Robust Node Localization for Wireless Sensor Networks

Radu Stoleru
Department of Computer Science, University of Virginia
Charlottesville, VA 22904
stoleru@cs.virginia.edu

John A. Stankovic
Department of Computer Science, University of Virginia
Charlottesville, VA 22904
stankovic@cs.virginia.edu

Sang H. Son
Department of Computer Science, University of Virginia
Charlottesville, VA 22904
son@cs.virginia.edu

ABSTRACT

The node localization problem in Wireless Sensor Networks has received considerable attention, driven by the need to obtain a higher location accuracy without incurring a large, per node, cost (dollar cost, power consumption and form factor). Despite the efforts made, no system has emerged as a robust, practical, solution for the node localization problem in realistic, complex, outdoor environments. In this paper, we argue that the existing localization algorithms, individually, work well for single sets of assumptions. These assumptions do not always hold, as in the case of outdoor, complex environments. To solve this problem, we propose a framework that allows the execution of multiple localization schemes. This "protocol multi-modality" enables robustness against any single protocol failure, due to its assumptions. We present the design of the framework, and show a 50% decrease in localization error in comparison with state of art node localization protocols. We also show that complex, more robust, localization systems can be build from localization schemes that have limitations.

Categories and Subject Descriptors

C.2.4 [**Computer-Communications Networks**]: Distributed Systems; C.3 [**Special-Purpose and Application-Based Systems**]: Real-Time and embedded systems

General Terms

Design, Performance, Reliability

Keywords

Wireless sensor network, framework, robust localization, protocol composition.

1. INTRODUCTION

Wireless Sensor Network (WSN) systems have been recently developed for several domains: military surveillance, environmental monitoring, habitat monitoring and structural monitoring. These application domains emphasize the requirements for WSN systems: they are expected to work in very diverse environments, the systems need to be reliable and operate un-tethered. When considering the extremely scarce resources available to each sensor node (processing, communication, storage), the aforementioned requirements pose significant challenges. One such challenge is how to accurately find the location of each sensor node, at a low cost. The node localization problem has received tremendous attention from the research community, thus emphasizing that it is an important problem and that it is a difficult problem. It is an important problem because the quality of the data obtained from the WSN and the operation of the network can both be significantly impacted by inaccurate node locations.

Despite the attention the localization problem in WSN has received, no universally acceptable solution has been adopted for realistic, outdoor, environments. There are several reasons for this. One reason is that, in order to obtain a higher location accuracy, localization protocols either make simplifying assumptions (e.g., line of sight with sensor nodes, high density of anchor nodes, deployment knowledge) or require sophisticated hardware. In large scale, realistic outdoor deployments, these assumptions do not always hold, and equipping all sensor nodes with expensive hardware is not feasible. Another reason is that localization protocols that do not have strong simplifying assumptions are generally inaccurate. The research challenge that we face is how to obtain a highly accurate node locations in large scale sensor networks deployed in complex environments, at the lowest cost possible.

Instead of aiming for the "perfect, universally applicable" node localization solution, we propose a framework that allows the execution of several, existing, localization schemes. A system designer decides before deployment what localization protocols to include in the system, the order in which they execute and the conditions that trigger subsequent localization scheme executions. This set of localization schemes is organized in a hierarchical structure. The hierarchy and the localization schemes that are members of the hierarchy are stored on each sensor node. A run-time system is responsible for the coordination, among neighboring nodes, of which localization scheme to execute, and its execution. The main contributions of this paper are the following: a) we propose a new research direction, which is localization robustness through protocol multi-modality; b) we provide a taxonomy for node localization solutions, that allows a system designer to choose an appropriate set for a particular deployment; c) through simulations we show a reduction of 50% in localization error, in a complex environment with obstructions.

