Chapter 11

PROBABILISTIC METHODS FOR LOCATION ESTIMATION IN WIRELESS NETWORKS

Petri Kontkanen, Petri Myllymäki, Teemu Roos, Henry Tirri, Kimmo Valtonen and Hannes Wettig
Complex Systems Computation Group, Helsinki Institute for Information Technology, University of Helsinki & Helsinki University of Technology, P.O.Box 9800, 02015 HUT, Finland

Abstract: Probabilistic modeling techniques offer a unifying theoretical framework for solving the problems encountered when developing location-aware and location-sensitive applications in wireless radio networks. In this paper we demonstrate the usefulness of the probabilistic modelling framework in solving not only the actual location estimation (positioning) problem, but also many related problems involving pragmatically important issues like calibration, active learning, error estimation and tracking with history. Some interesting links between positioning research done in the area of robotics and in the area of wireless radio networks are also discussed.

Key words: Location estimation, probabilistic modeling, wireless networks

1. INTRODUCTION

The location of a mobile terminal can be estimated using radio signals transmitted or received by the terminal. The problem is called with various names such as location estimation, geolocation, location identification, location determination, localization, and positioning. The traditional, geometric approach to location estimation is based on angle and distance estimates from which a location estimate is deduced using standard geometry. Instead of the geometric approach, we consider the probabilistic approach which is based on probabilistic models that describe the dependency of observed signal properties on the location of the terminal, and

the motion of the terminal. The models are used to estimate the terminal's location when signal measurements are available.

The feasibility of the probabilistic approach in the context of wireless networks has already been demonstrated to some extent in a number of recent papers (Castro et al., 2001; Ladd et al., 2002, Roos et al., 2002a, Roos et al., 2002b; Schwaighofer et al., 2003; Youssef et al., 2003). Probabilistic methods have also been extensively used in robotics where they provide a natural way to handle uncertainty and errors in sensor data (Smith et al., 1990; Burgard et al., 1996; Thrun, 2000). In a recent survey (Thrun, 2003), Sebastian Thrun summarizes the central role of probabilistic methods in robotic mapping as follows:

> ''Virtually all state-of-the-art algorithms for robotic mapping in the literature have one common feature: They are probabilistic. […] The reason for the popularity of probabilistic techniques stems from the fact that robot mapping is characterized by uncertainty and sensor noise. ''

Many of the probabilistic methods developed in the robotics community, in particular those related to mapping, location estimation and tracking, are also applicable in the context of wireless networks. In the following we discuss selected topics in probabilistic location estimation, many of which are well-known in probabilistic modeling, but have received relatively little attention in the domain of wireless networks.

We focus primarily on wireless local area networks, WLANs, but most of the ideas and concepts are applicable to many other wireless networks as well, including those based on GSM/GPRS, CDMA or UMTS standards. The rest of te paper is organized as follows: In Section 2 we discuss *calibration*, the process of obtaining a model of the signal properties at various locations. The actual location estimation and tracking phase following calibration is considered in Sections 3 and 6. Issues related to the optimal choice of calibration measurements are discussed in Section 4. In many cases, it is useful to complement a location estimate with information on its accuracy; in Section 5 we describe methods for error estimation and visualization. Conclusions are summarized in Section 7.

2.   CALIBRATION

In order to obtain a positioning model, we need to estimate the distribution of the signal properties, e.g., signal strength, as measured by the device to be localized for the various locations in consideration. This has traditionally been done using knowledge of radiowave propagation. Several propagation prediction or *cell planning* tools are available for this purpose

(Andersen et al., 1995; Wölfle and Landstorfer, 1999). We adopt an empirical approach, i.e. we estimate the required distributions from *calibration data* gathered at different locations in consideration. Experimental studies suggest that propagation methods are not competitive against empirical models in terms of positioning accuracy due to insufficiently precise signal models (Bahl and Padmanabhan, 2000, Roos et al., 2002b).

