

Electronic Communications of the EASST  
Volume 56 (2013)



Proceedings of the Combined workshop on  
Self-organizing, Adaptive, and Context-Sensitive  
Distributed Systems  
and  
Self-organized Communication in Disaster Scenarios  
(SACS/SoCoDiS 2013)

Robust Exploration Strategies for a Robot exploring a Wireless Network

Christian Blum and Verena V. Hafner

12 pages

Guest Editors: Michael Zapf, Florian Evers  
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer  
ECEASST Home Page: <http://www.easst.org/eceasst/>

ISSN 1863-2122

# Robust Exploration Strategies for a Robot exploring a Wireless Network

Christian Blum<sup>1</sup> and Verena V. Hafner<sup>2</sup>

[blum@informatik.hu-berlin.de](mailto:blum@informatik.hu-berlin.de)<sup>1</sup>

[hafner@informatik.hu-berlin.de](mailto:hafner@informatik.hu-berlin.de)<sup>2</sup>

<http://koro.informatik.hu-berlin.de/>

Institut für Informatik, LS Kognitive Robotik  
Humboldt-Universität zu Berlin, Germany

**Abstract:** Integration of robots into wireless networks is important for a number of scenarios. One of the tasks is network exploration for which the most basic case is finding the physical outline of the network. We propose a robust algorithm for exploring the outline of a network with a mobile robot. For this algorithm we study robustness against noise for several sensory inputs.

**Keywords:** Wireless networks, Mobile robots, Unmanned aerial vehicles

## 1 Introduction

Wireless networks, ranging from cellular networks to ad hoc networks as used in car-to-car communication, are becoming a crucial part of our communication infrastructure. Furthermore the paradigm of the internet of things will add network capabilities to many objects, often via wireless interfaces [MF10]. These objects then will commence machine-to-machine communication.

Robots have to be able to integrate themselves in these different wireless networks for a lot of usage scenarios like reading out the data of a decentralized sensor network. This enables on the one hand machine-to-machine communication during which the robot can gather useful information. On the other hand, integration into such networks or even constructing the networks only from robots can offer third parties an ad hoc communication infrastructure as it is needed for example in disaster scenarios.

All wireless technologies share the air as a physical medium. Many of these technologies work in unlicensed Industrial, Scientific and Medical (ISM) bands which implies a lot of competition for bandwidth between network technologies and also between users. This in turn leads to very noisy characteristics of the network parameters of these networks which is on the one hand a problem and offers interesting dynamics for new algorithms on the other hand. For static nodes many algorithms and techniques exist that try to deal with these challenges. However robots, as mobile network nodes, offer new ways of simplifying these tasks by exploiting sensorimotor interaction. Most of the time this is done by letting the robot move and make use of correlations between the gathered sensory data and knowledge about its egomotion.

Integration into these networks also means physical integration of the robot. Wireless communication has more degrees of freedom than wired communication since no direct physical connection is needed but it still poses some restrictions on the positioning of the robot depend-

ing on the task. These tasks include finding a position with good reception or localizing other network nodes.

Since these tasks should be solved in an autonomous manner, no prior knowledge about the network, i.e. topology, physical location of nodes, etc., should be necessary for the robotic algorithms. Additionally all algorithms should only be based on local information in order to enable scaling-up to several robots and arbitrary network sizes.

## 1.1 Problem Statement

Assuming no prior knowledge about the network, the first step of integrating a robot into a wireless network is to gain knowledge about the physical and topological characteristics of this network. This means exploration of the network. The first tasks to be solved for this include finding the nodes and outlining the network. We assume that some very basic knowledge about the position of the network is available insofar as the robot knows in which general direction the network is located or that it is located in communication range of some network node.

We propose not to employ actual localization techniques for the network nodes because localization — in most cases trilateration — of nodes using only data from network parameters is a difficult task. These difficulties are caused by the nonlinear relationship between Received Signal Strength Indication (RSSI) and node distance which is especially pronounced in non-line-of-sight scenarios. Furthermore most of these techniques use known positions of so called anchor-nodes.