2. FRAMEWORK FOR ROBUST LOCALIZATION

The main idea of our framework is to allow the execution of multiple node localization schemes in a WSN deployment. The goal is to reduce the impact of any single localization protocol to the average location estimation error in a partic-

EmNets'07, June 25-26, 2007, Cork, Ireland

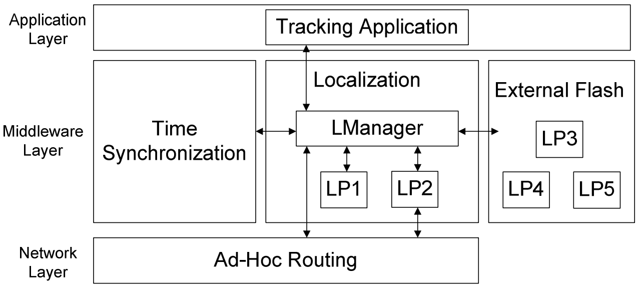

Figure 1: Node Software Architecture.

ular WSN deployment. In the remaining part of this section, we present the architecture and the taxonomy for node localization and our robust node localization framework.

2.1 Node Localization Architecture

The architecture for node localization is shown in Figure 1. We assume that a time synchronization protocol has already executed and that the sensor network is static.

The components of interest for node localization are the Localization Manager (LManager) and the localization protocols that are executing. The manager provides the Application Layer with location services, such as node position. The localization protocols implement a generic interface, for starting and stoping their execution. Due to space constraints, we omit the description of the interface. The Localization Manager is responsible for dynamically loading localization schemes (LPi) from the external flash and for their execution. As shown, the localization protocols that execute can use the network layer for ad-hoc communication.

For the design of our localization framework (localization manager and localization schemes) we need to consider several research questions: are there combinations of localization schemes that are incompatible, and hence can not be executed simultaneously or sequentially? Will the localization schemes be executed serially or in parallel? Should the system execute all localization schemes, for all nodes, and then combine the results? Should the system execute localization schemes only on nodes that have not been localized? How to improve location accuracy through the execution of multiple protocols? We address these questions in the remaining part of the paper.

2.2 Taxonomy for Node Localization Protocols

In this section we propose a taxonomy that partitions the existing body of localization schemes into equivalence classes. These classes provide guiding principles for choosing existing localization schemes to be executed in a WSN system.

Several localization systems and algorithms have been proposed in the past [10] [13] [11] [6] [1] [4] [8] [9] [15]. The large body of solutions for the node localization problem can be categorized based on where the localization algorithms are executed, on the type of ranging hardware, if any, that is available to the sensor node and on the density of anchor nodes (if anchor nodes are used). These categories are described as follows:

2.2.1 *Centralized vs. Distributed Localization Algorithms.*

Centralized localization algorithms require the gathering of connectivity data (range or proximity) from the network to a more computationally powerful device. Once the node locations are computed, they are disseminated back to the network. Examples of such localization solutions are [14] and [2]. While the approach of gathering data centrally is feasible for localizing nodes in some WSN deployment, we believe that this approach is very costly for the envisioned, large scale, WSN deployments. In this paper, instead we focus exclusively on distributed localization algorithms. It is important to remark that some inherently centralized localization algorithms can be made distributed relatively easily. For example, in Spotlight [15], a central device creates well controlled (in time and space) events in the network. The sensor nodes detect and timestamp these events. From the spatiotemporal knowledge for the created events and the temporal information provided by sensor nodes, nodes spatial information can be obtained. In its original version Spotlight requires that the timestamps of the detected events be sent to a central device. An easy modification is to pre-program the sensor nodes with knowledge about the light event that the sensor node will later detect (e.g., where the event is first created, its propagation speed and direction). In this manner, the sensor node can easily compute its location.

2.2.2 *Hardware/Range/Event-based vs. Connectivity-based Algorithms.*

The hardware, range-based schemes are typically high accuracy solutions. The Global Positioning System (GPS) [10] is well known today, widely used, both in military and civil applications. Ranges to several satellites (by measuring the difference in the time of arrival (TDoA) of signals from different satellites) are used in a multilateration procedure to infer the position of the receiver. This localization scheme has a high accuracy, but requires hardware that is both expensive and consumes significant power. Similar TDoA localization techniques, that are based on sophisticated ranging hardware are AHLoS [13], and Cricket [11]. Since the goal of our framework is to obtain a high accuracy in node localization, the hardware/range/event-based localization schemes will be given preference (they will be executed first, since they can provide the most accurate location information). It is important to remark that we do not advocate to equip all the sensor nodes with the sophisticated hardware.