Consider a finite set of calibration points *l*, which are labelled by their *x* and *y* coordinates (and possibly a third coordinate *z* or other additional information). For each calibration point we gather calibration data, i.e., a number of observation vectors *o* to estimate the distribution of signal properties from. For a discussion on how this can and should be done for a given set of calibration points and observations see (Castro et al., 2001; Roos et al., 2002a; Schwaighofer et al., 2003; Youssef et al., 2003). When we then want to position a device with current signal readings *o*, we calculate the probabilities *p(l | o)* for each possible location *l* using the Bayes rule and the distributions estimated from the calibration data as described in Section 3. For simplicity, we assume that the set of possible locations can be considered equal to the set of calibration points. If this is not the case, e.g., when a continuous location variable is used, we need to interpolate in order to obtain a distribution of the signal properties at locations from which no calibration data is available. Note that without interpolation we can model only a finite—and for practical reasons preferably not too large—set of possible locations. We can then determine for example in which room a device is located (with certain probability), but have no probabilities associated to locations inbetween the calibration points.

But how should we choose the set of possible locations and how should we collect calibration data? A simple solution is to use a *probability grid* (Burgard, 1996), dividing the positioning space into cells of some size, e.g., 1m × 1m. In order to obtain a distribution of the signal properties at each grid point without interpolation, one then needs to collect a sufficient number of training vectors at each grid point. However, it may be impractical to remain at each grid point for the time it takes to gather enough data—and favorably move around in order to capture variance due to orientation and within the area of the cell—before moving on to the next cell.

A more convenient way is to gather data vectors continuously just walking (or driving) around. We only need to record the time label of each observation and the time labels and coordinates of those locations at which the calibrator changes direction and/or speed. This way we quickly obtain a large number of observations equipped with their exact location. However,

we (usually) get only one observation per location, which does not suffice to reliably estimate the distribution of signal properties. Furthermore it is computationally problematic to deal with such a large number of possible locations in a model; note that when a device supplies us with an observation vector every 500ms, a calibration round of an hour already yields up to 7200 locations.

A natural way of dealing with this situation is to group the locations into clusters (Youssef, 2003). Each cluster should consist of a sufficient number of vectors to supply a good estimate of the signal properties in its area, and as its location we may take—for example—the center of gravity of its measurements' positions. An interesting and theoretically appealing way to produce such clustering is given by the principle of *Minimum Description Length* (MDL) in its most recent form, the *Normalized Maximum Likelihood* (NML) (Rissanen, 1996), for details see (Kontkanen et al., 2004). Figure *1* shows such clustering of a calibration tour. Note, that there is no need to decide on the number of clusters in advance, the algorithm will choose as many as can reliably be distinguished from the data collected.
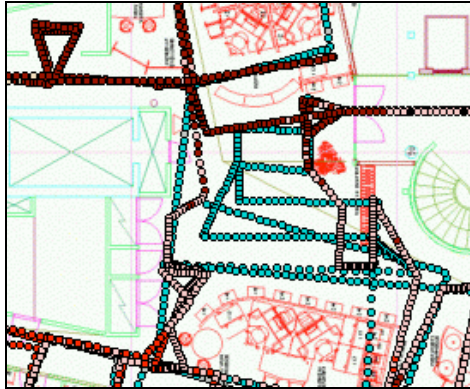

Figure 11-1. NML clustering of signal data collected continuously along a calibration tour. Each circle is represents one vector of measurements gathered at its position, the different clusters are colour-coded.

3.   LOCATION ESTIMATION

After the calibration phase we have, for any given location l, a probability distribution $p(o \mid l)$ that assigns a probability (density) for each measured signal vector o. By application of the Bayes rule, we can then

obtain the so called *posterior distribution* of the location (Roos et al., 2002b):

$$p(l \mid o) = p(o \mid l) \, p(l) \, / \, p(o) = p(o \mid l) \, p(l) \, / \left(\sum_{l' \in L} p(o \mid l') \, p(l') \right),$$

where $p(l)$ is the *prior probability* of being at location l before knowing the value of the observation variable, and the summation goes over the set of possible location values, denoted by L. If the location variable is continuous, the sum is replaced by the corresponding integral.

The prior distribution $p(l)$ gives a principled way to incorporate background information such as personal user profiles and to implement tracking as described in Section 6. In case neither user profiles nor a history of measured signal properties allowing tracking are available, one can simply use a uniform prior which introduces no bias towards any particular location. As the denominator $p(o)$ does not depend on the location variable l, it can be treated as a normalizing constant whenever only relative probabilities or probability ratios are required.

The posterior distribution $p(l \mid o)$ can be used to choose an optimal estimator of the location based on whatever loss function is considered to express the desired behavior. For instance, the squared error penalizes large errors more than small ones, which is often useful. If the squared error is used, the estimator minimizing the expected loss is the expected value of the location variable:

$$\mathrm{E} \, [\, l \mid o \,] = \sum_{l \in L} l \, p(l \mid o),$$

assuming that the expectation of the location variable is well defined, i.e., the location variable is numerical. Location estimates, such as the expectation, are much more useful if they are complemented with some indication about their precision. We discuss error estimates in Section 5 below.