Thus, our algorithm consists of measuring only the relative direction of the nodes and to use RSSI measurements only in a very basic and thus robust way. The direction measurements can be performed in an arbitrary way, two of which are gradient estimation [HAK<sup>+</sup>09] and usage of a directional antenna. The usage of RSSI should be restricted in a way which avoids the difficulties of actual localization techniques in order to have low computational load and a robust algorithm.

We then determine the robustness of our algorithm in terms of sensory noise for both direction estimation and RSSI measurements. For this we perform a number of simulation runs and conduct a parameter study in the amount of noise for both parameters.

## 1.2 Related Work

Localization of network nodes is a topic which has been extensively studied in the past for wireless (sensor) networks [PAK<sup>+</sup>05]. Most of the related research is concerned with static networks and cooperative localization of the static nodes or with localization of one mobile node using the knowledge of the static nodes.

Robotic applications of exploration or navigation in wireless networks are often concerned with ground based robots. Rotatable directional antennas on fixed nodes [EAM07] can be seen as first very basic robotic nodes exploiting sensorimotor interaction. This approach has further been explored for the usage with mobile robots [DFK11]. Directional antennas are not necessary in order to extract directions from sensory information. Spatial gradients in RSSI measurements can be extracted using odometry in order to get directions to network nodes [DGS09]. With even more sophisticated statistical methods and a good a priori model of signal propagation, nodes can even be accurately localized using RSSI measurements, odometry and information from a

laser range finder [FK10].

For flying robots, more basic but also more robust algorithms are used because environmental influences like wind are more pronounced and odometry is less reliable. RSSI based tethering and also swarming has been explored extensively in projects like SMAVNET [HLZF10] or Air-Shield [DRGW10]. Both use RSSI in a basic way to estimate distance to neighboring nodes and thus maintain connectivity between the swarm nodes.

Our general idea of network robotics and the basic wireless network properties are described in [BH12]. Further details and signal propagation models as well as noise models for wireless communication are discussed in [Gol05].

We propose a new algorithm to explore the outlines of a wireless network. This is an important task when for example the coverage of a sensor network has to be evaluated or when a mesh network is partly destroyed after a disaster. Our algorithm is computationally simple, uses only directions to network nodes and uses RSSI measurements only in a very basic way. This algorithm is very robust against different kinds of noise for both sensory inputs. Furthermore, no highly accurate odometry information is necessary.

## 2 Problem Setting

### 2.1 Robot Model

This algorithm is designed for a flying robot in open space which does not have to be concerned with collision avoidance related to walls or other objects. Furthermore, we do not take special dynamic characteristics of some typed of flying robots like fixed-wing flying robots into account but assume a holonomic platform such as a multicopter.

No local positioning or odometry is necessary for this algorithm. Nevertheless, gradient estimation algorithms may be based on odometry. Furthermore, odometry may be necessary in order to use the information gathered by this exploration algorithm, i.e., measuring network coverage. Most methods for estimating odometry employed on flying robots are based on vision.

Directional information needed for this algorithm is not assumed to come from a specific source. The noise characteristics are assumed to be Gaussian which is true for a wide range of (virtual) sensors. For this scenario a directional antenna can easily be exploited because we assume a holonomic robot. Gradient based direction estimation needs basic odometry but has been shown to also work well [DGS09].

### 2.2 Wireless Communication

In most cases direct signal-to-noise plus interference (SNIR) measurements are not possible because commodity radios such as those based on IEEE 802.11 are used. These radios do not allow user access of actual physical parameters such as noise measurements or signal strengths. Nevertheless, they often allow the user to access the so-called RSSI value of each received packet. This value is sufficiently correlated with the real SNIR to use it as an estimator for basic tasks. Actual localization using only this value however is a rather difficult task. The advantage of using RSSI is that the measured values can uniquely be assigned to a specific network node. Information about packages that were only partly received or were not strong enough to be decoded

completely are mostly lost.

Most wireless communication technologies work in so called unlicensed ISM bands. Because they are unlicensed, there can be a lot of competition for bandwidth resulting in very space- and time-varying characteristics of network parameters. Thus, algorithms using these parameters have to be very robust.

Even in free-space, RSSI measurements cannot easily be used for distance estimation. This is due to the nonlinear relationship between RSSI and distance [Gol05]. For realistic scenarios this relationship can also become very noisy because of physical characteristics such as multipath fading or by interference from external sources. Sophisticated models such as those used in [FK10] can partly deal with these problems but they are often computationally costly and rely on accurate odometry and local self-localization.

