On Structural Identification of 2D Regression Functions for In-door Bluetooth Localization

Rene Mayrhofer[1], Stephan Winkler[2], Helmut Hlavacs[1], Michael Affenzeller[2], and Stefan Schneider[1]

[1] University of Vienna, Faculty of Computer Science, Dr.-Karl-Renner Ring 1, A-1010 Wien, Austria
{rene.mayrhofer,helmut.hlavacs}@univie.ac.at,
[2] Heuristic and Evolutionary Algorithms Laboratory, Upper Austrian University of Applied Sciences, Campus Hagenberg, Softwarepark 11, A-4232 Hagenberg, Austria
{stephan.winkler,michael.affenzeller}@heuristiclab.com

Abstract. In-door localization of mobile devices is a common problem for many current and future applications, for example to control infrastructure services or for personalized in-building navigation systems. Sufficiently capable Bluetooth support is often available in off-the-shelf mobile devices such as mobile phones, which makes Bluetooth an attractive technology for cheap and widely available in-door localization systems. However, Bluetooth has been optimized to deal with effects of radio frequency transmission such as reflection and multi-path propagation. It therefore produces highly non-linear relationships between the distance of devices and their perceived signal strength. In this paper, we aim to identify these relationships for a specific dataset of 2D device positions using structural identification methods. Driven by an extended genetic algorithm, we aim to find optimal mappings in form of non-linear equations for x and y coordinates, thus producing formal regression functions.

1 Introduction

Bluetooth localization has many interesting applications in the area of ubiquitous and pervasive computing, both out-doors and in-doors. Its main advantage is broad availability in devices — most of today's mobile phones already include sufficiently capable Bluetooth chipsets with low power consumption that can continuously remain in visible mode without significantly impairing battery life. However, Bluetooth has not been optimized for localization, as we show based on collected real-world data. Depending on the accuracy and precision reached by a specific method, certain application areas may not be supportable (such as selecting a printer to use from an array of side-by-side ones when the localization accuracy is only in the meter range). We therefore aim for highest possible accuracy given off-the-shelf mobile hardware.

Localization methods are usually distinguished between infrastructure-based (public) and client-based (private) ones. Bluetooth localization has the potential to support both, that is, to allow the infrastructure to localize mobile clients as

well as to support self-localization of these clients based on the environment. This potentially supports a wide range of applications, for example: to use a mobile phone to interact with infrastructure services such as printers, displays, or doors based on their location; to track users (that is, the mobile phones they carry) as part of an in-building navigation system that displays personalized arrows on infrastructure displays; or as an in-building tour guide executed locally on the mobile phone to display localized content.

The major problems of Bluetooth localization are accuracy and update speed. Accuracy is impaired by radio frequency (RF) effects such as reflection, absorption, and multi-path propagation of signals between a sender and a receiver. Bluetooth has been developed to deal with – and partially make use of – such effects to provide noise-resilient communication in the free 2.4 GHz range. On the other hand, these effect along with building elements (e.g. doors, walls, windows, etc. all have different RF characteristics) lead to non-linear relationships between the measured signal strength and the real distance between two devices. Our aim is to model these relationships with highest possible accuracy.

2 Related Work

Indoor localization in general and Bluetooth-based localization in particular have recently seen increasing research interest. Apart from outdoor applications with GPS as the standard localization method, the most commonly employed indoor sensing technologies are 802.11 WLAN (e.g. [1,2]), 802.15 Bluetooth (e.g. [3]), various ultra-wide band (UWB) implementations (e.g. the commercial Ubisense system), and ultrasound (e.g. [4]). In comparison to UWB and ultrasonic localization systems, Bluetooth and WLAN typically provide significantly worse accuracy (in the area of meters compared to cm-range for UWB and ultrasound), but support off-the-shelf mobile devices such as laptops or smart phones with their built-in wireless networking hardware. The major advantage of mobile device-based methods is that self-localization does not necessarily reveal this highly sensitive information to third parties. Users can benefit from localized services while still safeguarding their own location tracks. While UWB systems are typically infrastructure-based, Bluetooth, WLAN, and ultrasound support both.

