end for
    for b ∈ M do                                    ▷ Energy from 2-neighbours
        U ← U + k_r / (1 + ||a − b||)
    end for
    return U
end function

procedure LOCALIZATION                              ▷ Finds optimal location
    ℓ ← RANDOM                                      ▷ Initialise location
    for i ← 1 to N_ITERATIONS do
        t ← ℓ + RANDOM                              ▷ Perturb location
        if ENERGY(t) < ENERGY(ℓ) then
            ℓ ← t                                   ▷ Update location
        end if
    end for
    return ℓ
end procedure
```

Fig. 1. Virtual Localization Algorithm

3 Packet Radio Network



(a) Laboratory network testbed



(b) Packet radio module

Fig. 2. Monash WSRNLab's [14] experimental wireless mesh network.

A wireless mesh network testbed (Fig. 2(a)) was created using 33 packet radio modules (Fig. 2(b)). These modules are cheap devices operating in the 433 MHz ISM band, with a serial connection (over USB) to a computer. The network was set up in a computer lab, with each module controlled by a desktop computer. The lab contains many obstacles between the nodes, including chairs, tables and other computers.

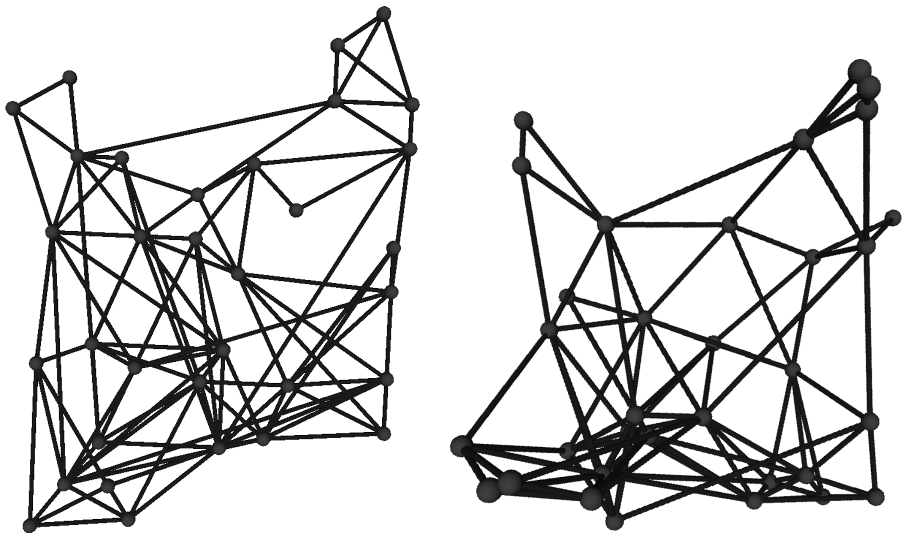
The virtual localization algorithm was implemented in Python, along with a basic wireless medium access control at the data link layer (ALOHA [1]). The program was run on the lab computers for each node, and the connectivity and location information were recorded and analysed.

4 Results

4.1 Virtual Locations

Virtual localization can generate coordinates in almost any metric space, but three dimensional Euclidean space was chosen. Even though the actual geometry of the network is planar (the nodes were placed at the same height), the extra dimension allows more complex *topologies* to be represented. Figures 3(a) and 3(b) show a typical topology of the lab network, and the corresponding virtual configuration achieved by the algorithm. The virtual locations vaguely resemble the actual locations, but the denser parts of the network tend to spread out more, as the algorithm cannot distinguish between 'long' and 'short' links.

The network topology in Figure 3(a) would not usually be considered when conducting simulations. This is because simulations frequently use the unit-disk graph (UDG) model, or some variant of it (usually quasi-UDG). In actual wireless networks such as the one in Figure 2(a), slight differences between different nodes (such as antenna length/transceiver sensitivity) have a dramatic impact on the quality of links between nodes. Some very long links are stable and reliable,



(a) Physical locations (b) Virtual locations

Fig. 3. Network topology and geometries

while some shorter links are very noisy and cannot be used reliably. Obstacles and the environment the network operates in also affect the topology significantly. No current simulation tools can accurately model such complex conditions, so a realistic evaluation of the algorithms can only be obtained on real networks.