Measuring directions to network nodes is more robust than distance measurements because it needs less information but can nevertheless be very noisy. The algorithm has to be designed to be able to deal with rather high levels of noise in the direction measurements.

### 3 Algorithm

---

**Algorithm 1** Network exploration algorithm using direction measurements. The robot is moving with a speed of  $\vec{v}$  ( here  $|\vec{v}| = 1$  for simplicity) and has some inertia abstracted by the use of  $\alpha$  ( $< 1$ ). Network nodes are denoted by  $node_i$ . Parameters of the algorithm are `lowThreshold` and `highThreshold` respectively. The user is required to have a general idea of the location of the network in order to set the robot off into its general direction as a starting point for the algorithm.

---

**Require:** move with constant speed  $\vec{v}$  in general direction of network

```

while not reached starting point do
  if some node in communication range then
    measure RSSI of visible nodes  $\{node_i\}_{vis}$ 
    choose node with highest RSSI  $node_j$ 
    estimate direction  $\vec{d}$  to  $node_j$  (with  $|\vec{d}| = 1$ )
    if RSSI of  $node_j < lowThreshold$  then
       $\vec{v} \leftarrow (1 - \alpha)\vec{v} + \alpha\vec{d}$  ▷ move in direction of  $node_j$ 
    else if RSSI of  $node_j > highThreshold$  then
       $\vec{v} \leftarrow (1 - \alpha)\vec{v} - \alpha\vec{d}$  ▷ move away from direction of  $node_j$ 
    else
       $\vec{q} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} -d_2 \\ d_1 \end{pmatrix}$ 
       $\vec{v} \leftarrow (1 - \alpha)\vec{v} + \alpha\vec{q}$  ▷ move perpendicular to direction of  $node_j$  to the left
    move one step with  $\vec{v}$ 

```

---

As discussed above, we only want to use directions to network nodes and no explicit distance estimation. The idea of this algorithm is to use RSSI measurements in order to keep some fixed distance to the closest node while circling it until the next node becomes the closest one. This fixed distance is unknown and does not actually matter for the algorithm as long as it is in some sensible range determined by the mean distance between neighboring nodes of the network and

the maximum communication distance. The algorithm is described in Algorithm 1 and a typical run is depicted in Fig. 1.