In this paper, we assume that not only the visibility of Bluetooth devices (the most commonly analyzed information for simple Bluetooth localization systems, e.g. [3,5,6]), but the relative signal strength readings between the fixed base stations and the mobile device are used as the basis for location estimation. This specific method has already been studied before, and the following three publications were especially inspiring for our work. Kotanen et al. [7] present a client-based localization system that tries to explicitly estimate the distance to each of the base stations (whose positions are assumed to be known to the mobile device) from Bluetooth RSSI readings and a consecutive Kalman filtering step. The reported average absolute error is 3,76 m, which indicates the difficulty of directly estimating the distance between two Bluetooth devices from their signal strength. In contrast, we use machine learning methods to map from

trained signal strength readings to absolute positions and therefore implicitly support more complex, location-dependent models for these relationships. By using structure identification, we lessen the disadvantage of black-box behavior that typical machine learning methods such as neural networks exhibit. Ye [8] presents an infrastructure-based Bluetooth signal strength sensing system, but omitted details on the localization heuristic due to a claimed patent application. The average absolute error is reported in the range of 7 m. Genco et al. [9] were first to apply genetic algorithms to infrastructure-based Bluetooth localization, albeit not for the actual location estimation but for minimizing the number of required base stations. The best reported accuracy is 37,5 cm.

3 Problem Specification

We assume multiple Bluetooth devices (typically USB "dongles") to be distributed over a limited environment, for example an office consisting of multiple rooms. These infrastructure devices are placed strategically and fixed in their location. A mobile Bluetooth device then roams freely within this environment and should be localized based on Bluetooth readings. Both the infrastructure devices (either connected to the same host or to multiple networked hosts) and the mobile device can perform so-called Bluetooth inquiries to query which other Bluetooth devices are in range and can create direct connections to those devices that have been found. When a connection has been opened, both communication partners can locally determine an estimate of the signal strength. These measurements are specific to the respective device and generally not comparable. We assume a calibration phase involving multiple different mobile devices but the same set of infrastructure "sensors" during which signal strengths are systematically recorded for multiple locations (of the mobile device).

The problem is thus, given a set of n signal strength estimates (between the mobile device and each of the infrastructure devices), to determine the approximate 2D position in the form of x and y values. In our current implementation, the signal strength estimate is based on RSSI readings as provided by the Bluetooth HCI link-level API. Potential positions at which the mobile device should be localized may not have been included in the calibration set, and some form of interpolation or regression is therefore required.

4 Location Estimation

This mapping problem can be addressed with different approaches. Practical experience shows that simple mappings are not sufficient (cf. Fig. 2b). We therefore focus on two machine learning methods that have already been applied successfully to related problems: standard neural networks and genetic algorithms.

4.1 Neural Network Approximation

Multi-Layer Perceptrons (MLPs) are feed-forward neural networks composed of multiple layers of neurons. The first layer (also called input layer) directly

encodes the inputs of the mapping problem, in our case the 4 signal strength estimates. The last layer (also called output layer) computes the required outputs, in this case the approximation of x and y coordinates. We use the standard approach of smoothing input values (to lessen the influence of noise) by averaging over a sliding time window and scaling to $[0.1; 0.9]$. Out of the available measurements, a random selection of 60% is used for training the MLP using backpropagation learning, 20% for validation during training, and the remaining 20% for testing and computing the accuracy. The training process is manually stopped when the error rate on the test set converges to a perceived minimum.

4.2 Evolutionary System Structure Identification

Genetic programming (GP) is based on the theory of genetic algorithms (GAs) and utilizes a population of solution candidates which evolves through many generations towards a solution using certain evolutionary operators and a selection scheme increasing better solutions' probability of passing on genetic information; the goal of a GP process is to produce a computer program solving the optimization problem at hand. In the case of *structure identification*, solution candidates represent mathematical models; these models are evaluated by applying the formulae to the given training data and comparing the generated output to the original target data. Figure 1 visualizes the GP cycle: As in every evolutionary process, new individuals (in GP's case, new programs) are created and tested, and the fitter ones in the population succeed in creating children of their own; unfit ones die and are removed from the population [10].