4.2 Performance

The performance of the algorithm was assessed using the *reachability* and *path length* metrics. These were calculated at each time step by considering each pair of nodes (total of $33 \times 32 = 1056$) and applying the greedy routing algorithm. Packets are dropped when a void is encountered. The statistics were calculated using the actual (physical) locations and the virtual locations, and were compared to the optimal solution (packet flooding for reachability and shortest routes for path length). Figs. 4 and 5 show that using the virtual locations significantly improves the performance of greedy routing. This is because the virtual locations more accurately describe the topology of the network than the actual locations, thus reducing the number of voids.

The topology of the network was observed to change dramatically over the duration of each one hour run, with some links being made and others broken at seemingly random times. This is likely due to the probabilistic nature of correctly receiving packets, and reflects a typical real-world scenario where, for example, the weather may influence the quality of wireless links. The success of the algorithm in such dynamic network conditions suggest that it may also be suitable for mobile ad-hoc networks.

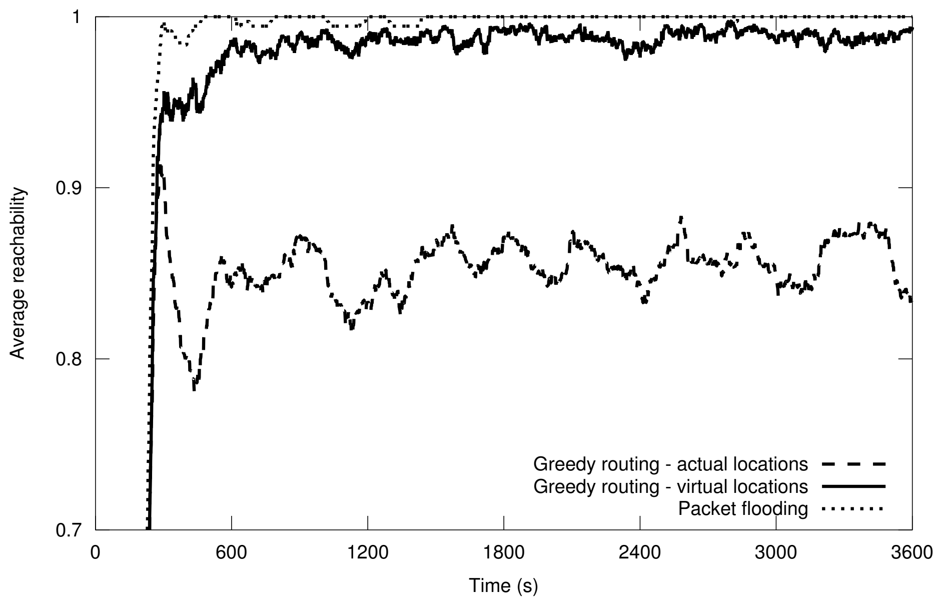


Fig. 4. Reachability of network

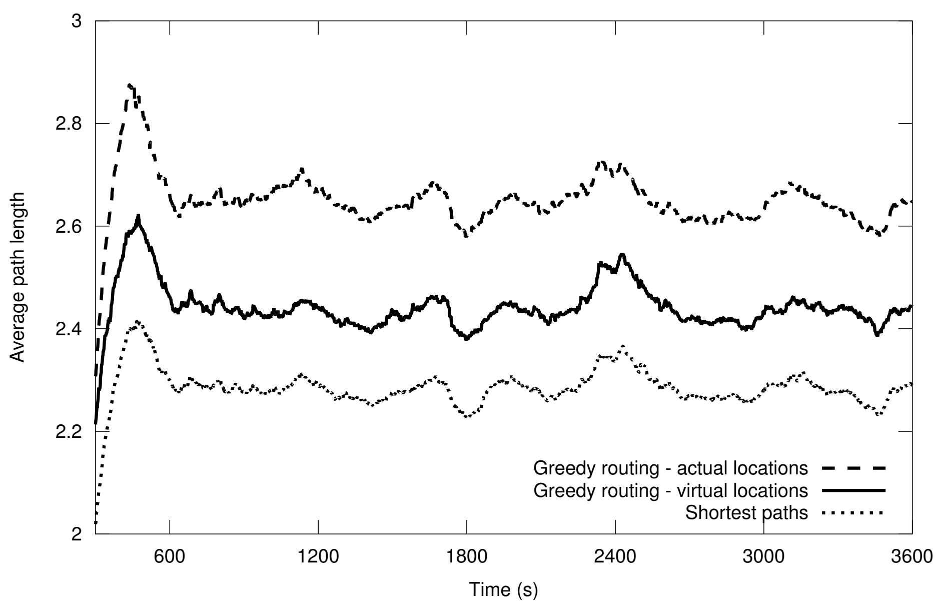


Fig. 5. Length of paths in network

5 Conclusion

A wireless mesh network of packet radio modules was created. The virtual localization algorithm [7], which generates virtual coordinates for networks of arbitrary size scalably, was implemented in this network. The virtual coordinates of the nodes represent the topology of the network in three dimensions better than the two dimensional coordinates in physical space. Greedy routing over the virtual coordinates delivers packets much more reliably than over the physical locations, and results in paths with a lower average hop count. Virtual localization not only improves performance of greedy routing, but also removes the requirement of external localization hardware for the nodes.

