

---

# Adaptive Localization in Wireless Networks

Henning Lenz<sup>1</sup>, Bruno Betoni Parodi<sup>2</sup>, Hui Wang<sup>2</sup>, Andrei Szabo<sup>2</sup>, Joachim Bamberger<sup>2</sup>, Joachim Horn<sup>3</sup>, and Uwe D. Hanebeck<sup>4</sup>

<sup>1</sup> Siemens AG, Automation and Drives, Advanced Technologies and Standards, Process Automation, A&D ATS 33, Oestliche Rheinbrueckenstr. 50, 76187 Karlsruhe, Germany [henning.lenz@siemens.com](mailto:henning.lenz@siemens.com)

<sup>2</sup> Siemens AG, Corporate Technology, Information and Communications, CT IC 4, Otto-Hahn-Ring 6, 81730 Munich, Germany [betoni@gmail.com](mailto:betoni@gmail.com), {[hui.wang.ext](mailto:hui.wang.ext@siemens.com), [andrei.szabo](mailto:andrei.szabo@siemens.com), [joachim.bamberger](mailto:joachim.bamberger@siemens.com)}@siemens.com

<sup>3</sup> Helmut-Schmidt-University / University of the Federal Armed Forces Hamburg, Department of Electrical Engineering, Institute for Control Engineering, Holstenhofweg 85, 22043 Hamburg, Germany [joachim.horn@hsu-hh.de](mailto:joachim.horn@hsu-hh.de)

<sup>4</sup> Universität Karlsruhe, Fakultät für Informatik, Institut für Technische Informatik, Kaiserstr. 12, 76128 Karlsruhe [uwe.hanebeck@ieee.org](mailto:uwe.hanebeck@ieee.org)

**Summary.** Indoor positioning approaches based on communication systems typically use the received signal strength (RSS) as measurements. In order to work properly, such a system often requires many calibration points before its start. Based on theoretical propagation models (RF planning) and on self-organizing maps (SOM) an adaptive approach for Simultaneous Localization and Learning (SLL) has been developed. Applying SLL, a self-calibrating RSS-based positioning system with high accuracies can be realized without the need of cost intensive calibration measurements during system installation or maintenance.

The main aspects of SLL are addressed as well as convergence properties and statistical conditions for successful use. Results for real world DECT and WLAN setups are given, showing that the localization starts with a basic performance slightly better than Cell-ID, finally reaching the accuracy of pattern matching using calibration points.

## 1 Introduction

Localization is almost a synonym for the ubiquitous Global Positioning System (GPS). Within the car navigation systems GPS can successfully be applied [17]. Unfortunately GPS does not achieve the same accuracy in indoor or campus environments as in outdoor due to signal attenuation and multipath propagation. The indoor localization systems can be classified in systems using dedicated sensors and those that use existing infrastructure, as a communication system. In the first category fall many systems like those which make use of infrared beacons, e.g., the Active Badge location System [27],

ultrasound time of arrival, e.g., the Active Bat System [10] and the Cricket Location Support System [20], and received signal strength (RSS), e.g., the LANDMARC [16] with RFIDs. In the second category, there are systems based on GSM networks that combine the RSS with the time advance of mobiles terminals [12], use smart antennas to measure angles [14], or based on WLAN / DECT networks measuring the RSS [1, 2, 22, 25, 26] or measuring the propagation time [9]. There are still many other systems which use even contact sensors and images retrieved from cameras. Those systems are most suitable for robotics application.

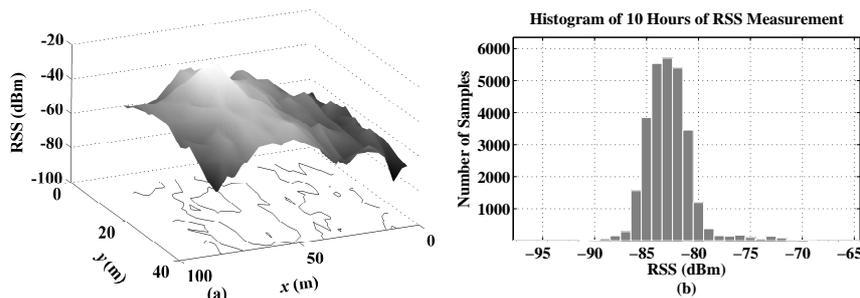
For localization systems based on communication systems, the computation of location out of measured features can be done using theoretical propagation models. The advantage of such solution is the low complexity due to the absence of pre-calibration. The disadvantage of such systems is that its accuracy is generally low, being more accurate only for special cases, e.g., measuring propagation time in Line of Sight (LOS) conditions [18, 19]. Unfortunately, this limits highly the solution applicability. Another approach is the measurement of RSS and use pattern matching on pre-calibration measurements. The achievable accuracy is suitable for localization of humans and objects, but the pre-calibration and maintenance costs are high.

In this chapter a unsupervised learning algorithm is developed. The Simultaneous Localization and Learning (SLL) avoids the requirement for manually obtained reference measurements using an initial propagation model with few parameters, which can be adapted by a few measurements, like the mutual measurements of the access points. Linear propagation models and more involved dominant path models incorporating map information are applied for the initialization. Thus, a feature map is obtained with the predicted RSS measurements at the map grid points. After the initialization the operating phase starts, which performs two tasks: localization and learning. Localization is done by pattern matching using the feature map. In addition, the RSS measurements are collected and used batch-wisely for learning. The learning is a refinement of the feature map, so that it corresponds better to the reality, therefore reducing the localization error. This new developed method uses a modified Kohonen's SOM learning algorithm. A closed form formulation for the algorithm, as well as algebraic and statistical conditions that need to be satisfied are given, deriving some convergence properties.

In Sect. 2 an overview of signal propagation models typically used for localization is given. Section 3 describes localization algorithms using the propagation models. The SLL approach is first presented in Sect. 4 and tested in different real world scenarios in Sect. 5.

## 2 RF Propagation Modelling

For RSS based indoor positioning system, it is important to know the RSS distribution accurately. Typically, two kinds of approaches are used: model



**Fig. 1.** (a) Spatial distribution of received power; (b) Temporal distribution of received power at a fixed point

based and pattern matching. The model based approach assumes that the propagation of radio signals follows a parameterized channel model. If the parameters are known, the RSS distribution can be predicted. In contrast, a pattern matching approach does not assume any prior information of radio propagation. It measures received power at various sample points to construct the radio distribution of the whole area. In the following these two categories of approaches are discussed.

## 2.1 Characteristics of the Indoor Propagation Channel

The propagation of Radio Frequency (RF) signals can be modelled at two scales. At large a scale the RSS decreases with the propagation distance. At a small scale, obstacles in the propagation path (due to walls, windows, cabinets and people), shadowing and multi-path effects cause a strongly varying RSS on small distances. Due to the complexity of indoor environments, this effect is unpredictable and typically modelled as a random variable.

Figure 1 shows an example of the spatial variation of the RSS in a real scenario. In subplot (a) it can be observed that the RSS has a decreasing trend, when the transmitter-receiver distance increases. This tendency is superimposed with different distortions effects due to shadowing and fading. In subplot (b) the time variation for the RSS measurement at a fixed position is shown.

## 2.2 Parametric Channel Models

The following models use a parameterized description of the radio channel. The parameters can be fit to a set of measurements so that the model is more accurate.

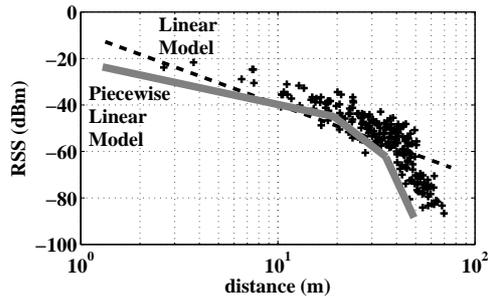


Fig. 2. Example of Linear Model and Piecewise Linear Model with 3 segments

### Linear Model (LM)

The theoretical propagation in free space follows a linear decay of the received power with distance on a logarithmic scale, [21]:

$$p_r = p_0 - 10 \cdot \gamma \cdot \log\left(\frac{d}{d_0}\right), \quad (1)$$

where  $p_r$  is the received power in dBm as a function of distance  $d$  and  $p_0$  is a constant term representing the received power in dBm at distance  $d_0$ . The model parameter  $\gamma$  is in free space equal with 2. In indoor environments  $\gamma$  typically has a value between 2 and 6.

There are two ways to obtain the parameters of the LM: One way is to use standard values found in literature. However, since radio properties in indoor scenarios vary a lot, this approach may cause a large model error. Another way is to use the values from several calibration points to tune the model. With  $m$  calibration points, the linear model can be defined as a system with  $m$  equations and two variables, the parameters  $p_0$  and  $\gamma$ , while  $d_0$  is fixed. These parameters can be estimated using least squares optimization [24].