Within the last years we have set up a GP based structure identification framework that has been successfully used in the context of various different kinds of identification problems (e.g. mechatronics, medical data analysis, and the analysis of steel production processes [11]). One of the most important problem independent concepts used in our implementation of GP-based structure identification is offspring selection [12], an enhanced selection model that has enabled genetic algorithms and genetic programming implementations to produce superior results for various kinds of optimization problems. As in the case of conventional GAs or GP, offspring are generated by parent selection, crossover, and mutation. In a second (offspring) selection step, only those children become members of the next generation population that outperform their own parents. This process of creating new children is repeated until the number of successful offspring is sufficient to create the next generation's population.

Genetic programming can be used for data based modeling. A given system is to be analyzed and its behavior is to be modeled formally. This process is (especially in the context of modeling dynamic physical systems) called *system identification* [13]. The main goal here is to determine the relationship of a dependent (target) variable t to a set of specified independent (input) variables z. Thus, we search for a function f that uses z and a set of coefficients w such that $t = f(z, w) + \epsilon$ where e represents the error (noise) term. The structure of f is not pre-defined – it is part of the GP based identification process to identify

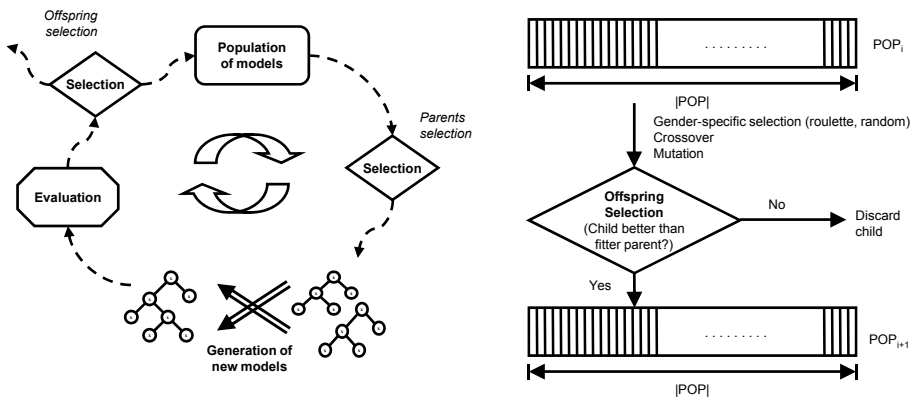


Fig. 1. Left: The extended genetic programming cycle including offspring selection. **Right:** Strict offspring selection as used here within the GP process.

the optimal structure for a formula, to find a set of relevant variables, and to optimize the terminals' parameters.

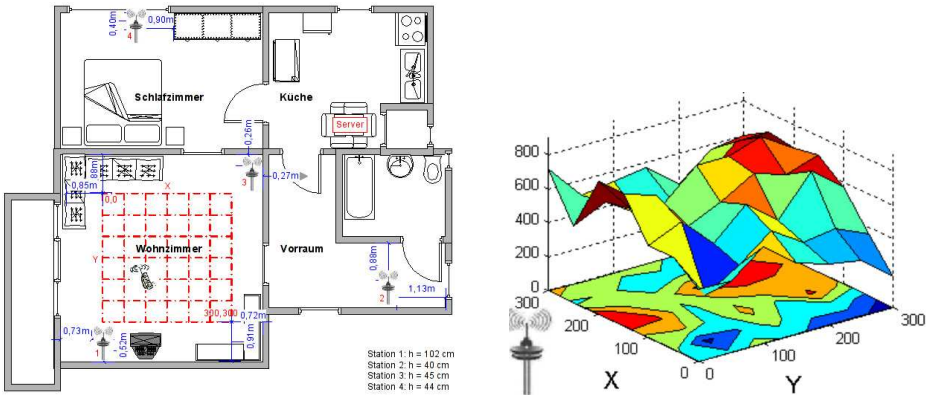
Applying this procedure we assume that a model can be created with which it will also be possible to predict correct outputs for other data examples (test sample); from the training data we want to generalize to situations not known (or allowed to be analyzed) during the training phase.