The presented probabilistic approach can be contrasted with the more traditional, geometric approach to location estimation used in methods such as angle-of-arrival (AOA), time-of-arrival (TOA), and time-difference-of-arrival (TDOA). In the geometric approach the signal measurements are transformed into angle and distance estimates from which a location estimate is deduced using standard geometry. One of the drawbacks of the geometric approach is that there is no principled way to deal with the incompatibility of the angle and distance estimates caused by measurement errors and noise.

On the other hand, the geometric approach is usually computationally very efficient.

4.   ACTIVE LEARNING

In Section 2 we only considered the problem of obtaining a model of the signal properties given training data collected from known locations. The resulting model is strongly dependent on where and how much training data is collected. Obviously, the training data should not leave large areas uncovered or otherwise there would be no way to reliably infer the signal properties in such areas. Also, for various reasons, for some areas the signal model is required to be more accurate than in general, in order to achieve accurate location estimation. For instance, two distinct locations may be roughly similar in terms of signal properties so that they can be told apart only by a small margin. In such areas, more extensive calibration is required.

In practice, if it is possible to collect a large amount of training data, a reasonable calibration result is obtained by collecting training data roughly uniformly from each location. Areas where the signal properties are expected to vary within small distances due to, for instance, large obstacles, may be better covered with relatively higher density, whereas large open areas where the signal is likely to be constant, can be left with less attention. In case extensive calibration is costly or otherwise impossible, it becomes critical to choose the calibration points as well as possible. The problem of choosing optimal actions in order to reduce uncertainty has been studied in the robotics literature under the name *robotic exploration*. In general, optimal decision strategies are intractable and various heuristics are used (Burgard et al., 2000; Thrun, 2003).

A practical method for locating potentially useful candidates for new calibration points is based on the estimate of the future expected error. This estimate is calculated by summing over all possible future observations o:

$$\mathrm{E}\,[err \mid l\,] = \sum\nolimits_{o} \mathrm{E}\,[err \mid l,o]\,\mathrm{p}(o),$$

where l is the calibration point candidate, and $\mathrm{E}[err \mid l,o]$ is the expected error:

$$\mathrm{E}\,[err \mid l,o\,] = \sum\nolimits_{l' \in L} \mathrm{p}(l' \mid o)\,d(l',l),$$

for the preferred distance function d.

The candidate points *l* can be chosen by using a tight grid. For example, the grid spacing could be approximately one meter. One or more grid points with a high expected error, or points surrounded by several such grid points, are then used as new calibration points. If the dimensionality of the observation vector *o* is so high that the summing over all *o* as above is not feasible, the sum can be approximated by sampling. An ever simpler approach is to use the calibration data as the set over which the sampling is performed, in which case one only needs to sum over the calibrated observations.

To implement the method based on equations above, one needs to determine the probability distribution or density over the future observations. In practice, it has to be approximated from the calibration data. One possible approximation method is as follows. When computing E[*err* | *l*] for some location *l*, one replaces the *p(o)* by the probability distribution based on the past observations made at the calibration point closest to *l*. The efficiency of the method can then be further improved by approximating E[*err* | *l,o*] by *d(l*, l)*, where *l** is the point estimate produced by the positioning system after seeing observation *o*.

5.   ERROR ESTIMATION AND VISUALIZATION

In order to visualize the uncertainty associated with the location, we assume that we have a probability distribution *p*, either a probability mass function or a density, which describes the uncertainty about the actual location. In addition to reporting a point estimate—here taken to be the expected value—we can visualize the uncertainty related to distribution *p*. This can be done, for instance, by drawing an ellipse centered at the expected location such that the orientation and size of the ellipse describes the uncertainty of the location estimate as well as possible.

As a first step of obtaining such an "uncertainty ellipse" one first needs to obtain certain summary statistics from the distribution *p*. These statistics are, in addition to the expectation, contained in the *variance-covariance matrix*. The variance-covariance matrix describes the variance of the location in both *x* and *y* coordinates together with the correlation of the two coordinates. The second step is to evaluate the two *eigenvectors* of the variance-covariance matrix. This is a simple exercise in linear algebra. For instance, in case the two coordinates *x* and *y* happen to be independent in the distribution *p*, i.e., there is no correlation, the eigenvectors are parallel to the two coordinate axes. Finally, one displays an ellipse whose axes are parallel to those given by the two eigenvectors of the variance-covariance matrix. The lengths of

the axes are given by the *eigenvalues* multiplied by a scaling constant. We give a rule for determining the value of the scaling constant below, after we have first discussed the interpretation of the ellipse.