### Piecewise Linear Model (PLM)

Another common model is the piecewise linear model (PLM) [8]. A PLM is fitted to calibration measurements with  $N$  segments, with slopes given by  $\{\gamma_1, \dots, \gamma_N\}$ , specified by  $N - 1$  breakpoints  $\{d_1, \dots, d_{N-1}\}$ . Different methods can be used to determine the number and location of breakpoints [8]. Once these are fixed, the slopes corresponding to each segment can be obtained by linear regression, similarly to the least square estimation used for the LM. An example of LM and PLM is shown in Fig. 2.

## 2.3 Geo Map Based Models

The following models use also a parameterized description of the radio channel. Additionally, information about the environment geometry, i.e., a map,

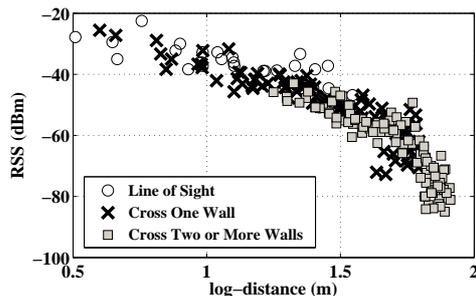


Fig. 3. Groups of Calibration Points with Different Number of walls

and physical properties are used with the measurements to fit the model parameters.

### Multi-Wall Model (MWM)

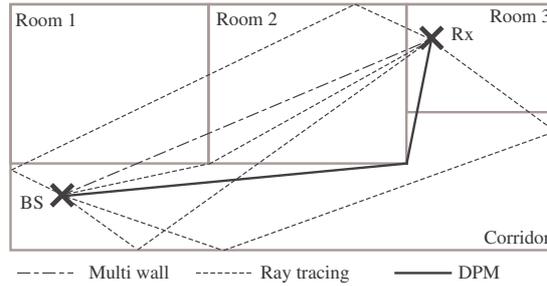
In indoor environments, walls are the major obstacles and they cause large attenuation of the RSS. In Fig. 3, the received signals in an indoor environment are plotted. Additionally, the number of walls between the transmitter and the receiver is shown. From the plot, we can see that at the same distance, the lower received power typically refers to a high number of walls between the transmitter and the receiver. To compensate the influence of walls, a new channel model based on the building layout is needed.

One such model is named Multi-Wall Model (MWM) or Motley-Keenan Model [21], described by the following equation:

$$p_r = p_0 - 10 \cdot \gamma \cdot \log \left( \frac{d}{d_0} \right) - \begin{cases} \sum WAF & \text{if } l < w \\ C & \text{if } l \geq w \end{cases}, \quad (2)$$

where  $\gamma$  and  $p_0$  have the same meaning as in the LM.  $WAF$  is Wall Attenuation Factor, which represents the partition value of the walls encountered by a single ray between the transmitter and the receiver.  $l$  is the number of walls encountered. The equation means that if the number of encountered walls is less than  $w$ , the power loss by walls can be computed as the sum of each  $WAF$ . And if the number of the encountered walls is more than  $w$ , the maximum path loss takes the value  $C$ .

The key parameter in the multi-wall model is the  $WAF$ . This parameter can either be set to a standard value or be estimated from the measurements at calibration points. A lot of experiments and simulations are made to get the value of  $WAF$  [6, 8, 15, 23]. These reported values differ due to the signal frequency, the type, the material, the shape and the position of the walls. Typically, some standard values, which average the results of several different experimental environments, are used in the modeling. A table with standard  $WAF$  values at 2GHz can be found at [21].



**Fig. 4.** Possible paths between a transmitter and a receiver

If there are calibration data available, it is possible to obtain the mean value of  $WAF$  from the real measurements by the regression estimation. Assuming uniform  $WAF$  for each wall, (2) can be expressed as:

$$p_r = p_0 - 10 \cdot \gamma \cdot \log \left( \frac{d}{d_0} \right) - \begin{cases} l \cdot WAF & \text{if } l < w \\ w \cdot WAF & \text{if } l \geq w \end{cases} \quad (3)$$

Using  $m$  calibration points, a system with  $m$  equations and three variables, the parameters  $p_0$ ,  $\gamma$  and  $WAF$ , can be resolved also by least square estimation, similarly to LM.

### Dominant Path Model (DPM)

The MWM only takes the direct ray between the transmitter and the receiver into consideration. This will lead to an underestimation in many cases. Ray tracing algorithms take into consideration every possible path between transmitter and receiver, requiring a high computation effort. In Fig. 4, the BS emits many possible rays to the receiver Rx. The direct path penetrates three walls and some paths bounce, reflect and refract until reach Rx, each of them with single contribution to the RSS that will be added by ray tracing. One path particularly has the strongest contribution among all others, presenting the optimal combination of distance, number of walls crossed and other attenuation factors. This is the dominant path.

The DPM [28] aims to find the dominant path, which could be a direct path as well, and use this single contribution to estimate the RSS. This overcomes the underestimation of only using the direct path and is considerably faster than ray tracing. Refs. [24, 28] give the algorithm to find the dominant path, which will not be described here.

### 2.4 Non-Parametric Models

Another radio map generation approach is based only on recording of calibration data. Considering the propagation profile as a continuous function,

then the calibration procedure is actually the sampling of this function at the positions specified for the radio map. The task can be understood as a regression problem, i.e., to estimate a continuous function from a finite number of observations.

The simplest way is to use a sampled radio map instead of a continuous one. This approach only takes the calibration points in consideration, neglecting other locations in the map. Therefore, estimated locations using matching algorithms are also restricted to these calibration points or the interpolation of them. Such radio map sampling is used in the pattern matching localization.

### 3 Localization Solution

Since the RSS vectors can be modelled as random variables, the statistical theory can be used to solve the matching problem. Being  $\mathbf{x}$  the location of a mobile terminal, and  $\mathbf{p} = [p_1, \dots, p_N]^T$  the vector with RSS from  $N$  BSs, where  $\{p_1, \dots, p_N\}$  are the respective RSS values from BS<sub>1</sub> to BS <sub>$N$</sub> . Then the probability of being at the true location  $\mathbf{x}$  given the measurement  $\mathbf{p}$  is expressed as  $\Pr(\mathbf{x}|\mathbf{p})$ , which can be written using the Bayesian rule as:

$$\Pr(\mathbf{x}|\mathbf{p}) = \frac{\Pr(\mathbf{p}|\mathbf{x}) \cdot \Pr(\mathbf{x})}{\int \Pr(\mathbf{p}|\mathbf{x}) \cdot \Pr(\mathbf{x}) d\mathbf{x}} \quad (4)$$

Usually the prior probability  $\Pr(\mathbf{x})$  is set as uniformly distributed, assuming that all positions are equally probable, hence independent of  $\mathbf{x}$ . Then (4) can be simplified as:

$$\Pr(\mathbf{x}|\mathbf{p}) = \frac{\Pr(\mathbf{p}|\mathbf{x})}{\int \Pr(\mathbf{p}|\mathbf{x}) d\mathbf{x}} \quad (5)$$

The conditional probability  $\Pr(\mathbf{p}|\mathbf{x})$  is determined from the data stored in the radio map. Using  $\Pr(\mathbf{x}|\mathbf{p})$  a location estimate of the of mobile terminal can be obtained using Bayesian inference.

There are two common methods for solving Bayesian estimation problems. The first one uses the Minimum Mean Variance Bayesian Estimator or Minimum Mean Square Error (MMSE) Estimator. This method gives an unbiased estimator with minimum mean square error. Considering  $\mathbf{x}$  as the true location and  $\hat{\mathbf{x}}$  as its estimate, the mean square error is written as  $Cost = \int \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 \cdot \Pr(\mathbf{x}|\mathbf{p}) d\mathbf{x}$ . Making  $\partial Cost / \partial \hat{\mathbf{x}} = 0$ , the best estimation for  $\mathbf{x}$  is given by:

$$\hat{\mathbf{x}} = E[\mathbf{x}|\mathbf{p}] = \int \mathbf{x} \cdot \Pr(\mathbf{x}|\mathbf{p}) d\mathbf{x}, \quad (6)$$

where  $E[\mathbf{x}|\mathbf{p}]$  is the expected value for  $\mathbf{x}$  given  $\mathbf{p}$ .

Another method uses the Maximum A Posteriori (MAP) estimator:

$$\hat{\mathbf{x}} = \arg \max_x \Pr(\mathbf{x}|\mathbf{p}) \quad (7)$$

From (5) and taking into account that  $\int \Pr(\mathbf{p}|\mathbf{x}) d\mathbf{x}$  is a normalizing constant, it follows that (7) can also be written as:

$$\hat{\mathbf{x}} = \arg \max_x \Pr(\mathbf{p}|\mathbf{x}), \quad (8)$$

which is also known as the Maximum Likelihood (ML) method.

