Demo Abstract: GRAIL: General Real-Time Adaptable Indoor Localization

Yingying Chen[†], Eiman Elnahrawy[‡], John-Austen Francisco[†], Konstantinos Kleisouris[†],
Xiaoyan Li[*], Hongyi Xue[†], Richard P. Martin[†,‡]
[†]{yingche,deymious,kkonst,xuehy58,rmartin}@cs.rutgers.edu
[‡]eiman@kordinate.com, [*]xili@cs.lafayette.edu

[†]Dept. of Computer Science
Rutgers University
Piscataway, NJ 08854

[‡]Kordinate LLC
200 Centennial Ave., Ste 200
Piscataway, NJ 08854

[*]Dept. of Computer Science
Lafayette College
Easton, PA 18042

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: wireless communication

General Terms

Algorithms, Measurement, Performance, Design, Experimentation

Keywords

Localization, Wireless Local Area Networks, 802.11

1 Introduction

Positioning nodes in sensor networks is important because the location of sensors is a critical input to many higher-level networking tasks, such as tracking, monitoring and geometric-based routing. Working towards the goal of a scalable and practical sensor node location service, we present the Generalized Real-Time Adaptable Indoor Localization (GRAIL) system. GRAIL provides general purpose localization for wireless packet networks. In this demonstration, we show GRAIL simultaneously positioning several 802.11 and 802.15.4 devices with an accuracy of 2-3m within 2 seconds.

2 System Goals

Localization is a diverse area covering everything from lower-layer physical problems to application-level services. Within this extensive field, GRAIL seeks to perform general-purpose localization in building-sized environments where devices have easy access to gateway nodes. Additional goals of the GRAIL system include:

General Purpose. A primary goal of the GRAIL system is to localize wireless devices over a variety of physical modalities and networks. Much as a networking system should support multiple media access layers, a general purpose localization system should support multiple physical modalities and methods of localization.

GRAIL was designed to localize using any wireless network that supports physical layer measurements of packet data. Some properties we have used include the received signal strength indicator (RSSI), Angle-of-Arrival (AoA), and time-of-arrival (TOA). In this demonstration, we show the GRAIL system simultaneously localizing both 802.11 devices (laptops) as well as 802.15.4 devices (Telos motes) using RSSI.

Real-time. Latency is a key property of localization systems because it defines the maximum mobility that can be supported. Our demonstration can return results within 2 seconds, allowing us to support both stationary devices as well as those moving at walking speeds.

Adaptable. A common problem with many systems is that they are too brittle; they require specific environments, hardware, or much training data related to a specific set-up. GRAIL uses feedback to dynamically calibrate its parameters due to changing radio conditions. This demonstration shows GRAIL localizing devices throughout the demonstration floor during the typical chaotic radio conditions that arise during exhibitions.

Indoor. Indoor environments are especially challenging due to reflections, refractions, and scattering, which result in substantial multi-path effects. GRAIL manages the uncertainly of these effects. The demonstration shows how GRAIL can expand and contract the possible set of locations as we introduce or reduce the uncertainty in the environment.

3 Architecture

Figure 1 shows the architecture of the GRAIL system. It is built around 4 logical types of objects: landmarks, a server, solvers and transmitters. Localization using the GRAIL system thus assumes the sensor field has access to powerful computing nodes. More specifically, the abstractions GRAIL uses are:

Transmitters. Any device that transmits packets can be positioned, including weak sensor nodes. Packet-level localization is also important because it allows us to localize using existing traffic. This gives GRAIL tremendous advantages in generality and ease of use over other approaches. For example, often the application code does not need to be altered on a sensor node to localize it.

Landmarks. These gateway-class nodes listen to packet traffic and know their location. Landmarks also summarize

SenSys'06, November 1–3, 2006, Boulder, Colorado, USA.