One interpretation for the uncertainty ellipse is that assuming (pretending) that the estimated density of the location is bivariate Gaussian, the ellipse is the smallest area that contains a fixed probability mass. Given the probability mass to be covered by the ellipse, one can obtain the aforementioned scaling constant by taking the square root of the Chi-squared value with two degrees of freedom. For instance, if 95 % coverage is required, the scaling constant becomes $\sqrt{5.991} = 2.448$. An illustration of the error ellipse is shown in Fig. *2*.
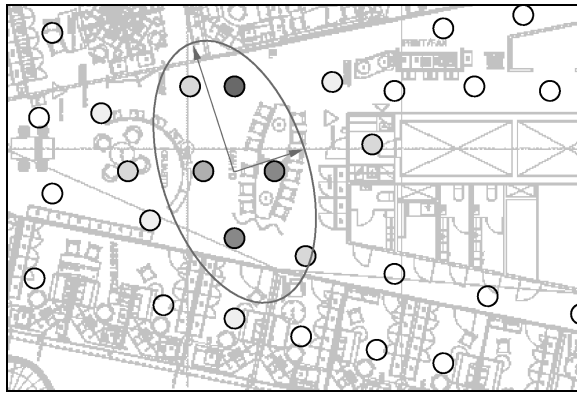
Figure 11-2. Uncertainty ellipse. Probabilities at a discrete set of locations are denoted by circles; dark shading implies high probability. The ellipse centered at the expected location has axes parallel to eigenvectors of the variance-covariance matrix and lengths proportional to eigenvalues.
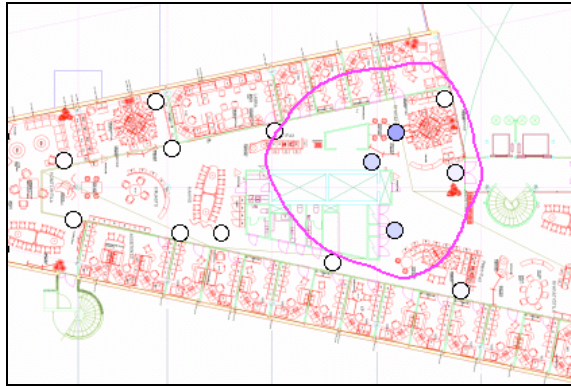
Figure 11-3. Uncertainty about the estimate represented by a polar coordinate system placed at the point estimate. Calibration points are marked by circles, colored depending on *p(x,y)*. The relative amount of uncertainty in each direction away from the point estimate is visualized by the curve.

Whereas the ellipse approach shows the uncertainty about location in two orthogonal directions with respect to the point estimate, a generalization to an arbitrary number of directions can be obtained by mapping *p(x,y)* to a polar coordinate system centered on the point estimate. In this method, the origin is placed at the point estimate and each calibration point mapped to the polar coordinate system *a(x,y),d(x,y)*, where *a(x,y)* is the angle w.r.t. the point estimate and *d(x,y)* is the distance. It is convenient to discretize both *a(x,y)* and *d(x,y)*, resulting in the case of two-dimensional space in a set of segments that partition the space disjointly and exhaustively.

We gain a discrete two-dimensional distribution $p_p(a(x,y), d(x,y))$ over the location space. The curve visualizing a wanted contiguous portion of the total mass can then be derived from $p_p(a(x,y), d(x,y))$. Relative distances from the origin are first determined for each sector based on expected distances. The resulting shape describes relative probability mass in each "direction" (sector). To represent the spread of uncertainty as well, the curve can be scaled so that it covers a desired fraction of $p_p(a(x,y),d(x,y))$. For a screen shot of an implementation, see Fig. *3*.