In practice, an expression for  $\Pr(\mathbf{p}|\mathbf{x})$  is hard to model, so several simplifying assumptions are made: First, the measurements from several BSs are assumed independent so that  $\Pr(\mathbf{p}|\mathbf{x})$  can be factorized. Further on, it is assumed that the measurements follow a Gaussian distribution. Then:

$$\Pr(\mathbf{p}|\mathbf{x}) = \prod_{n=1}^N \frac{1}{\sigma_n \sqrt{2\pi}} \exp\left(-\frac{(p_n - \bar{p}_n)^2}{2\sigma_n^2}\right), \quad (9)$$

where  $\bar{p}_n$  is the mean RSS from BS<sub>*n*</sub> at position  $\mathbf{x}$ , and  $\sigma_n$  is the standard deviation at this position.

With these assumptions, the Maximum Likelihood solution becomes:

$$\hat{\mathbf{x}} = \arg \min_x \sum_{n=1}^N \left( \ln \sigma_n + \frac{(p_n - \bar{p}_n)^2}{2\sigma_n^2} \right) \quad (10)$$

If the standard deviation  $\sigma_n$  for every position is constant, then (10) reduces to:

$$\mathbf{x} = \arg \min_x \sum_{n=1}^N (p_n - \bar{p}_n)^2 \quad (11)$$

This equation describes the method known as Nearest Neighbor (NN), which compares the distances in signal space between an input vector (here as  $\mathbf{p}$ ) and each stored data (here the recorded radio map calibration data). The nearest stored data to the input, i.e., with smallest distance in signal space, is chosen as best match and the position of this data is returned as the estimation for  $\mathbf{x}$ . The advantage of this algorithm is that it does not require any statistical prior knowledge of the calibration data since only the mean is recorded. However, due to this simplicity, the accuracy is degraded if the hypothesis of constant  $\sigma$  does not correspond to the measurements.

## 4 Simultaneous Localization and Learning

In order to work with a reasonable accuracy, pattern matching often requires many measurements as a way to build a detailed feature map. Thus, the calibration effort prior to system start (offline phase) is usually very time consuming and expensive. The collection of measurements is labelled with the true location where they were taken until the samples are significant enough to represent the desired feature.

For this reason, the research aiming at so called calibration free systems has risen rapidly in the past few years. In [5] a first version of a new algorithm was presented to reduce the calibration effort significantly: the Simultaneous Localization and Learning (SLL). The SLL is a modified version of the Kohonen Self Organizing Map (SOM) [13] that can straightforwardly be used for RSS-based localization in environments with already available infrastructure as WLAN or DECT.

#### 4.1 Kohonen SOM

SOMs are a special class of neural networks, which are based on competitive learning. In a SOM, the neurons are placed at the nodes of a lattice that is typically one- or two-dimensional. The neurons become selectively adapted to various input patterns in the course of a competitive learning process. The locations of the adapted, i.e., winning neurons become ordered with respect to each other in such a way that a meaningful coordinate system for different input feature is created over the lattice [13]. A SOM is therefore characterized by the formation of a topological map of input patterns in which the spatial locations of the neurons are indicative of intrinsic statistical features contained in the input patterns [11].

The principal goal of Kohonen's SOM is to transform an incoming signal pattern of arbitrary dimension into a one- or two-dimensional (1D or 2D) discrete map, and to perform this transformation adaptively in a topologically ordered fashion. This is achieved by iteratively performing three steps in addition to the initialization: competition, cooperation and adaptation. During the initialization the synaptic weights in the neural network are set, typically using a random number generator. In the competitive step the winning neuron  $i$  with the weight vector  $w_i = [w_{i1}, \dots, w_{in}]$  in the  $n$  dimensional input space shows the smallest cost with respect to a given input feature vector  $\xi = [\xi_1, \dots, \xi_n]^T$ , that is,  $i = \arg \min\{|\xi - w_j|\}$ , with the index  $j$  going through all neurons in the lattice. The winning neuron will be the center for the adaptation process.

The cooperation determines which neurons will be adapted together with the winning neuron  $i$ . A neighbourhood function  $h_{ij}(k)$ , dependent on the discrete time step  $k$ , is used to find the neuron  $j$  close to the winner and to weigh it accordingly with the distance to the winner in the lattice. The amount of adaptation decreases monotonically with distance from the center neuron, i.e., the winner.

A typical choice for the neighbourhood function at 1D problems is the constant function, set to 1 for the winner and for an equal number of neighbours, forward and backward (usually just 2 neighbours are taken). For two or three dimensional maps the Gaussian function is usually chosen, so that:

$$h_{ij}(k) = \eta(k) \cdot \exp\left(-\left(\frac{d_{ij}}{2 \cdot \sigma(k)}\right)^2\right), \quad (12)$$

where  $\eta(k)$  is the learning rate,  $\sigma(k)$  is the effective width of the topological neighbourhood, both dependent on  $k$ .  $d_{ij}$  is the distance from neuron  $j$  to neuron  $i$  at the center. The adaptation law, given by

$$w_j(k+1) = w_j(k) + h_{ij}(k) \cdot (\xi(k) - w_j(k)), \quad (13)$$

ensures that the response of the winning neuron to the subsequent application of a similar input pattern is enhanced [27].

The adaptive process consists of two phases: the self-organizing or ordering phase and the convergence phase. In the ordering phase the topological ordering of the weight vectors takes place. During this phase the learning rate and the neighbourhood area should decrease. The neighbourhood area goes from complete coverage to a few neurons or even to the winning neuron itself. In the convergence phase the fine tuning of the feature map takes place in order to provide an accurate statistical quantification of the input space. The learning rate should stay constant or it could decay exponentially [11].

The Kohonen algorithm is surprisingly resistant to a complete mathematical study (cf. [7]). The only thorough analyses could be achieved for 1D case in a linear network. For higher dimensions, the results are only partial.

## 4.2 Main Algorithm

The SLL is an iterative algorithm that can be understood as the following scenario describes: a measurement is taken and then used to locate a user, based on a coarse feature map. The found location is then used as center for a learning step, where the neighbourhood surrounding this center is adapted towards the measurement. These operations continue repeatedly at each new measurement, improving the feature map.

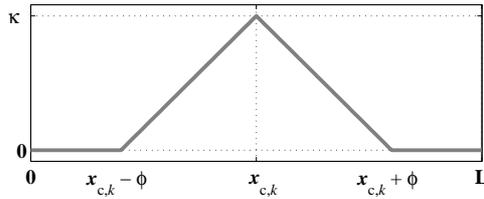
The modelling used for the SLL originally assumed that the feature map contained the mean value of RSS of each BS as a function of position. This feature map was then termed radio map [5]. However, the SLL is by no means constrained to this feature only. Other features with spatial gradients, like propagation times or angles of arrival, could be used as well.

The model  $\mathbf{p}_k(\mathbf{x})$  describes the RSS propagation through space at the discrete time  $k$ . The dimension of  $\mathbf{p}_k(\mathbf{x})$  defines the number of BSs considered for the localization scenario.  $\mathbf{x}$  defines a position in some fixed reference frame.  $\mathbf{p}_0(\mathbf{x})$  represents the initial model at the initial time  $k = 0$ . The measurement  $\mathbf{p}_M$  is associated to  $\mathbf{x}_M$ , a (not necessarily known) measurement position.  $\mathbf{p}_{M,k}$  is the measurement taken at the discrete time  $k$ .

Starting from this model, the SLL is defined by the following feedback or update law (implicitly described in [5]):

$$\mathbf{p}_{k+1}(\mathbf{x}) = \mathbf{p}_k(\mathbf{x}) + f_{c,k+1} \cdot (\mathbf{p}_{M,k+1} - \mathbf{p}_k(\mathbf{x})), \quad (14)$$

where  $f_{c,k} = f(\mathbf{x}_{c,k}, \mathbf{x}, \kappa, \phi)$  is a function of the centering position  $\mathbf{x}_{c,k}$  at time  $k$ , of  $\mathbf{x}$ , and of the SLL control variables  $\kappa$  and  $\phi$ .  $f_{c,k}$  spatially



**Fig. 5.** Weighting function  $f_{c,k}$

bounds and weights the update based on the difference between the actual measurement  $\mathbf{p}_{\mathbf{M},k+1}$  and the present model  $\mathbf{p}_k(\mathbf{x})$ .

$f_{c,k}$  can have different forms, like a polynomial or Gauss distribution. Its important characteristics is that it is symmetric around  $\mathbf{x}_{c,k}$  and has its magnitude bounded by the interval  $[0; \kappa]$ , with  $\kappa < 1$ . If the distance from  $\mathbf{x}$  to  $\mathbf{x}_{c,k}$  is greater than  $\phi$  then  $f_{c,k} = 0$ . The function  $f_{c,k}$  reaches its maximum at  $\mathbf{x}_{c,k}$  with value  $\kappa$ , and falls to smaller values until the boundary defined by  $\phi$ . The location  $\mathbf{x}_{c,k} = \mathbf{x}_{c,k}(\mathbf{p}_{\mathbf{M},k})$  corresponds to the measurement  $\mathbf{p}_{\mathbf{M},k}$ . The determination of  $\mathbf{x}_{c,k}$  depends on the localization technique chosen (see Sect. 3). Fig. 5 shows a qualitative 1D example for the function  $f_{c,k}$  and the meaning of the function arguments.