2.2.3 *One hop vs. Multi Hop Algorithms.*

Localization solutions that primarily use connectivity information for inferring proximity to a set of anchors, and hence be able to obtained their locations, have also been proposed [1] [4] [5] [9]. The connectivity-based localization schemes, typically, have a lower accuracy than the hardware /range/event-based schemes. In the Centroid localization scheme [1], the anchor nodes broadcast their location, and a sensor node localizes to the centroid of its proximate anchors (from which it has received beacons). Centroid requires a relatively high anchor density. Localization protocols that can work with a lower anchor density require multi-hop communication, so that any node can infer its distance (in hop counts) to several anchor nodes. Examples of such localization protocols are part of the DV-* family [9]. DV-Hop [9] uses the hop count between sensor nodes and anchors to infer the distances among them. The protocol contains two phases. In the first phase anchors flood the network with beacons and each node records the shortest hop count to each of the anchors. In the second phase, which occurs when one anchor receives beacons from other anchors, anchor nodes estimate the physical distance (Euclidian) of one radio hop. This information is flooded back to the networks, allowing sensor nodes to infer physical distances to anchor nodes (and, hence, localizing themselves).

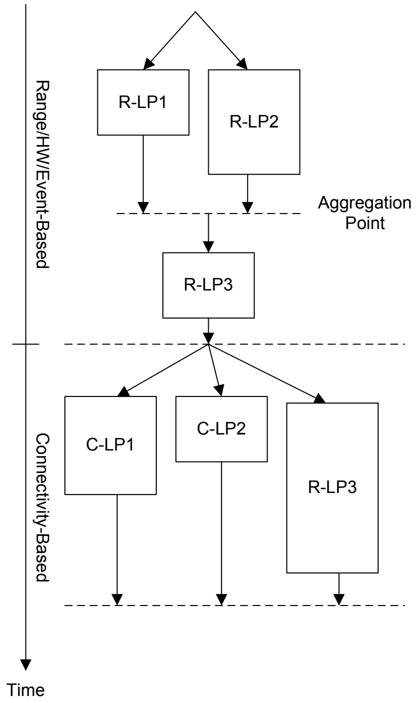

Figure 2: Hierarchical Framework.

2.3 Hierarchical Localization Framework

The main idea for designing how localization protocols are executed is (based on the aforemention taxonomy) to use a hierarchical, multi-phase, multi-protocol operation. The localization protocols that are part of this hierarchy are the protocols available to the Localization Manager (stored in the flash). A generic hierarchical structure is shown in Figure 2.

As shown, the framework operates in two phases. The goal of the first phase (R-LP* boxes in Figure 2) is to achieve the highest location accuracy possible (given the existing capabilities of sensor nodes). Localization schemes that are candidates for this phase are typical Range/Hardware/Event-based schemes. The goal of the second phase (C-LP* boxes in Figure 2) is to estimate the locations of nodes for which the first phase was not successful.

The node localization process starts, for each sensor node, from the top of the hierarchy. The Localization Manager is responsible for loading all localization schemes that execute simultaneously, between two Aggregation Points. When the actively running localization schemes finish execution, and an Aggregation Point is reached, the Localization Manager, "aggregates" through a weighted mean the node locations produced by the localization protocols. For example, the R-LP1 and R-LP2 in Figure 2 could be GPS and Spotlight localization schemes. Because equipping all sensor nodes with GPS is not a feasible solution only a small set of sensor nodes will successfully obtain a location estimate from GPS. Nodes that are in line-of-sight to a Spotlight device are able to obtain their location from executing the Spotlight localization scheme.

If a node obtains a location (and it reaches an Aggregation Point), it stops following the hierarchical localization graph. The reason for this is that the order of execution of localization schemes is such that later localization schemes can not produce a more accurate location estimation.