5 Experimental Evaluation

Initial data collection was carried out using four base stations distributed over different rooms of a small flat and recording RSSI estimates to two different mobile devices: a laptop (Bluetooth class 2) and a mobile phone (Bluetooth class 3). Systematic training data was then collected by placing the mobile devices on a 7x7 grid with 50 cm field width, resulting in an overall area of 3x3 m^2 (see Fig. 2a) and collecting roughly 15 minutes of measurements for each grid point at 0.5 Hz. Figure 2b gives an example of the recorded signal strengths from a single sensor (base station 4) to the mobile phone at all grid points and shows clearly that we can not assume a simple linear dependency between distance and Bluetooth signal strength.

5.1 Results using Neural Network Approximation

For MLPs, the number of so-called "hidden" layers and numbers of neurons in each of these layers have a significant influence on the overall performance. Unfortunately, these are specific to the problem and data set and therefore need to be optimized alongside the actual neural weights. For this data set, a fully connected 4-60-30-2 network, i.e., the required 4 input and 2 output neurons with 2 hidden layers with 60 and 30 neurons, respectively, performed best.



(a) 7x7 grid with 50 cm field width with base station sensors in different rooms

(b) Measurements for mobile phone signal strength estimates (sensor 4)

Fig. 2. Experimental setup

After smoothing, 67 measurements were available for each grid point. Randomly chosen sets for training and testing resulted in an averaged absolute error of 9.6 cm (median 5.4 cm) with the class 2 device (laptop) and 3.4 cm (median 1.9 cm) with the class 3 device (mobile phone).

5.2 Results using Evolutionary System Structure Identification

We have also applied enhanced genetic programming (i.e., GP with strict offspring selection) for identifying mathematical models that are able to predict the x and y coordinates of the mobile devices. Based on manual optimization, we used a population size of 500 and maximum model complexity of 10 levels (i.e., the maximum height of the evolved structure trees was set to 10) with single point crossover and single point mutation (mutation probability: 10%) as genetic operators, mean squared error (mse) as evaluation function, strict offspring selection (i.e., success ratio and comparison factor were both set to 1.0 and maximum selection pressure set to 300), and the maximum selection pressure as termination criterion.

Using these algorithmic settings we have executed 5 independent test runs each for identifying models for the x and y coordinates. The identification data available for the algorithm have been split into a set of training data (containing 80% of the identification data) and a validation set (containing the remaining 20% of the identification data); the algorithms were configured to use the set of training samples for optimizing the evolved models, and eventually those models were presented as results that performed best (with respect to mse) on the validation samples. The resulting models have been evaluated on test data not seen by the identification algorithm and not smoothed to obtain more

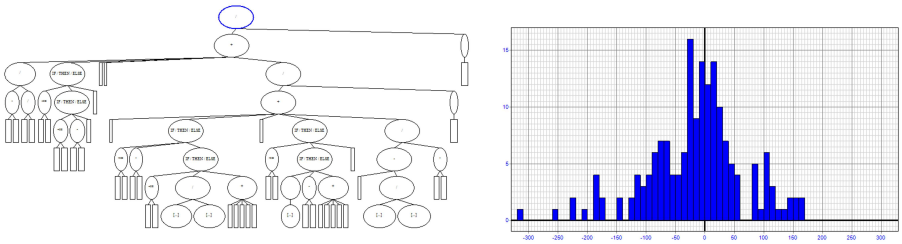


Fig. 3. Left: Structure tree representation of the best model identified by GP for x coordinates. **Right:** Resulting error distribution of this model.

realistic error estimates. Due to the independent training of models for x and y coordinates, we also analyze them separately:

- The models for x coordinates performing best on validation data show a *mse* of 41.87 cm ($\sigma = 5.92$) on training data, and 44.15 cm ($\sigma = 7.24$) on the test data set. Figure 3 shows a structure tree representation of the evaluation of the best model (with respect to training quality) and the distribution of the errors made using this model on test data.
- The models performing best on validation data show a *mse* of 55.68 cm ($\sigma = 5.43$) on training data, 55.5 cm ($\sigma = 8.16$) on test data.