### 4.3 Comparison between SOM and SLL

It is noteworthy the resemblance between (13) and (14); in fact the update law for SLL is the same as for SOM. However, the SOMs exist over a discrete lattice with a finite number of neurons, as the SLL can work in a continuous space (in which case the concept for a single neuron vanishes, since there are infinite neurons). The neighbourhood function  $h_{ij}$  becomes the weighting function  $f_{c,k}$  and the input vector  $\xi$  becomes the measurement vector  $\mathbf{p}_{\mathbf{M}}$ .

The SLL starts with a coarse initial model, which at least presents some plausible physical property (for example, a radio map where the maximum RSS is placed at the BS position and decays with increasing distance). This ensures that the initial model can be used for localization queries, even if it is not very accurate. Proceeding with the analogy to SOM, the initial model represents an already ordered state, as the feature map is never initialized with random values, therefore only the cooperation and convergence phases are of interest for the SLL.

The localization has also an important role for the SLL. Since it determines the location where the update will be made, if the localization delivers a  $\mathbf{x}_{c,k}$  very far from the (usually) unknown measurement position, the SLL will fail to improve the initial model.

In [3, 4] some algebraic properties as well as the statistical conditions for a successful use of SLL were presented. The analysis was then focused on the 1D problem. Now some generalizations for more dimensions are presented.

#### 4.4 Convergence Properties of SLL

The recursive SLL formula given in (14) can be written in closed form:

$$\mathbf{p}_k(\mathbf{x}) = \mathbf{p}_0(\mathbf{x}) + \sum_{j=1}^k \left( f_{c,j} \cdot (\mathbf{p}_{\mathbf{M},j} - \mathbf{p}_0(\mathbf{x})) \prod_{i=j+1}^k (1 - f_{c,i}) \right), \quad (15)$$

with

$$\prod_{i=a}^b (\cdot) = 1, \quad \forall a > b, \quad (16)$$

as it has been proved in [3]. Defining the utility functions:

$$\mathbf{F}_k = \sum_{j=1}^k \left( f_{c,j} \prod_{i=j+1}^k (1 - f_{c,i}) \right), \quad (17)$$

and

$$\mathbf{P}_{\mathbf{M},k} = \sum_{j=1}^k \left( f_{c,j} \cdot \mathbf{p}_{\mathbf{M},j} \prod_{i=j+1}^k (1 - f_{c,i}) \right). \quad (18)$$

Then (15) can be compactly written as:

$$\mathbf{p}_k(\mathbf{x}) = \mathbf{p}_0(\mathbf{x}) - \mathbf{p}_0(\mathbf{x})\mathbf{F}_k + \mathbf{P}_{\mathbf{M},k}. \quad (19)$$

#### Limit Value

In [3, 4] it has been proved that  $\lim_{k \rightarrow \infty} \mathbf{F}_k = 1$  inside a closed interval for the one dimensional case. This result can be extended to higher dimensions, as long as the weighting function  $f_{c,k}$  has the properties explained in the last section and that the series  $\mathbf{x}_{c,k}$ , where the  $f_{c,k}$ s are centered, cover repeatedly the entire closed interval.

$\mathbf{P}_{\mathbf{M},k}$ , in contrast to  $\mathbf{F}_k$ , cannot reach a steady state. Each measurement  $\mathbf{p}_{\mathbf{M},j}$ , as it appears in (18), pushes  $\mathbf{P}_{\mathbf{M},k}$  towards  $\mathbf{p}_{\mathbf{M},j}$  inside the hypersphere centered in  $\mathbf{x}_{c,j}$  and with radius  $\phi$ . Since the measurements vary through space,  $\mathbf{P}_{\mathbf{M},k}$  continuously changes.

In this way, when  $k \rightarrow \infty$ ,  $\mathbf{p}_k(\mathbf{x})$ , as given in (19) tends to:

$$\lim_{k \rightarrow \infty} \mathbf{p}_k(\mathbf{x}) = \mathbf{p}_0(\mathbf{x}) - \mathbf{p}_0(\mathbf{x}) + \lim_{k \rightarrow \infty} \mathbf{P}_{\mathbf{M},k} = \lim_{k \rightarrow \infty} \mathbf{P}_{\mathbf{M},k} \quad (20)$$

This shows an important result of SLL: Eventually, the initial model  $\mathbf{p}_0(\mathbf{x})$  will be replaced entirely by  $\mathbf{P}_{\mathbf{M},k}$ , a term that depends on the measurements and on the location estimation. Since  $\mathbf{p}_0(\mathbf{x})$  disappears with increasing iterations, there is no need to make the initial model extremely precise. It suffices to start with a coarse and hence relatively simple model, e.g. linear model. The

requirement for feature map initialization is a reasonable location estimation. Another effect of SLL is that old measurements have a smaller contribution to  $\mathbf{P}_{\mathbf{M},k}$  than newer ones. This can clearly be seen in (18), where the  $\mathbf{p}_{\mathbf{M},j}$ s are multiplied by products of  $(1 - f_{c,i}) \leq 1 \forall i$ . The older the measurements are, the bigger is the number of terms in the product, which will tend to zero. The consequence is that the model is always updated, as long as new measurements are considered and as the control parameters are non-zero.

### Measurement noise

Assuming that each measurement  $\mathbf{p}_{\mathbf{M}}$  is corrupted by stationary Gaussian noise  $\zeta(\mathbf{x})$  with mean  $\boldsymbol{\mu}(\mathbf{x})$  and variance  $\boldsymbol{\sigma}^2(\mathbf{x})$ , it is desirable to know the remaining effect of this noise after some iterations of the SLL.

Returning to (14), the recursive equation regarding the noise  $\zeta_{k+1} = \zeta(\mathbf{x}_{\mathbf{M},k+1})$  at the new measurement position  $\mathbf{x}_{\mathbf{M},k+1}$  becomes:

$$\mathbf{p}_{k+1}(\mathbf{x}) = \mathbf{p}_k(\mathbf{x}) + f_{c,k+1} \cdot (\mathbf{p}_{\mathbf{M},k+1} + \zeta_{k+1} - \mathbf{p}_k(\mathbf{x})), \quad (21)$$

which, similarly to (15), leads to the closed form:

$$\mathbf{p}_k(\mathbf{x}) = \mathbf{p}_0(\mathbf{x}) + \sum_{j=1}^k \left( f_{c,j} \cdot (\mathbf{p}_{\mathbf{M},j} + \zeta_j - \mathbf{p}_0(\mathbf{x})) \prod_{i=j+1}^k (1 - f_{c,i}) \right) \quad (22)$$

The noise term can be separated from (22) defining the utility function

$$\mathbf{Z}(\mathbf{x}, \mathbf{x}_{\mathbf{M},1:k}, \mathbf{x}_{\mathbf{c},1:k}) = \mathbf{Z}_k = \sum_{j=1}^k \left( f_{c,j} \cdot \zeta_j \prod_{i=j+1}^k (1 - f_{c,i}) \right), \quad (23)$$

such that the following short form is attained using (17) and (18):

$$\mathbf{p}_k(\mathbf{x}) = \mathbf{p}_0(\mathbf{x}) - \mathbf{p}_0(\mathbf{x})\mathbf{F}_k + \mathbf{P}_{\mathbf{M},k} + \mathbf{Z}_k, \quad (24)$$

which corresponds to (19) with the extra term  $\mathbf{Z}_k$  modelling the influence of the measurement noise. It is important to note that  $\mathbf{Z}_k$  depends not only on the considered location  $\mathbf{x}$ , but also on the sequence of true measurement locations, defined by  $\mathbf{x}_{\mathbf{M},1:k} = \{\mathbf{x}_{\mathbf{M},1}, \dots, \mathbf{x}_{\mathbf{M},k}\}$  as well as on the sequence of estimated locations, defined by  $\mathbf{x}_{\mathbf{c},1:k} = \{\mathbf{x}_{\mathbf{c},1}, \dots, \mathbf{x}_{\mathbf{c},k}\}$ , where the weighting functions are centered.

The similarity between  $\mathbf{P}_{\mathbf{M},k}$  in (18) with  $\mathbf{Z}_k$  in (23) is notable. They differ only in the scalar term introduced with new iterations: i.e.,  $\mathbf{p}_{\mathbf{M},j}$  and  $\zeta_j$ , respectively.

$\mathbf{Z}_k$  cannot reach a steady state for the same reason as  $\mathbf{P}_{\mathbf{M},k}$ . However, departing from the assumption that each  $\zeta_j$  is an independent Gaussian random variable, it is possible to calculate expectations of mean and variance of  $\mathbf{Z}_k$  based on the mean  $\boldsymbol{\mu}_j$  and variance  $\boldsymbol{\sigma}_j^2$  of each  $\zeta_j$ .