As shown in Figure 2, by the R-LP3 box, it is possible to have sequential execution of a localization scheme. If a scheme is considered to be "heavier" because it utilizes resources (for example GPS) it can be scheduled to execute only on nodes for which previous schemes have failed. When designing the hierarchical structure for node localization the designer needs to consider that not all nodes may be available for execution of this heavier localization protocol (some nodes successfully localized using earlier localization schemes).

When the Localization Manager finishes the execution of the first phase, it is ready to execute the second phase, if no location information was obtained. The second phase is based on localization schemes that use proximity to anchor nodes. Since neighboring nodes may have obtained their location, the Localization Manager needs to coordinate with its neighbors. This is accomplished by a broadcast of a HELP message by nodes which do not know their locations. The Localization Manager is responsible for broadcasting the HELP message. Only nodes which do not have a location forward the HELP message, and they forward only the first HELP message. This has an effect of controlled flooding of HELP messages. If a node was localized and receives a HELP message, it immediately transitions to the second phase of operation, and it acts as an anchor. It thus, broadcasts its location in a single hop or multi-hop, depending on the type of localization protocols included in the second phase of the hierarchy.

To better understand the second phase of framework, let's assume that the Centroid and DV-Hop are chosen as localization protocols to be executed during the second phase of the framework. Let's consider two scenarios. In one, an isolated sensor node fails to obtain its location from a Range/HW/Event-based scheme. In the second one, a large group of nodes (multi-hop radius) fail to obtain their locations. In the first case, the HELP message sent by the Location Manager on the isolated node is received by the neighboring nodes (successfully localized). They immediately transition to the second phase and act as anchors, broadcasting their locations (simultaneously executing Centroid and DV-Hop). In order to limit the area where beacon nodes are forwarded, we enforce that anchor nodes do not forward beacon nodes from other anchors. In our scenario, the isolated node would then successfully compute its location as the average of locations obtained from the Centroid and DV-Hop schemes. In the second scenario, the nodes that have not been localized and are positioned at the perimeter of the "void" area are able to localize themselves using both Centroid and DV-Hop (and aggregate the results), while the non-localized nodes that are multiple hops away from the perimeter are successfully localized using DV-Hop. This scenario is further described in the case study that follows.

3. CASE STUDY - IMPLEMENTATION AND PERFORMANCE EVALUATION

In order to evaluate the performance of our framework we adopted existing TinyOS implementations of the following localization schemes: Spotlight [15], GPS [10], Centroid [1] and DV-Hop [9]. We modified Spotlight so that sensor nodes do not need to report their timestamps to a central device, and instead, compute their own locations. Since TinyOS does not have capabilities for dynamically loading modules, we statically linked the four localization schemes into the localization manager. The hierarchical framework that we decided to evaluate is depicted in Figure 3.

In TOSSIM, we simulate a sensor network of 400 nodes deployed in a $300 \times 300ft^2$ area. The radio range was fixed at 50 ft. We assumed that 10% of sensor nodes are equipped with GPS devices [10]. To simulate harsh environments, we designated a $200 \times 200ft^2$ area, in the center of the deployment, to be heavily wooded and hence the lack of line of

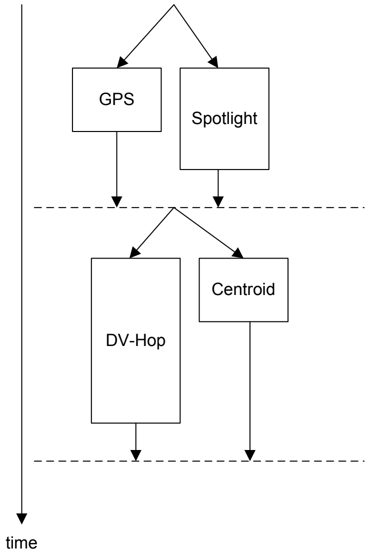

Figure 3: Robust Localization Case Study.

