Modelling Location Sensors

Jörg Roth
University of Hagen
58084 Hagen
Germany
Joerg.Roth@Fernuni-hagen.de

1 Introduction

Location-aware applications rely on location sensor input to provide location-dependent output. Many frameworks assume ideal location sensors with high accuracy, availability and coverage and ignore potential measuring costs. Real location sensors, however, have certain limitations concerning location input und lead to certain costs (e.g. drain the battery or lead to monetary costs. In this paper, we present a framework to model location sensors. The framework is included into the Nimbus platform for location-based services [9].

2 Location Sensor Models

A typical domain for modelling location sensors is robotics. Mobile robots have to determine their location with the help of sensors such as distance sensors, odometers, or visual sensors. The concept of sensor data fusion is strongly related to location sensor models: to increase availability, coverage and accuracy, the output of multiple location sensors can be combined to a single output. There are three major ways to model location sensor output: *points*, *position probabilities* and *areas*.

Points in space: The simplest model of a location is a single point in space. Ignoring accuracy and uncertainty issues, a measurement could be described by a single coordinate such as N51^{o}22.579/E7^{o}29.615/169 m. Although it is very unlikely that the measured and the true location are identical, many applications simply take the measured value for further processing without considering the measurement error. As some positioning systems have a high accuracy, this approach is often suitable.

Position probabilities: More advanced approaches model location output by a statistical distribution of measurements. For two dimensions we get a probability distribution for a certain measurement. For a small grid element (e.g. of 10 m x 10 m), the distribution expresses the probability for a user to reside in this grid element. In this context we talk about the position probability of a mobile user. Statistical methods based on Bayes' rule, the Kalman Filter or Position Probability Grids can be used to fuse multiple senor input.

Areas: As a last model for location output, areas could be used. An area describes the set of possible user locations when a certain measurement is performed. In particular, location output based on cell of origin paradigm (e.g. GSM cell positioning) can easily be described by areas. The area model is suitable to represent the locations as we can easily specify whether a certain location belongs to a cell or not. The area model could be viewed as a simple statistical representation where the area border separates two regions with a uniform distributed position probability – inside the area the sum of probabilities is 1, outside it is 0.

3 The Nimbus Location Sensor Model

To model the different attributes of location sensors, we introduce the *Virtual Positioning System* (*VPS*) which hides the specific details of underlying real positioning systems. With this component, the framework can use arbitrary positioning systems. These systems range from satellite positioning systems (e.g., GPS), positioning with cell phone networks (e.g., GSM) to indoor positioning systems (e.g. our PalmSpot system [9]). In addition, a simulation tool can simulate sensor information and can be used to set up realistic test scenarios without going to the respective locations.

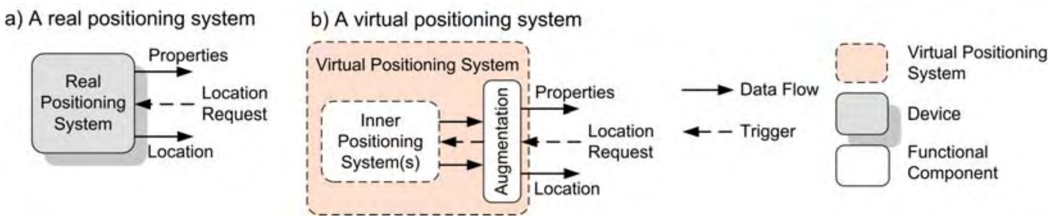

Fig. 1: Real and virtual positioning systems

A real positioning system (fig. 1a) is modelled as a black box that produces a location output on location request. The main idea of a virtual positioning system is to provide location output with higher quality (e.g., higher availability, greater coverage) and, at the same time, to preserve the general interface of location, properties and location request. Fig. 1b shows a virtual positioning system component. It contains one or more inner positioning systems and an augmentation component. The augmentation component takes the location output of the inner systems and generates new location output with improved characteristics. Typical types of augmentations hide details of inner systems, provide access functions and run a capturing protocol, convert one type of location information into information with a richer meaning or a standardized format or merge the output of several positioning systems into a single output.

In principle, virtual positioning systems can be arbitrarily nested. Several combinations are conceivable which can even be established at runtime. Nimbus uses a fixed structure that is a direct result of the initial project goals. It contains the following augmentation components (from the innermost VPS to the outermost):

- *Access:* integrate real positioning system via a driver framework.
- *Mapping*: convert local location data into location data with a global meaning.
- *Selection & Collection*: provide a single location measurement even though different positioning systems are available.
- *Resolution*: convert physical into semantic locations using Nimbus semantic resolution capabilities [5, 9].

These components lead to the structure of nested VPSs as presented in fig. 2.

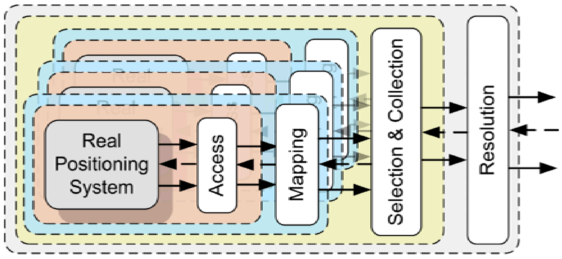

Fig. 2: Virtual positioning systems in Nimbus

3.1 Modelling Locations

From the alternatives presented above, Nimbus uses the area model to model location sensor output. Often, positioning systems are delivered without any detailed information about the probability distribution. To find out the actual distribution by measurements is very cost-intensive, so usually the distribution is simply estimated. In such cases, the area representation is a reasonable alternative to the exact probability distribution. Using the area model to represent systems with normal distributed location values is not optimal, but the region that covers 95% of the possible locations can be used for the area model without losing too much information.