For one particular fixed point  $\mathbf{x}_f$ , (23) shows that  $\mathbf{Z}(\mathbf{x}_f, \mathbf{x}_{M,1:k}, \mathbf{x}_{c,1:k}) = \mathbf{Z}_k(\mathbf{x}_f)$  is formed as a weighted sum of random variables. And therefore, the following properties for linear operations on independent random variables can be used, provided that  $a$  and  $b$  are scalars:

$mean\{a + b\zeta_i\} = a + b\mu_i$	$mean\{\zeta_i + \zeta_j\} = \mu_i + \mu_j$
$var\{a + b\zeta_i\} = b^2\sigma_i^2$	$var\{\zeta_i + \zeta_j\} = \sigma_i^2 + \sigma_j^2$

In this way, using the recursive formulation for (23):

$$\mathbf{Z}_{k+1}(\mathbf{x}_f) = \mathbf{Z}_k(\mathbf{x}_f) \cdot (1 - f_{c,k+1}) + f_{c,k+1} \cdot \zeta_{k+1}, \quad (25)$$

setting  $mean\{\mathbf{Z}_k(\mathbf{x}_f)\} = \mathbf{M}_{\mathbf{Z}}(\mathbf{x}_f, \mathbf{x}_{M,1:k}, \mathbf{x}_{c,1:k}) = \mathbf{M}_{\mathbf{Z},k}(\mathbf{x}_f)$  and  $var\{\mathbf{Z}_k(\mathbf{x}_f)\} = \mathbf{S}_{\mathbf{Z}}^2(\mathbf{x}_f, \mathbf{x}_{M,1:k}, \mathbf{x}_{c,1:k}) = \mathbf{S}_{\mathbf{Z},k}^2(\mathbf{x}_f)$ , and using the properties above listed, it is possible to express the mean of  $\mathbf{Z}_k(\mathbf{x}_f)$  recursively as:

$$\mathbf{M}_{\mathbf{Z},k+1}(\mathbf{x}_f) = \mathbf{M}_{\mathbf{Z},k}(\mathbf{x}_f) \cdot (1 - f_{c,k+1}(\mathbf{x}_f)) + f_{c,k+1}(\mathbf{x}_f) \cdot \boldsymbol{\mu}(\mathbf{x}_{M,k+1}), \quad (26)$$

and similarly for its variance as:

$$\mathbf{S}_{\mathbf{Z},k+1}^2(\mathbf{x}_f) = \mathbf{S}_{\mathbf{Z},k}^2(\mathbf{x}_f) \cdot (1 - f_{c,k+1}(\mathbf{x}_f))^2 + f_{c,k+1}^2(\mathbf{x}_f) \cdot \boldsymbol{\sigma}^2(\mathbf{x}_{M,k+1}) \quad (27)$$

Comparing (26) with the recursive formulation for  $F_k$  in [3], it follows that (26) has also the form of an exponential filter with the variable parameter  $f_{c,k+1}(\mathbf{x}_f)$  and with the variable input  $\boldsymbol{\mu}(\mathbf{x}_{M,k+1})$ .  $\lim_{k \rightarrow \infty} \mathbf{M}_{\mathbf{Z},k}(\mathbf{x}_f) = \boldsymbol{\mu}$  holds for constant  $\boldsymbol{\mu}(\mathbf{x}_{M,k}) = \boldsymbol{\mu} \forall k$ , constant measurement location  $\mathbf{x}_M$ , and constant estimated location  $\mathbf{x}_c$ .

Notwithstanding the similarity between (26) and (27), the latter cannot be treated as an exponential filter due to its quadratic terms. Even if  $\boldsymbol{\sigma}^2$  is constant in all space,  $\mathbf{S}_{\mathbf{Z},k}^2(\mathbf{x})$  will vary according to the sequence of  $\mathbf{x}_{c,s}$ . However,  $\mathbf{S}_{\mathbf{Z},k}^2(\mathbf{x})$  is upper-bounded by a maximum value. This maximum can be estimated considering a constant update center, i.e.,  $\mathbf{x}_{c,k} = \mathbf{x}_c$  for all  $k$  and assuming space-invariant and therefore also time-constant noise, i.e.,  $\boldsymbol{\sigma}^2(\mathbf{x}) = \boldsymbol{\sigma}^2$  and  $\boldsymbol{\mu}(\mathbf{x}) = \boldsymbol{\mu}$ .

Since  $\mathbf{x}_{c,k}$  is constant in time, so is  $f_{c,k} = f_c$  too for all  $k$  (assuming that neither  $\kappa$  nor  $\phi$  vary with time). The recursive equation for  $\mathbf{S}_{\mathbf{Z},k+1}^2$  can be written as:

$$\mathbf{S}_{\mathbf{Z},k+1}^2 = \mathbf{S}_{\mathbf{Z},k}^2 \cdot (1 - f_c)^2 + f_c^2 \cdot \boldsymbol{\sigma}^2 \quad (28)$$

Assuming a steady state, i.e.,  $\mathbf{S}_{\mathbf{Z},k+1}^2 = \mathbf{S}_{\mathbf{Z},k}^2 = \mathbf{S}_{\mathbf{Z},steady}^2$ ,

$$\mathbf{S}_{\mathbf{Z},steady}^2 = \frac{f_c^2 \cdot \boldsymbol{\sigma}^2}{1 - (1 - f_c)^2} = \frac{f_c \cdot \boldsymbol{\sigma}^2}{2 - f_c}, \quad (29)$$

holds. In particular at the position  $\mathbf{x} = \mathbf{x}_f = \mathbf{x}_c$ :

$$\mathbf{S}_{\mathbf{Z},steady}^2(\mathbf{x}_c) = \frac{\kappa \cdot \boldsymbol{\sigma}^2}{2 - \kappa}, \quad (30)$$

which is the maximum for this function.

Considering that  $\kappa \in [0; 1]$ , and that the upper bound is given by (30), it is easy to verify that  $\mathbf{S}_{\mathbf{Z}, steady}^2 \leq \sigma^2$ . This indicates that the variance of  $\mathbf{Z}_k$  at one particular position will be at most  $\sigma^2$ , and that only if  $\kappa = 1$ .

The important result is the noise reduction property of SLL: by exponential filtering and spatial weighting due to  $f_c$ , the variance of the learned radio map is reduced. At one particular position  $\mathbf{x}_f$ , this noise averaging is achieved not only using the single measurements at  $\mathbf{x}_f$ , but also using the noisy measurement of neighbouring positions (e.g. the measurement sequence  $\mathbf{x}_{M, 1:k}$ ).

### Limit Area for Perfect Localization

The concept of limit area appears if only one dimension in space is considered and with perfect localization. The extension of this result in more dimensions would result in limit hyper volumes and will not be treated here.

Perfect localization implies that, for a given measurement  $p_{M,k}$ , the associated position  $x_{c,k}$  corresponds exactly to the real measurement position  $x_{M,k}$ , i.e.,  $x_{c,k} = x_{M,k}$ . The measurements follow a propagation law  $g$ :

$$p_M(x_M) = g(x_M, p_{out}, \gamma), \quad (31)$$

where  $p_{out}$  is the output power of a BS and  $\gamma$  is an attenuation factor. The signal propagation is assumed to be monotonic, i.e.,  $\partial g / \partial x$  is always negative. Without loss of generality, the BS is assumed to be placed on the left side of the radio map such that the measurement at  $x_f - \phi$  is:

$$p_M(x_f - \phi) = p_{M+} = g(x_f - \phi, p_{out}, \gamma), \quad (32)$$

being  $x_f$  a fixed position and the measurement at  $x_f + \phi > x_f - \phi$  is:

$$p_M(x_f + \phi) = p_{M-} = g(x_f + \phi, p_{out}, \gamma) < p_{M+}, \quad (33)$$

A perfect initialization means that the starting radio map at the instant  $k = 0$  has exactly the same propagation profile as the measurements. Hence, the start model at some position  $x_f$  is:

$$p_0(x_f) = g(x_f, p_{out}, \gamma) \quad (34)$$

As the weighting function  $f_c$  has a bounded support, the measurement positions  $x_M$  which can change the radio map at the considered position  $x_f$  belong to the interval  $[x_f - \phi; x_f + \phi]$ . Hence, there are two measurement cases to consider:

If  $x_M$  lies outside of the support of the weighting function  $f$ , i.e.,  $x_M \notin (x_f - \phi; x_f + \phi)$ , then  $f_{c,k+1}(x_f) = 0$  and the update law in (14) results in  $p_{k+1}(x_f) = p_k(x_f)$ . In this case, at  $x_f$  no update is made.

If  $x_M \in (x_f - \phi; x_f + \phi)$  then  $f_{c,k+1}(x_f) \in (0; \kappa]$  and the update will cover  $x_f$ . Accordingly, the update law reduces the difference between the radio map

at  $x_f$  and the current measurement:  $|p_{k+1}(x_f) - p_{M,k+1}| \leq |p_k(x_f) - p_{M,k+1}|$ . Due to perfect initialization, for  $k = 0$  the model will stay unchanged at  $x_f$  only if the measurement equals the radio map, i.e.,  $p_M = p_0(x_f)$  and  $x_M = x_f$ . In this case no improvement can be accomplished since the radio map is already perfect. For all other measurement values in the interval  $[p_{M+}; p_{M-}]$  the model will be disturbed at  $x_f$  towards the actual measurement. Since  $p_{M,k+1} \in [p_{M+}; p_{M-}]$  and considering the assumed monotony,  $p_{k+1}(x_f)$  is also limited by this interval. The maximal positive disturbance at  $x_f$  is given by  $p_{M+}$  and the maximal negative disturbance is given by  $p_{M-}$ .  $p_{M+}$  or  $p_{M-}$  can be obtained, if and only if measurements are repetitively taken either at  $x_f - \phi$  or at  $x_f + \phi$ , respectively.