These results are promising, but also show potential for further improvement of the quality of the results achievable using GP. We see that the test quality of the results is close to their training quality, i.e., overfitting does not seem to be major issue in this context. We are confident that the use of filtered data as well as more complex models should lead to significantly better results, potentially even comparable to those achieved by the MLP but with the advantage of explicit models for obtaining the coordinate estimates.

6 Conclusions and Outlook

The general approach of Bluetooth localization based on a learned model of systematic signal strength readings is usable for both infrastructure-based and client-based localization, but needs extensive training. Because this training step only needs to be performed once for each location, it can still be practical for in-door settings that require high localization accuracy and can afford certain fixed infrastructure components. The learned and potentially manually optimized model (which is small and can be used efficiently at run-time) can then be transmitted automatically to all clients (for example using Bluetooth OBEX push to any new devices entering the area).

Compared to other approaches, accuracy is potentially better, because the complex models constructed by the two machine learning methods studies in this paper can potentially accommodate difficult settings (e.g. with metal and

different surfaces in in-door scenarios) where linear or other simple relationships between distance and signal strength readings can not be assumed.

Current results achieved using a neural network are impressive, but still preliminary. Instead of training a certain percentage of the measurements from *all* grid points (i.e., all the data that is being tested has already been provided to the network during training), it would be more practical to use only a smaller number of grid points for training and then testing with points for which *no* measurements were trained. Our GP models were trained with this approach and show that, in principle, errors should not increase significantly. Work is underway to make the results of both methods more comparable.

A significant advantage of structure identification is the creation of a "white-box" model, offering introspectability and the possibility for manual tuning as well as small models (which consist of only two explicit mathematical formulae for x and y). In the future, we aim to further optimize the GP structure identification approach towards the results currently only achieved with "black-box" neural networks, as well as a live implementation.

References

1. Bahl, P., Padmanabhan, V.N.: RADAR: An in-building rf-based user location and tracking system. In: Proc. IEEE Infocom 2000, IEEE CS Press (2000) 775–784
2. LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, P., Borriello, G., Schilit, B.: Place Lab: Device positioning using radio beacons in the wild. In: Proc. Pervasive 2005, Springer-Verlag (May 2005) 116–133
3. Bruno, R., Delmastro, F.: Design and analysis of a Bluetooth-based indoor localization system. In: Personal Wireless Communications, Springer-Verlag (2003)
4. Hazas, M., Kray, C., Gellersen, H., Agbota, H., Kortuem, G., Krohn, A.: A relative positioning system for co-located mobile devices. In: Proc. MobiSys 2005, ACM Press (June 2005) 177–190
5. Bargh, M., de Groote, R.: Indoor localization based on response rate of Bluetooth inquiries. In: Proc. MELT'08, ACM Press (September 2008)
6. Jevring, M., de Groote, R., Hesselman, C.: Dynamic optimization of Bluetooth networks for indoor localization. In: Proc. AASN 2008. (October 2008)
7. Kotanen, A., Hännikäinen, M., Leppäkoski, H., Hämäläinen, T.D.: Experiments on local positioning with Bluetooth. In: Proc. ITCC'03. (April 2003) 297–303
8. Ye, J.Y.: Atlantis: Location based services with Bluetooth. Department of Computer Science, Brown University (2005)
9. Genco, A., Sorce, S., Scelfo, G.: Bluetooth base station minimal deployment for high definition positioning. Dipartimento di Ingegneria Informatica, Universitá di Palermo (2005)
10. Langdon, W.B., Poli, R.: Foundations of Genetic Programming. Springer Verlag, Berlin Heidelberg New York (2002)
11. Winkler, S.: Evolutionary System Identification - Modern Concepts and Practical Applications. PhD thesis, Institute for Formal Models and Verification, Johannes Kepler University Linz (2008)
12. Affenzeller, M., Wagner, S., Winkler, S.: Goal-oriented preservation of essential genetic information by offspring selection. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO) 2005. Volume 2., Association for Computing Machinery (ACM) (2005) 1595–1596
13. Ljung, L.: System Identification – Theory For the User, 2nd edition. PTR Prentice Hall, Upper Saddle River, N.J. (1999)