References

1. Abramson, N.: The ALOHA system – another alternative for computer communications. In: Proc. November 17–19, 1970, Fall Joint Computer Conf. pp. 281–285. ACM, New York, NY, USA (1970)
2. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Comput. Netw. 38, 393–422 (March 2002)
3. Caruso, A., Chessa, S., De, S., Urpi, A.: GPS free coordinate assignment and routing in wireless sensor networks. In: INFOCOM 2005. Proc. 24th Annu. Joint Conf. IEEE Computer and Communications Societies. vol. 1, pp. 150–160. IEEE (2005)
4. Chen, W., Mei, T., Meng, M.Q.H., Liang, H., Liu, Y., Li, Y., Li, S.: Localization algorithm based on a spring model (LASM) for large scale wireless sensor networks. Sensors 8(3), 1797–1818 (2008)
5. Huc, F., Jarry, A., Leone, P., Rolim, J.: Virtual raw anchor coordinates: A new localization paradigm. In: Scheideler, C. (ed.) Algorithms for Sensor Systems, Lecture Notes in Computer Science, vol. 6451, pp. 161–175. Springer Berlin / Heidelberg (2010)
6. Langendoen, K., Reijers, N.: Distributed localization in wireless sensor networks: a quantitative comparison. Comput. Netw. 43(4), 499–518 (2003)
7. Moore, N., Şekercioğlu, Y.A., Egan, G.: Virtual localization for mesh network routing. In: Proc. IASTED Int. Conf. Networks and Communication Systems (NCS2005). ACTA Press (2005)
8. Niculescu, D., Nath, B.: Ad hoc positioning system (APS) using AOA. In: INFOCOM 2003. Proc. 22nd Annu. Joint Conf. of the IEEE Computer and Communications Societies. vol. 3, pp. 1734–1743 vol.3 (2003)
9. Rao, A., Ratnasamy, S., Papadimitriou, C., Shenker, S., Stoica, I.: Geographic routing without location information. In: Proc. 9th Annu. Int. Conf. Mobile Computing and Networking. pp. 96–108. MobiCom '03, ACM, New York, NY, USA (2003)
10. Sarkar, R., Yin, X., Gao, J., Luo, F., Gu, X.D.: Greedy routing with guaranteed delivery using Ricci flows. In: Proc. 2009 Int. Conf. Information Processing in Sensor Networks. pp. 121–132. IPSN '09, IEEE Computer Society, Washington, DC, USA (2009)
11. Savarese, C., Rabaey, J.M., Langendoen, K.: Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In: Proc. USENIX Technical Annu. Conf. pp. 317–327. USENIX Association, Berkeley, CA, USA (2002)

12. Savvides, A., Park, H., Srivastava, M.B.: The bits and flops of the N-hop multilateration primitive for node localization problems. In: Proc. 1st ACM Int. Workshop on Wireless Sensor Networks and Applications. pp. 112–121. WSNA '02, ACM, New York, NY, USA (2002)
13. Stojmenovic, I.: Position-based routing in ad hoc networks 40(7), 128–134 (Jul 2002)
14. Wireless Sensor and Robot Networks Laboratory (WSRNLab), Monash University, Melbourne, Australia. http://wsrnlab.ecse.monash.edu.au