Since there is no assumption on the placement of  $x_f$ , this result can be generalized to the whole radio map interval  $[0; L]$ . Thus an upper and lower limit curve can be defined for the whole interval  $[0; L]$ , where  $p_k(x)$  is not expected to cross. The radio map is restricted to a limit area, which is defined by moving the true propagation model by  $\phi$  to the left and to the right:

$$g(x + \phi) \leq p_k(x) \leq g(x - \phi) \quad (35)$$

If the measurements are not taken repeatedly at the same position, i.e., any position inside the interval  $[0; L]$  could be taken, then the radio map  $p_k(x)$  still stays inside the defined limit area.

Relaxing the initialization constraint, i.e., allowing an arbitrary wrong initial model, it can be shown that with perfect localization and sufficient iterations the model can be brought in finite time to the limit area.

The starting model can, e.g., be defined as:

$$p_0(x) = g(x, p_{\text{out}} + \Delta p_{\text{out}}, \gamma + \Delta\gamma), \quad (36)$$

where  $\Delta p_{\text{out}}$  and  $\Delta\gamma$  are offsets on the output power and on the attenuation factor, respectively. Considering  $x_f$ , there are again two measurement cases to be looked at. Case 1 behaves as with perfect initialization: if  $x_M$  lies outside of the support of the weighting function no update is made.

Considering the measurement case 2, the inequality  $|p_{k+1}(x_f) - p_{M,k+1}| \leq |p_k(x_f) - p_{M,k+1}|$  still holds, but now  $p_0(x_f)$  is arbitrary wrong and a finite number of iterations is required in order to bring the radio map at  $x_f$  into the limit area, defined before. In order to reach a radio map that is inside the limit area, all positions on the radio map must be sufficiently updated and therefore the measurements must be taken to cover the complete area.

Assuming that the initial radio map lies completely outside the limit area and that the considered fixed position  $x_f$  is given by  $x_M$ , i.e.,  $x_f = x_M \forall k$ , so that  $p_M(x_f)$  is constant, then applying (36) on (19) at  $x_f$  gives:

$$p_k(x_f) = g(x_f, p_{\text{out}} + \Delta p_{\text{out}}, \gamma + \Delta\gamma) \cdot (1 - F_k) + p_M(x_f) \cdot F_k \quad (37)$$

Due to the constant measurement positions and the exponential filter form of  $F_k$  [3], the following relation holds:

$$F_k(x_f) = 1 - (1 - \kappa)^k \quad (38)$$

Hence, the influence of the initialization will decrease according to  $(1 - \kappa)^k$  and the measurement term will reach  $p_M(x_f)$  according to  $1 - (1 - \kappa)^k$ . In finite time  $p_k(x_f)$  will reach the bounded region defined by  $p_{M+}$  and  $p_{M-}$ .

Assuming a linear propagation model, making  $p_M(x_f) = p_{out} - \gamma \cdot x_f$ , making  $g(x_f, p_{out} + \Delta p_{out}, \gamma + \Delta \gamma) = p_{out} + \Delta p_{out} - (\gamma + \Delta \gamma) \cdot x_f$  and setting  $p_k(x_f)$  at the bounded region as  $p_{out} - \gamma x_f + \gamma \phi$ , then substituting these terms in (37) an inequality can be written:

$$(1 - \kappa)^k \leq \gamma \phi / (\Delta \gamma x_f - \Delta p_{out}), \quad (39)$$

which can be used to determine the required convergence time  $k$ .

The analytical investigations show that both SLL parameters should be small:  $\phi$  should be small to enable a tight limit area;  $\kappa$  should be small to reduce noise effects. Otherwise the larger the two parameters are, the faster (large  $\kappa$ ) and wider (large  $\phi$ ) the radio map learning is. A good trade-off between accuracy and speed could be achieved by starting with larger parameters reducing them over time.

#### 4.5 Statistical Conditions for SLL

In this section the influence of real localization on the performance of SLL is investigated. After definition of real localization, statistical conditions for convergence towards the analytical bound are identified using simulations. The statistical conditions are confirmed by real world experiments.

#### Real Localization

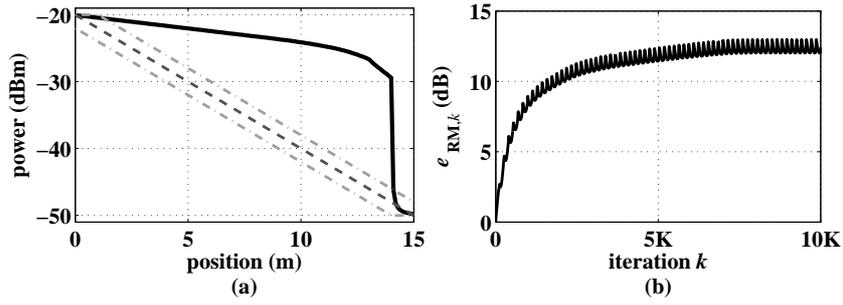
When the localization is not perfect, then the estimated position  $x_c$ , where the update function  $f_c$  is centered, is not guaranteed anymore to be the measurement position  $x_M$ .

Now  $x_{c,k}$  is estimated using the measurement  $p_{M,k}$  and a NN search (cf. Sect. 3) on  $p_{k-1}(x)$ , looking for the best  $x = x_{c,k}$  for whom  $p_{k-1}(x_{c,k})$  is the closest match for  $p_{M,k}$ . For the simulations the NN algorithm is run on the quantized 1D space comprised in the interval  $[0; L]$ .

#### Simulations

In [5] first results with SLL applied to real world data have been shown. In [3,4] an analysis was performed in order to reveal the statistical conditions that need to be satisfied to reliably obtain a good result. For that, some experiments were defined in order to determine those statistical conditions:

The experiments performed differ with respect to the distribution and the order of measurements. From the analytical considerations it is known that



**Fig. 6.** Radio map learning for sequentially ordered measurements. The distance between two succeeding measurement positions is smaller than  $\phi$

both SLL parameters,  $\kappa$  and  $\phi$ , should be small at the end of learning time to achieve a high accuracy and larger at the beginning to achieve a fast learning. To avoid a superposition of the statistical effects here investigated with effects possibly introduced by time-varying parameters,  $\kappa$  and  $\phi$  are set to fixed values and kept constant over time.

The radio map is defined on the interval  $[0; 15]$  meters. The distance  $\Delta x$  between consecutive positions on the discrete radio map is 0.1m. The learning parameter  $\kappa$  is set to 0.5. The measurement positions  $x_M$  are uniformly distributed over space. The plots show the radio map at the initialization (dashed lines) and at the end of simulation (bold lines), the measurements (thin line or circles) and the limit area (dash dotted lines), with power as a function of position. The plots also show the radio map error, defined as the RMS error between the measurements (as labelled data and taken at once) and the actual radio map at some instant  $k$ :

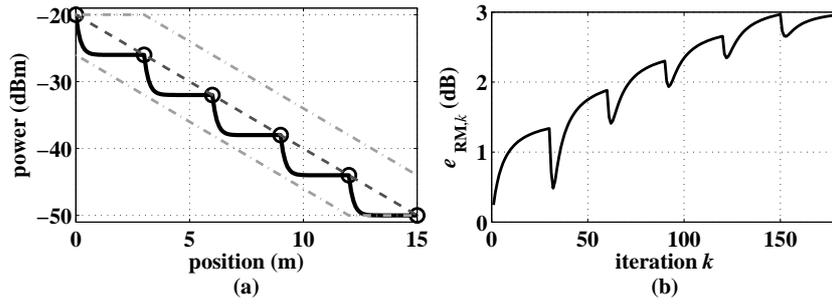
$$e_{RM,k} = \sqrt{(p_k(x) - p_M(x))^T (p_k(x) - p_M(x)) / m}, \quad (40)$$

with  $m$  as the number of reference positions on the radio map.

Fig. 6 (a) and (b) depicts the experiment 1. The distance  $\Delta x_M$  between two succeeding measurement positions is 0.1m, which results in 150 measurements.  $\phi$  is set to 1m, which is bigger than  $\Delta x_M$ . The initial radio map, as well as the measurements are given by the linear equation  $p_0(x) = p_{out} - \gamma \cdot x$ , with  $p_{out} = -20$  dBm,  $\gamma = 2$  dBm/m (since the initial radio map and measurements are coincident, only the dashed line is plotted).