6.   TRACKING

Location estimation accuracy can be greatly improved if instead of a single signal measurement, a series of measurements is available unless the mobile device is moving with very high speed or the time interval between measurements is very long. Such a series of measurements allows keeping

track of the device's location as a function of time, also called *tracking*. It is convenient to model the situation as a *hidden Markov model* (Rabiner, 1989) illustrated in Fig. *4*.
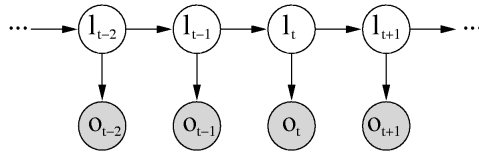

Figure 11-4. Hidden Markov model. State variables (white nodes) are hidden (not observed). Observed variables are denoted by shaded nodes. Horizontal arrows correspond to transition probabilities between successive states. Vertical arrows correspond to observation probabilities given state.

In a hidden Markov model, the variables $l_1, l_2, \ldots$ correspond to a sequence of states indexed by time t. In our location estimation domain, the state correspond to location and hence, the state sequence constitutes a trajectory of the located device. The model also has a set of corresponding observation variables, denoted by $o_1, o_2, \ldots$. Each observation variable, o_t, is assumed to be dependent only on the current location, l_t. In the model in Fig. *4*, the location at time t is dependent on the earlier locations only through the previous location l_{t-1}. Generalizations to higher order dependencies are easily expressed in the general framework of graphical probabilistic models (Cowell et al., 1999; Pearl, 1988).

The power of the hidden Markov model stems from the fact that inference in the model is effective. Given a series of observations, $o_1, \ldots, o_n$, the probability distribution of the location at any given time can be computed in order $O(n)$ operations using the standard probabilistic machinery developed for graphical models. Furthermore, maintaining the distribution of the current location, as observations are made one by one, can be done iteratively such that for each new observation, only constant, $O(1)$, time is needed. However, one should be cautious about the multiplicative factors hidden in the $O(n)$ and $O(1)$ notation. We return to this issue shortly below. Other possible inferences include tracking with a k step *lag*, i.e., maintaining the distribution of the location variable l_{t-k} instead of the most recent location, l_t. This is called *smoothing* as the evolution of the location variable l_{t-k} as a function of time t is smoother than the evolution of the current location l_t. The *Viterbi* algorithm gives the most likely trajectory given a sequence of observations, see (Rabiner, 1989).

In order to apply the hidden Markov model, one needs to specify two kinds of probabilities. First, one needs to determine the conditional

probability distribution of the observation variable given the state variable. This is exactly the aim of calibration as discussed in Section 2. Second, the conditional distribution of each state s_t given the previous state s_{t-1}, called the *transition probability*, has to be determined. The form of these two kinds of conditional probability distributions depends on whether the location and observation variables are continuous or discrete. A continuous linear-Gaussian model for both transitions and observations yields the well-known *Kalman filter* and *smoothing* equations (Kalman, 1960).

In the discrete case, the probability distributions are represented as probability tables, which for transition probabilities constitute an $N \times N$ matrix where N equals the number of possible locations. In the general case, the multiplicative factor in the *O(n)* and *O(1)* notation above for the computational complexity of inference is at least as large as N^2. Methods to reduce the computational complexity of tracking and smoothing when using discrete-valued location include the aforementioned clustering approach that reduces the number of locations N. In addition, a large proportion of state transition probabilities are usually extremely small or zero. In such a case the transition probability matrix is sparse which can be exploited to essentially reduce computational complexity. One can also resort to approximative inference using, for instance, *particle filtering* techniques that try to focus computation on areas of the state space where most of the probability mass lies (Fox et al., 1999).

7.　　CONCLUSIONS

We showed how the probabilistic modelling approach can be used for defining a unifying framework offering a theoretically solid solution to the location estimation problem, and what is more, also to many related, practically important problems involving issues like calibration, active learning, error estimation and tracking with history. Nevertheless, having said that, it must be acknowledged that problems in the real world are always more complicated than the textbook examples, and developing these theoretically elegant solutions to a robust, off-the-shelf software package like for example the Ekahau Positioning Engine (see www.ekahau.com), requires several minor but practically important technical tricks the details of which are outside the scope of this paper. However, we strongly believe that the best way to develop location-aware applications is to start with a theoretically correct, "ideal" solution, and then approximate that solution as accurately as possible given the pragmatic constraints defined by the real-world environment. Our experiences suggest that although it is not the only

possible approach for this, the probabilistic modeling framework offers a viable solution for developing practical applications in this domain.

ACKNOWLEDGEMENTS

REFERENCES

Andersen, J. B., Rappaport, T. S., and Yoshida, S., 1995, Propagation measurements and models for wireless communications channels, *IEEE Communications Magazine*, **33**:42–49.