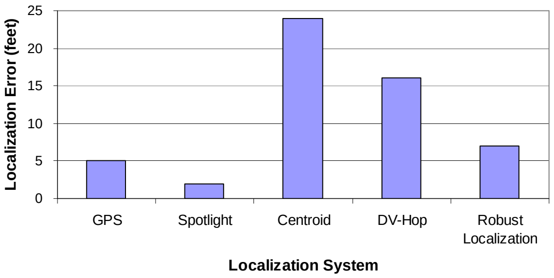

Figure 4: Localization Error.

sight. In addition, any node that is not positioned in the wooded area has a 15% probability of not having a line of sight with satellites or aerial vehicles (and hence, can not be localized through GPS or Spotlight).

The experimental results of 10 simulations with random seeds are shown in Figure 4, Figure 5 and Figure 6. As shown in Figure 4 Spotlight and GPS have the highest accuracies (2-5ft), but they fail to localize all the nodes in the sensor network. On the other hand, DV-Hop localizes the entire network (as shown in Figure 5), but the accuracy is low (16 ft average localization error). The Centroid scheme has also a low accuracy and it fails to localize the entire network, as shown in Figure 5, due to the low number of anchor nodes (we assumed 10% anchors). The Robust Localization scheme, however, successfully localizes the entire WSN and has an average localization error of about 7ft. Even though individual node localization schemes fail (GPS fails for 90% of nodes and Spotlight fails for 50% of nodes), the Robust solution localizes all sensor nodes with a good average localization accuracy.

The price paid for achieving a higher accuracy than individual components is shown in Figure 6 - the overhead (we consider only communication overhead). While GPS and Spotlight have no communication overhead, the Robust Localization scheme has a higher overhead than Centroid, but lower than DV-Hop, which requires a flooding of the entire network (the modified implementation of DV-Hop that the Robust Localization framework uses, does not need to flood the entire network). We consider the overhead relatively modest, when considering the obtained accuracy in node location.

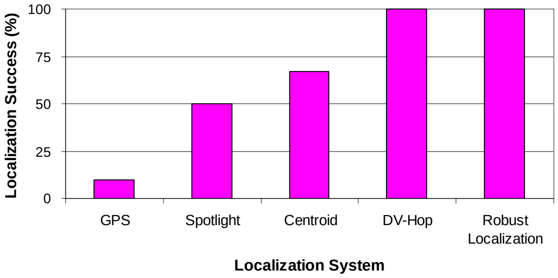

Figure 5: Localization Error.

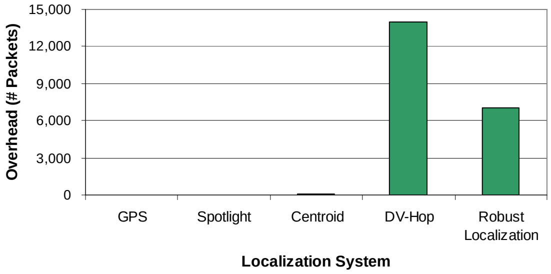

Figure 6: Localization Protocol Overhead.

4. FUTURE DIRECTIONS

The proposed localization framework is just a small part of the more complete solution that will allow a non-expert to build a robust and efficient localization system for a particular WSN deployment. We envision several areas that require further research:

- Develop higher level abstractions for composing localization protocols. Our current implementation for the Localization Manager is done by hand. We aim to provide a programming tool (with a script-like language) for building a localization system from individual localization protocols.

- Develop an analysis tool that evaluates the correctness of a hierarchical localization framework and, possibly, gives soft-guarantees (e.g., largest expected localization error and the overhead required to achieve it).

- Consider a broader set of existing localization schemes and see how well they are accommodated by our framework.

- How to optimize the simultaneous execution of protocols that use, for example, radio communication? Instead of having each protocol send/receive its own messages, an aggregation of data contained in these messages may significantly reduce the communication overhead.

- Could a sensor node be able to evaluate the accuracy of its location estimate and how? This could be a better indicator for stopping the node from executing subsequent localization schemes, than the currently proposed design.