The measurements are sequentially ordered, taken at increasing coordinate values, i.e., from 0 to  $L$ . The final radio map is given by the solid line after the ordered sequence of measurements is cyclically used 70 times, which results in 10570 iterations. The black line shows the final radio map. Noteworthy is that the slope of the final radio map depends directly on  $\Delta x_M$ .

$e_{RM,k}$  departs from 0, as the initialization is perfect and increases until a steady state is reached. The small oscillations visible in the radio map error are caused by the periodicity of the measurement locations. An equivalent



**Fig. 7.** Radio map learning for sequentially ordered measurements. The distance between two succeeding measurement positions is equal to  $\phi$

and complementary experiment, with the sequence of positions going from  $L$  to 0 was shown in [3].  $e_{RM}$  for that case was exactly the same and the same properties were verified.

An important feature of SLL, called edge effect, is shown in this figure. Due to the sequence of narrowing close positions, almost the entire radio map lay outside the theoretical limit area. However, because of the limitation of the localization space between 0 and  $L$  together with the monotonicity of the propagation function, the strongest RSS will be learned at  $x = 0$  and the weakest RSS will be learned at  $x = L$ , even for real localization. If the localization tries to locate a measurement outside the given interval, the estimated position is set to the corresponding edge.

Fig. 7 (a) and (b) shows the experiment 2, which has a slightly different setup as the first one. Here, the distance between measurement positions  $\Delta x_M$  has the same value as  $\phi$ , set to 3m, i.e.,  $x_M = \{0, 3, 6, 9, 12, 15\}$ m, marked with circles on the plot. The initial radio map and the measurements follow the same linear propagation as in the last simulation, for which reason once again only the dashed line is shown. Each measurement is taken repeatedly at the same position 30 times before going to the next position, which gives 180 iterations.

The final radio map is given by stair steps, with a spacing of  $\phi$  and achieved with the measurement positions going from 0 to  $L$ . This result can be explained: At the beginning the radio map is perfect and real localization delivers the exact true position. After 30 updates at this same position a step is formed, with width defined by  $\phi$ . On the next measurement position the real localization still delivers the exact position, since the radio map has not been changed at this position by the last update series. A new step is formed at this position, and one half of the previous step is entirely replaced by this new one, the other half remains.

$e_{RM,k}$  again starts from 0 and rises, although the radio map remains inside the limit area this time. This flattening effect, which forms the stair steps, is

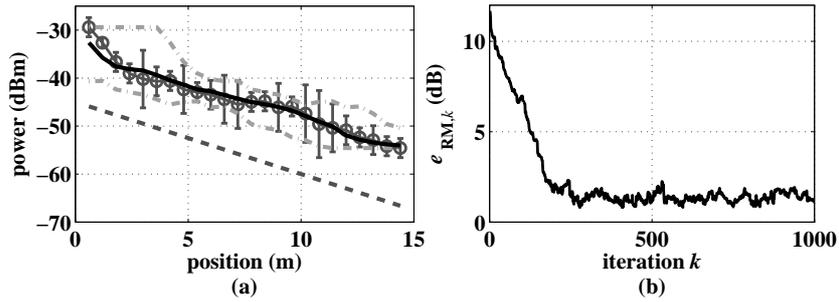


Fig. 8. Radio map learning for real world data

a direct result of (20), with the replacement of the initial model by a term dependent on the measurements  $p_{MS}$ .

In [3] another experiment shows that the uniform distribution of measurement positions is a requirement in order to SLL works well. In this example, a logarithmic propagation profile is learnt as a linear profile because the distribution was uniform in respect to the measurements rather to their positions.

Yet in [3], a discontinuity imposed by a wall is smoothly learnt, showing the versatility of the SLL.

### 1D Real World Experiment

A real world experiment with data collected at a LOS scenario has been performed. Real world experiments are different from the previously described simulations in the sense that real world signal propagation is more complex and not necessarily linear but close to logarithmic as can be seen in Fig. 8. The radio map has been initialized with a linear propagation profile, and with a large offset, being far from the true measurements such that the improvement by the SLL can be clearly seen.

Due to the findings in the previous subsection, SLL has been applied using a uniform distribution of measurement positions in combination with a random order.  $\Delta x_M$  is set to 0.6m. Each of the 24 positions was measured 30 times, so that a representative collection of noisy measurements for each position was achieved, being then randomly selected during 1000 iterations.

The final radio map (bold black line) shows that the real world propagation, which is close to logarithmic, is learned. The noise (see error bars) is significantly reduced by SLL such that the final error is within the analytical boundary, given by  $\phi = 3m$ .

Hence, the found statistical conditions, that is, the uniform distribution in space and random ordering of measurements, are verified by the real world experiment.

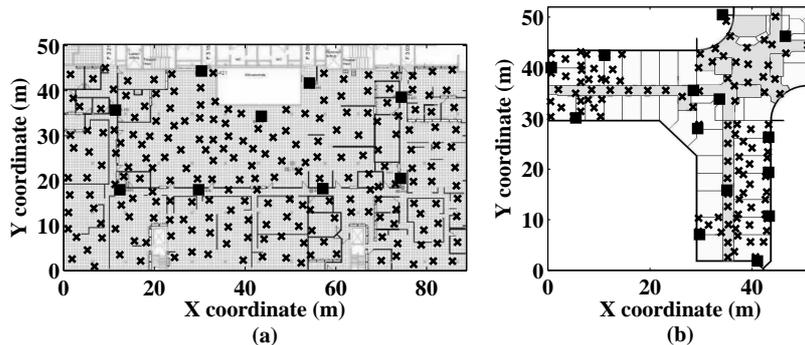


Fig. 9. Office plant with DECT (a) and WLAN (b) BSs

## 5 Results on 2D Real World Scenarios

Part of the proofs so far has only been shown for 1 dimensional examples, and that for the same reason as the counterparts proofs for SOM: they are unfeasible for more dimensions, although there always been practical examples in 2 or 3 dimensions that do converge.

The SLL was tested in real world scenarios with successful results and the results here verify the validity of this algorithm beyond theoretical set of containments.

Since SLL is independent from the radio technology used, here are shown examples with DECT and WLAN. The usual pattern matching with NN is compared with SLL using the same validation data and for different initial models.

The localization error  $e_x$  is calculated as the mean localization error among all validation positions at one given instant, i.e., the distance between the location of all validation positions and their correspondent located positions with the actual radio map  $\mathbf{p}_k(\mathbf{x})$ .

The weighting function  $f_{c,k}$  used for the 2 dimensional SLL has the form:

$$f_{c,k}(\mathbf{x}) = \begin{cases} \kappa \cdot \left(1 - \frac{d_{c,k}(\mathbf{x})}{\phi}\right), & \text{if } d_{c,k}(\mathbf{x}) \leq \phi \\ 0, & \text{if } d_{c,k}(\mathbf{x}) > \phi \end{cases} \quad (41)$$

with  $d_{c,k}(\mathbf{x}) = \sqrt{(\mathbf{x} - \mathbf{x}_{c,k})^T (\mathbf{x} - \mathbf{x}_{c,k})}$  and which correspond to Fig. 5, only that here  $f_{c,k}$  has the shape of a cone.

The first test environment is an office plant with 9 DECT BSs installed (marked as squares), as depicted in Fig. 9(a). There are 223 training positions (crosses) which are also used as validation points. The second test environment is also an office plant with 14 WLAN BSs installed (squares), as seen in Fig. 9(b). There are 114 training positions (crosses) used also as validation points.

Considering the DECT office plant, the 223 training positions were randomly selected, with repetitions allowed, during 5000 iterations and their measurements used as unlabeled input data for the SLL. The initial models considered were the LM and the DPM, as described in Sect. 2.

Figure 10 (a) shows the evolution of SLL for 5000 iterations, displaying  $e_x$  and the SLL parameters  $\kappa$  and  $\phi$ . From this plot it can be stated that the linear model is simpler than the DPM as the initial error is bigger for the first. In fact, the DPM is more complex than the LM, since environment layout information is built in the model. Nevertheless, the SLL steps improve both models, as it can be seen with the fall of the localization error. A steady state is achieved very near the theoretical lower bound for accuracy using common pattern matching, which for this scenario is calculated as  $\sqrt{A_{\text{DECT}}/223} = 4.5\text{m}$ , where  $A_{\text{DECT}}$  is the area of the considered office facility.

For the WLAN office plant, the same proceeding as for the DECT set up was used: 114 training positions were randomly selected, with repetitions allowed, and their measurements were used as unlabeled input data during 5000 iterations. The initial models were again the LM and the DPM.

Figure 10 (b) shows the results of SLL for this setup. Here, the initial error difference between the linear model and the DPM is even bigger. This can be explained by the fact that the walls at this office are covered with metallic foils that attenuates highly the signals. Accordingly the information about the walls gives already a very good initial model leading to low localization error prior to the start of learning (4.8m). Further learning by the SLL brings improvement, as it can be seen with the fall of the localization error, but not much. A steady state is again achieved near the theoretical accuracy lower bound using common pattern matching, calculated as  $\sqrt{A_{\text{WLAN}}/114} = 3.3\text{m}$ , where  $A_{\text{WLAN}}$  is the area of the WLAN office facility.