Compared to detailed statistical models, areas are a simplification: first, the area does not identify a specific position with the highest position probability. Second, a strong border often cannot be sharply defined, as there is usually no most distant possible location for a certain measurement. As an advantage, an area is a very compact representation of a location sensor output. In addition, this model leads to very efficient processing algorithms.

3.2 Selecting Sensors

Currently, many projects in the area of location-based services or applications assume a single positioning system. Future mobile end-user devices will probably have access to several positioning systems. In a typical future scenario, a user has access to satellite navigation via GPS, indoor positioning systems which cover some buildings and positioning via a cell phone network, whenever more accurate systems fail. Applications, however, do not want to deal with various positioning systems and require a single location. There are two mechanisms to deal with several positioning systems:

- a subset of all positioning systems currently connected to the mobile device is activated (*selection* component), and
- the location information of all activated positioning systems that currently provide output is merged into a single output (*collection* component).

Collection: A simple approach to merge output of different location sensors is to simply choose the most accurate one. In many cases this approach is effective. Whenever an accurate system is available, less accurate systems do not contribute substantial information. As an example we consider two positioning systems: the Cricket indoor system (accuracy approx. 0.3 m) and GSM positioning using cell IDs (accuracy some hundreds meter in cities). The latter system does not provide any useful information as long as the Cricket system is available. Collecting the most accurate system for output requires only low computational resources and is thus a suitable approach for weak clients.

An improved collection component uses sensor data fusion. Whenever two or more positioning systems provide similar accuracy, the fusion can significantly improve the accuracy. According to the area sensor model, a user location has to be inside each of the provided areas, thus the geometrical intersection of all outputs describes all potential user locations.

Selection: In principle, the selection component can activate all positioning systems that are currently connected to the mobile system. In this case, all available location data are collected. This simple approach is appropriate if positioning is free of charge. Unfortunately, the user is sometimes charged for positioning services, for example when mobile phone networks are involved. For mobile systems also non-monetary costs such as power consumption have to be considered. A selection component can for example deactivate an unused positioning system to save battery power.

As a major problem, the selection cannot determine, if a positioning system produces output before the system is activated. E.g. the selection can decide to activate GPS (which drains the battery), but the GPS receiver does not receive any satellites. As the availability status may undesirably change after a selection, the algorithm can only try to select the most appropriate systems. An algorithm presented in [9] makes a selection based on former measurements. It maximizes the accuracy and availability, but the expected positioning costs do not to exceed a certain cost limit. Fig. 3 shows a simulation of this algorithm.

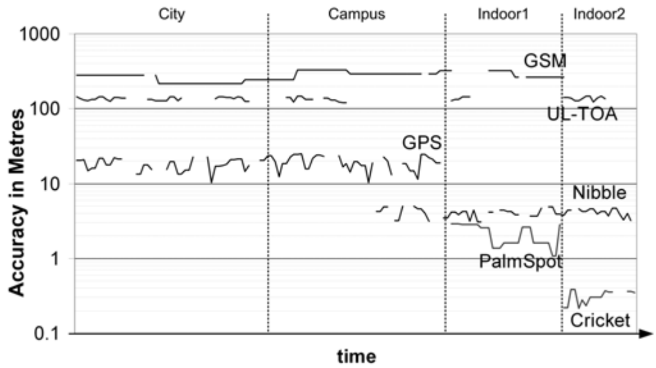

Fig. 3: Simulation of the selection component

4 Conclusion

In this paper, an approach to deal with different kinds of positioning systems is presented. Virtual positioning systems model all stages of abstractions from raw location output generated by real positioning systems up to high-level location output containing physical areas and semantic locations. The concept of drivers in particular simplifies the integration of new positioning systems in the future. Simulated positioning systems providing realistic testing scenarios are used during the application development phase.

Often, location output is considered as a single point in space. In reality, this assumption leads to misleading results, as many positioning systems only provide inaccurate location information. A location sensor model sketched in this paper addressed this issue.

1. Jörg Roth: *Mobile Computing*, dpunkt-Verlag, 2002
2. Jörg Roth: *Context-aware Web Applications Using the PinPoint Infrastructure*, IADIS International Conference WWW/Internet 2002, Lisbon (Portugal), Nov. 13-15, 2002, IADIS Press, 3-10
3. Jörg Roth: *Flexible Positioning for Location-based Services*, IADIS International Conference e-Society, Lisbon (Portugal), June 3-6, 2003, Vol. I, IADIS Press, 296-304
4. Jörg Roth: *Semantic Geocast Using a Self-organizing Infrastructure*, Innovative Internet Community Systems (I2CS), Leipzig, June 19-21, 2003, LNCS 2877, Springer-Verlag, 216-228
5. Jörg Roth: *Accessing Location Data in Mobile Environments – the Nimbus Location Model*, Mobile HCI 03 Workshop on Mobile and Ubiquitous Information Access, Udine (Italy), Sept. 8, 2003, LNCS 2954, Springer-Verlag, 256-270
6. Jörg Roth: *Flexible Positioning for Location-based Services*, IADIS Journal on WWW/Internet, Vol. I, Nr. 2, Dec. 2003, IADIS Press, 18-32
7. Thomas Hadig, Jörg Roth: *Proximity Services with the Nimbus Framework*, IADIS International Conference Applied Computing 2004, Lisbon (Portugal), March 23-26, 2004, IADIS Press, 437-444
8. Jörg Roth: *Data Collection*, in Jochen Schiller, Agnès Voisard (eds), *Location-Based Services*, Morgan Kaufmann Publishers, May 2004
9. Jörg Roth: *A Decentralized Location Service Providing Semantic Locations*, Habilitation Thesis, University of Hagen, Informatik Bericht 323, Jan. 2005