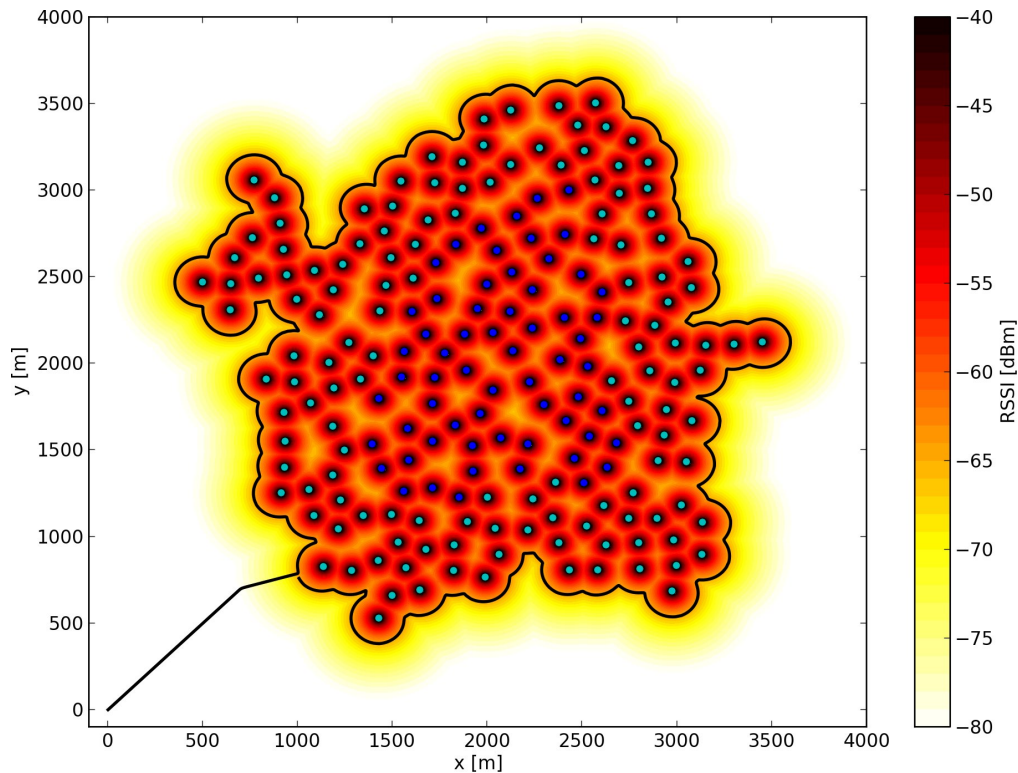


Figure 1: A typical run of the algorithm for a typical node placement is depicted. Shown are the nodes as dots where their color denotes if they have been visited or not. Cyan nodes were in communication range of the robot and blue nodes were not. Additionally, the maximum RSSI value from all nodes is shown for every point as the underlying color plot. The black line is the path the robot took in this run. In the shown simulation the measurements of the robot were noise-free.

This algorithm is designed for two dimensions but can trivially be extended to arbitrary dimensions. Typical values for `lowThreshold` and `highThreshold` are  $-65\text{dBm}$  and  $-60\text{dBm}$  respectively. We chose an abstracted robot inertia  $\alpha = 0.1$  as well as the aforementioned values for the two parameters of the algorithm for the quantitative analysis.

This algorithm exhibits several elegant characteristics. Since the movement is asymmetric, i.e., the robot always moves perpendicular to the left or on the axis of the direction of the node but never perpendicular to the right, it can never get stuck. This is the same effect as solving a maze by always keeping to the left wall [Sha51]. Because of the nondeterministic noise, the robot can get stuck in very tight situations for some time due to the noise in the RSSI measurements as shown in Fig. 2 but ultimately it will always escape due to noise again.

todo: reference to ants paper: always going to the left

The algorithm operates directly on the noisy direction measurement, which in this simulation