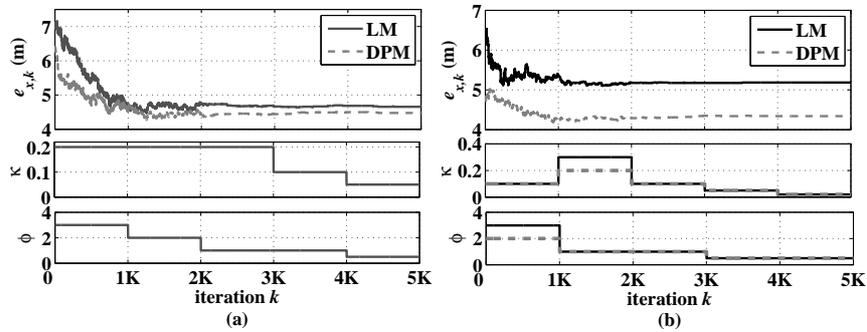


Fig. 10. Experiments at the DECT (a) and WLAN (b) office plant

## 6 Conclusions

Indoor positioning based on communication systems typically use the RSS as measurements. For proper operation such a system often requires many calibration points before its start. Applying SLL a self-calibrating RSS-based positioning system can be realized.

The algebraic and statistical conditions required to perform SLL have been explored. Important properties of SLL are the replacement of the initial radio map by the measurements, the reduction of noise by exponential filtering of different measurements, and the existence of a boundary limit defined by the adaptation width  $\phi$  and by the profile of the signal propagation.  $\phi$  and the learning rate  $\kappa$  should be kept as small as possible in order to achieve a high accuracy. A good trade-off between accuracy and speed can be achieved by starting with larger parameters and reducing them over time.

The statistical conditions impose the use of a uniform distribution of measurement positions over a limited interval and in combination with random ordering. Such implementation can be easily achieved by starting the system and performing SLL in batch, i.e., measurements are collected until the space is sufficiently covered, then the measurement collection is randomly ordered and SLL is performed to self-calibrate the RSS-based positioning system.

The initial model must be physically plausible and its complexity reflects directly the starting accuracy. Nevertheless, SLL iterations will improve the initial model and finally reach the accuracy boundary imposed by the measurement position density. The advantage of SLL is significant: in contrast to existing solutions no manual calibrations are required. The approach is self-calibrating thereby realizing a RSS based localization system with truly low costs for installation and maintenance.

## References

1. Bahl, P., Padmanabhan, V.N.: RADAR: An in-building RF-based user location and tracking system. In: IEEE InfoCom'00, pp. 775–784. Tel Aviv, Israel (2000)
2. Battiti, R., Brunato, M., Villani, A.: Statistical learning theory for location fingerprinting in wireless LANs. Tech. rep., Università di Trento (2002)
3. Betoni Parodi, B., Lenz, H., Szabo, A., Bamberger, J., Horn, J.: Algebraic and statistical conditions for the use of SLL. In: ECC'07. Kos, Greece (2007)
4. Betoni Parodi, B., Lenz, H., Szabo, A., Bamberger, J., Horn, J.: SLL: Statistical conditions and algebraic properties. In: IEEE 4<sup>textth</sup> WPNC'07, pp. 113–120. Hannover, Germany (2007)
5. Betoni Parodi, B., Lenz, H., Szabo, A., Wang, H., Horn, J., Bamberger, J., Obradovic, D.: Initialization and online-learning of RSS maps for indoor / campus localization. In: PLANS 2006, pp. 164–172. San Diego - CA, USA (2006)
6. Correia, L.M.: Wireless Flexible Personalized Communications. John Wiley & Sons, Inc., New York, NY, USA (2001)
7. Cottrell, M., Fort, J.C., Pagés, G.: Two or three things that we know about the Kohonen algorithm. In: ESANN'94, pp. 235–244. Brussels, Belgium (1994)

8. Goldsmith, A.: *Wireless communications*. Cambridge University Press (2005)
9. Günther, A., Hoene, C.: Measuring round trip times to determine the distance between WLAN nodes. In: *Networking 2005*. Waterloo, Canada (2005)
10. Harter, A., Hopper, A., Steggles, P., Ward, A., Webster, P.: The anatomy of a context-aware application. In: *MobiCom*, pp. 59–68. Seattle - WA, USA (1999)
11. Haykin, S.: *Neural Networks: A Comprehensive Foundation*. Prentice Hall, Upper Saddle River, NJ, USA (1998)
12. Horn, J., Hanebeck, U.D., Riegel, K., Heesche, K., Hauptmann, W.: Nonlinear set-theoretic position estimation of cellular phones. In: *ECC'03*. Cambridge, UK (2003)
13. Kohonen, T.: The self-organizing map. *Proc. of the IEEE* **78**, 1464–1480 (1990)
14. Liberti, J.C., Rappaport, T.S.: *Smart antennas for wireless communications: IS-95 and third generation CDMA applications*. Prentice Hall, Upper Saddle River, NJ, USA (1999)
15. Lott, M., Forkel, I.: A multi-wall-and-floor model for indoor radio propagation. In: *IEEE 53<sup>rd</sup> VTS'01 Spring*, vol. 1, pp. 464–468. Rhodes, Greece (2001)
16. Ni, L.M., Liu, Y., Lau, Y.C., Patil, A.P.: LANDMARC: Indoor location sensing using active RFID. *Wireless Networks* **10**, 701–710 (2004)
17. Obradovic, D., Lenz, H., Schupfner, M.: Sensor fusion in Siemens car navigation system. In: *14<sup>th</sup> IEEE Machine Learning for Signal Processing*, pp. 655–664. São Luiz, MA, Brazil (2004)
18. Oppermann, I., Karlsson, A., Linderbäck, H.: Novel phase based, cross-correlation position estimation technique. In: *IEEE ISSSTA'04*, pp. 340–345 (2004)
19. Pahlavan, K., Li, X., Makela, J.P.: Indoor geolocation science and technology. *IEEE Communications Magazine* **40**(2), 112–118 (2002)
20. Priyantha, N.B., Chakraborty, A., Balakrishnan, H.: The Cricket location-support system. In: *6<sup>th</sup> MobiCom*, pp. 32–43. Boston - MA, USA (2000)
21. Rappaport, T.S.: *Wireless communications principles and practice*, 2<sup>nd</sup> edn. Prentice Hall (2002)
22. Roos, T., Myllymäki, P., Tirri, H., Misikangas, P., Sievänen, J.: A probabilistic approach to WLAN user location estimation. *International Journal of Wireless Information Networks* **9**(3), 155–164 (2002)
23. Tila, F., Shepherd, P.R., Pennock, S.R.: 2GHz propagation and diversity evaluation for in-building communications up to 4MHz using high altitude platforms (HAP). In: *IEEE 54<sup>th</sup> VTS'01 Fall*, vol. 1, pp. 121–125 (2001)
24. Wang, H.: Fusion of information sources for indoor positioning with field strength measurements. Master's thesis, TU-München (2005)
25. Wang, H., Lenz, H., Szabo, A., Bamberger, J., Hanebeck, U.D.: WLAN-based pedestrian tracking using particle filters and low-cost MEMS sensors. In: *IEEE 4<sup>th</sup> WPNC'07*, pp. 1–7. Hannover, Germany (2007)
26. Wang, H., Lenz, H., Szabo, A., Hanebeck, U.D., Bamberger, J.: Fusion of barometric sensors, WLAN signals and building information for 3-D indoor/campus localization. In: *IEEE MFI'06*, pp. 426–432. Heidelberg, Germany (2006)
27. Want, R., Hopper, A., Falcão, V., Gibbons, J.: The Active Badge location system. *ACM Transactions on Informatic Systems* **10**(1), 91–102 (1992)
28. Wölfle, G., Wahl, R., Wertz, P., Wildbolz, P., Landstorfer, F.: Dominant path prediction model for indoor scenarios. In: *GeMIC'05*, pp. 176–179. Ulm, Germany (2005)

---

## Index

- DECT, 21
- DPM, *see* model, dominant path
- Kohonen, *see* SOM
- LM, *see* model, linear
- localization, 1, 7, 17
  - and learning, *see* SLL
- MAP, 7
- maximum a posteriori, *see* MAP
- maximum likelihood, *see* ML
- minimum mean square error, *see* MMSE
- ML, 7
- MMSE, 7
- model
  - dominant path, 6
  - linear, 3
  - multi wall, 5
  - non-parametric, 6
  - parametric, 3
  - piecewise linear, 4
  - propagation, 3
- MWM, *see* model, multi wall
- nearest neighbor, *see* NN
- NN, 8
- noise, 13
- PLM, *see* model, piecewise linear
- radio frequency, 3
- received signal strength, *see* RSS
- RF, *see* radio frequency
- RSS, 1, 3
- self organizing map, *see* SOM
- simulation, 17
- simultaneous localization and learning,
  - see* SLL
- SLL, 8, 10, 11, 21
- SOM, 9, 11
- WLAN, 21