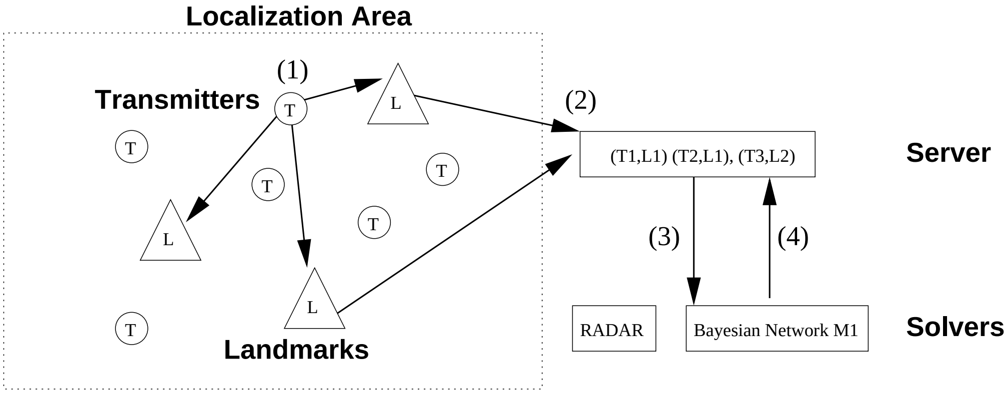

Figure 1. GRAIL Demonstration System Components and Process

data before forwarding it to the localization server. Landmarks do not need much computing capability, but need to aggregate observable traffic; e.g. averaging the RSSI over many observed transmissions.

Server. A server collects information from the landmarks. We found that a centralized solution has critical advantages that are often overlooked in the literature. First, it makes cleaning and summarizing the traffic observations much easier. Second, it enables a variety of additional services such as attack detection and tracking to use the same framework. Finally, we believe that centralization makes enforcing contracts and privacy policies tractable. However, we will leave open the issues of privacy contracts and policy enforcement as future work.

Solvers. A solver takes readings collected by the landmarks and returns localization results. GRAIL's architecture is flexible with regards to the solver; multiple solvers can run simultaneously. This allows it to run multiple localization algorithms, as well as load balancing between solvers.

The GRAIL demonstration currently can choose between two solvers. One solver utilizes the statistical WinBugs tool. In addition, we have implemented the *Fast Solver*. It was developed by using a novel sampling technique and can perform localization 9 to 17 times faster than the *WinBugs Solver* [1]. Additional localization algorithms can also easily be added to the system.

4 The Localization Process

Figure 1 shows the following 4 steps GRAIL uses to localize a wireless transmitter:

1. The senor node sends a packet. Some number of landmarks observe the packet and record physical layer properties of the packet. In the demonstration, RSS is the primary physical layer modality. However, Angle of Arrival and Time-of-Flight can easily be supported.
2. The landmarks forward traffic summaries of the physical layer properties observed from transmitting devices to the server. For example, for each transmitter, the observing landmarks and history of the RSSI would be sent.
3. The server presents the solver with the necessary data for localization. Depending on the algorithm, some additional selection of the data may occur on the server.
4. The solver returns the coordinates of each sensor node. In addition, we will show using the Bayesian network solver that an estimation of the uncertainty in the position can be returned as well.

5 Demonstration

We use 4 laptops as landmarks. Each laptop has 2 802.11 networks. One passively scans the Wifi traffic on all 12 channels. The other network sends summaries of packet-level observations to the server. A packet summary is an averaged RSSI for each source address of interest. The landmarks should be placed in a relatively square or triangular pattern with legs of at least 20-30ft.

The landmarks also have a Telos mote plugged in that passively observes 802.15.4 packet traffic. Summaries of this traffic are also sent to the localization server. A 5th laptop runs the localization server, solvers and GUI.

GRAIL can perform several demonstrations types. In the "hide the mote" demonstration, a conference attendee is given a mote and has to place it somewhere on the demo floor. The exhibitor then finds the mote using the GRAIL system.

In the second demonstration, the exhibitor shows the positions of many 801.11 devices in the room using the GUI. If sufficient traffic exists, we will show the locations of a number of other exhibitors' devices without interfering in their demonstrations via GRAIL's passive scanning. If such traffic is not available, we will show GRAIL localizing our own WiFi PDA's.

The third demonstration will show the expansion and contraction of the uncertainty of the localization result. As a user removes an external antenna to a Telos mote, the Bayesian Network M1 solver will expand and contract the 90^{th} percentile results of the X and Y positions.

Acknowledgments

This work was supported by NSF grant # CNS-0448062.

6 References

[1] K. Kleisouris and R. P. Martin, "Reducing the Computational Cost of Bayesian Indoor Positioning Systems," in *Proceedings of the Third IEEE International Conference on Sensor and Ad hoc Communcations and Networks (SECON),* September 2006.