Bahl, P., and Padmanabhan, V. N., 2000, Radar: An In-building RF-based user location and tracking system, in: *Proc. 19th Annual Joint Conf. of the IEEE Computer and Communications Societies (INFOCOM-2000)*, Vol. 2, Tel-Aviv, Israel, pp. 775–784.

Burgard, W., Fox, D., Hennig, D., and Schmidt, T., 1996, Estimating the absolute position of a mobile robot using position probability grids, in: *Proc. 13th National Conf. on Artificial Intelligence (AAAI-1996),* Portland.

Burgard, W., Moors, M., Fox, D., Simmons, R., and Thrun, S., 2000, Collaborative multi-robot exploration, in: *Proc. IEEE Int. Conf. on Robotics & Automation (ICRA-2000)*, San Francisco.

Castro, P., Chiu, P., Kremenek, T., and Muntz, R., 2001, A Probabilistic room location service for wireless networked environments, in: *Proc. 3rd Int. Conf. on Ubiquitous Computing (UBICOMP-2001)*, Atlanta.

Cowell, R., Dawid, P. A., Lauritzen, S., and Spiegelhalter, D., 1999, *Probabilistic networks and expert systems*. Springer-Verlag, New York.

Fox, D., Burgard, W., Dellaert, F., and Thrun, S., 1999, Monte Carlo localization: Efficient position estimation for mobile robots, in: *Proc. 16th National Conf. on Artificial Intelligence (AAAI-1999)*, Orlando, pp. 343–349.

Kalman, R. E., 1960, A New approach to linear filtering and prediction problems, *Transactions of the ASME–Journal of Basic Engineering*, **82**(Series D):35–45.

Kontkanen, P., Myllymäki, P., Buntine, W., Rissanen, J., and Tirri, H., 2004, An MDL framework for data clustering, in: *Advances in Minimum Description Length: Theory and Applications,* P. Grünwald, I. J. Myung, and M. Pitt, eds., MIT Press.

Ladd, A. M., Bekris, K., Rudys, A., Kavraki, L. E., Wallach, D. S., and Marceau, G., 2002, Robotics-based location sensing using wireless Ethernet, in: *Proc. 8th Annual Int. Conf. on Mobile Computing and Networking (MOBICOM-2002)*, Atlanta, pp. 227–238.

Pearl, J., 1988, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, San Mateo.

Rabiner, L. R., 1989, A Tutorial on hidden Markov models and selected applications in speech recognition, *Proc. of the IEEE*, **77**(2):257–286.

Rissanen, J., 1996, Fisher information and stochastic complexity, *IEEE Transactions on Information Theory*, **42**(1):40–47.

Roos, T., Myllymäki, P., and Tirri, H., 2002a, A Statistical modeling approach to location estimation, *IEEE Transactions on Mobile Computing*, **1**(1):59–69.

Roos, T., Myllymäki, P., Tirri, H., Misikangas, P., and Sievänen, J., 2002b, A Probabilistic approach to WLAN user location estimation, *Int. Journal of Wireless Information Networks,* **9**(3):155–164.

Schwaighofer, A., Grigoras, M., Tresp, V., and Hoffmann, C., 2004, GPPS: A Gaussian process positioning system for cellular networks, in: *17th Annual Conf. on Neural Information Processing Systems (NIPS-2003)*, Vancouver.

Smith, R., Self, M., and Cheeseman, P., 1990, Estimating uncertain spatial relationships in robotics, in: *Autonomous Robot Vehicles*, Springer-Verlag, Berlin-Heidelberg, pp. 167–193.

Thrun, S., 2000, Probabilistic algorithms in robotics, *AI Magazine*, **21**(4):93–109.

Thrun, S., 2003, Robotic mapping: A Survey, in: *Exploring Artificial Intelligence in the New Millennium*, Morgan Kaufmann Publishers, San Francisco, pp. 1–35.

Wölfle, G., and Landstorfer, F. M., 1999, Prediction of the field strength inside buildings with empirical, neural, and ray-optical prediction models, in: *7th COST-259 MCM-Meeting in Thessaloniki*, Greece.

Youssef, M. A., Agrawala, A., and Shankar, A. U., 2003, WLAN Location determination via clustering and probability distributions, in: *IEEE Int. Conf. on Pervasive Computing and Communications (PERCOM-2003)*, Fort Worth.