- Analysis of robustness against malicious attacks. Due to localization protocol multi-modality, it is more difficult for an attacker to compromise the integrity of the node localization service.

5. RELATED WORK

Protocol composition has been frequently obtained by layering protocols vertically, in stacks. Horus [16], for example, is based on a vertical stack of protocols, where the events are strictly passed from one layer to the adjacent one. A non-hierarchical protocol composition framework is JGroup/ARM [7]. It introduces the concept of a dependency graph among protocols/layers (graph obtained by each layer registering its interest in other layers). Our work has similarities with HLS [12] (which proposes a hierarchical framework for composing soft real-time schedulers) and [3] (which uses sensing multi-modality for robust localization). In this paper, however, we propose for the first time to provide robustness for node localization through protocol composition. This allows us to overcome the situation when a individual protocol fails (due to its assumptions) through a successful execution of a different localization protocol.

6. CONCLUSIONS

The problem of robust, practical, node localization in WSN remains an important open research problem. In this paper we argue that the existing solutions for node localization can be efficiently used by the proposed framework to ensure a high accuracy at a low cost. Our simulation results support our claims by showing an average reduction in localization error of 50% when compared to individual localization schemes. Our framework proposes a new research direction for robust node localization in WSN: localization protocol multi-modality.

7. ACKNOWLEDGMENTS

This work was supported by the ARO grant W911NF-06-1-0204, by the DARPA grant F336616-01-C-1905, and by the NSF grant CNS-0614870. We would like to thank the anonymous reviewers for their valuable feedback.

8. REFERENCES

[1] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, October 2000.

[2] L. Doherty, K. Pister, and L. Ghaoui. Convex position estimation in wireless sensor networks. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, 2001.

[3] L. Girod and D. Estrin. Robust range estimation using acoustic and multimodal sensing. In *Proceedings of IEEE/IRJ International Conference on Intelligent Robots and Systems (IROS)*, 2001.

[4] T. He, C. Huang, B. Blum, J. A. Stankovic, and T. Abdelzaher. Range-Free localization schemes in large scale sensor networks. In *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2003.

[5] L. Lazos and R. Poovendran. SeRLoc: Secure range-independent localization for wireless sensor networks. In *Proceedings of ACM Workshop on Wireless Security (WiSe)*, 2004.

[6] M. Maroti, B. Kusy, G. Balogh, P. Volgyesi, A. Nadas, K. Molnar, S. Dora, and A. Ledeczi. Radio interferometric geolocation. In *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2005.

[7] H. Meling. Non-hierarchical dynamic protocol composition in Jgroup/ARM. In *Proceedings of Norsk Informatikkonferanse*, 2006.

[8] R. Nagpal, H. Shrobe, and J. Bachrach. Organizing a global coordinate system from local information on an ad hoc sensor network. In *Proceedings of International Conference on Information Processing in Sensor Networks (IPSN)*, 2003.

[9] D. Niculescu and B. Nath. DV-based positioning in adhoc networks. *Telecommunication Systems*, 2003.

[10] B. Parkinson and J. Spilker. Global positioning system: theory and applications. *Progress in Aeronautics and Astronautics*, 1996.

[11] N. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket compass for context-aware mobile applications. In *Proceedings of International Conference on Mobile Computing and Networking (MobiCom)*, 2001.

[12] J. Regehr and J. A. Stankovic. HLS: A framework for composing soft real-time schedulers. In *Proceedings of IEEE Real Time Systems Symposium (RTSS)*, 2001.

[13] A. Savvides, C. Han, and M. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of International Conference on Mobile Computing and Networking (MobiCom)*, 2001.

[14] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz. Localization from mere connectivity. In *Proceedings of ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2003.

[15] R. Stoleru, T. He, J. A. Stankovic, and D. Luebke. A high-accuracy low-cost localization system for wireless sensor networks. In *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2005.

[16] R. van Renesse, K. P. Birman, R. Friedman, M. Hayden, and D. A. Karr. A framework for protocol composition in Horus. In *Principles of Distributed Computing*, 1995.