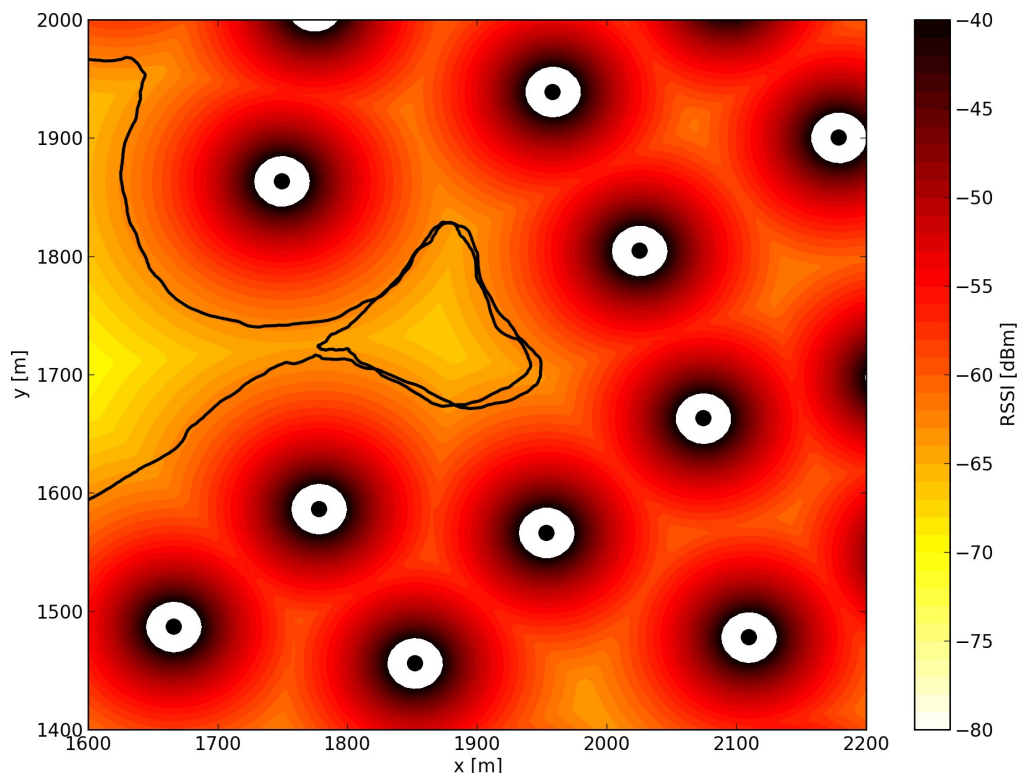


Figure 2: A situation is shown where the robot got stuck in a narrow cavity between several nodes forming some kind of cavity inside of the network. As can be seen, the robot does one additional turn before it can escape this situation. This is a fully random process since it is mainly due to the noise in RSSI measurements. The black line is the path the robot took, the nodes are shown as black dots and the maximal RSSI value from all nodes is shown for every point as the underlying color plot.

comes from some abstract sensor. In principle the movement of the robot should therefore also be very noisy. This effect is however partly compensated by the inertia of the robot which acts as a kind of mechanical low pass filter smoothing out some of the noise.

This algorithm could be trivially further improved by explicit averaging of the sensory input and thus decreasing of the effective noise but this would only result in a speed increase and not in an increased robustness. Since we were only interested in the robustness characteristics of this algorithm, we chose not to employ this method.

## 4 Simulator

The simulator used here is strictly two dimensional which is a good approximation of a flying robot at some fixed altitude. The robot is simulated as a point-like entity which is a good approximation since we assume a holonomic robot and since the length scale of the communication distance is of orders of magnitudes longer in relation to the size of the robot. The dynam-

cal characteristics of the robot were abstracted to only using inertia. We chose not to simulate environmental factors such as wind.

Our main goal was to investigate the robustness of the algorithm against noise so we chose to abstract from actual signal propagation models and to only use the so called simplified path loss model with an exponent of 3 [Gol05]. Additionally signal propagation models for a flying robot are in general more simple because they often operate under line-of-sight conditions. We also tested more complex — and thus more computationally costly — models which yielded qualitatively identical results.

We chose to work with a flying robot because we are not interested in obstacle avoidance in this scenario. Furthermore, having to rely on obstacle avoidance implies non-line-of-sight conditions which in turn would necessitate much more complex signal propagation models which don't change our results qualitatively.

Noise was simulated by simply adding a random variable with the correct distribution to the output of the simulator and feeding these values to the virtual sensors of the robot. The random number generation was carried out by Python's random package.

The nodes were placed in a random manner while guaranteeing a connected network. The algorithm was inspired by NPART [MM09]. We modified this algorithm to get a more uniform distribution of the nodes more typical of sensor or mesh networks. A typical distribution is depicted in Fig. 1.

The simulated area was  $4\text{ km}$  by  $4\text{ km}$  with 200 simulated nodes with a communication range of  $150\text{ m}$ . The simulator was implemented in Python using the standard packages NumPy/Scipy [JOP<sup>+</sup>] and multiprocessing. Visualization was done using the matplotlib [Hun07] package.

## 5 Simulation

We are interested in the robustness of the algorithm against noise. The algorithm takes two kinds of measurements, namely direction and RSSI. We model the noise on both sensory inputs as Gaussian with zero mean and some standard deviation. The noise on the direction measurement acts directly on the relative bearing angle between the robot and the network node for which the direction is measured.

We performed a parameter study in the standard deviations of the two kinds of noise. For the noise of the direction we chose an interval from approximately  $9^\circ$  to  $72^\circ$ . The RSSI noise spanned an interval from  $0.5\text{ dBm}$  to  $4\text{ dBm}$ . Both intervals were sampled at 8 positions yielding 64 different noise configurations. For both kinds of noise the two most extreme cases are depicted in Fig. 3 in order to get a graphical intuition of the noise intervals used.

### 5.1 Effectiveness

For all runs we checked if the outline was a closed loop which is a measure of the effectiveness of the algorithm. A closed loop implies that the network was outlined correctly. This is true because the network in simulation is known to be finite and by construction connected. The initial movement towards the network, which can be seen as a straight line in Fig. 1, was not taken into account in order to only assess the real outline of the exploration algorithm.



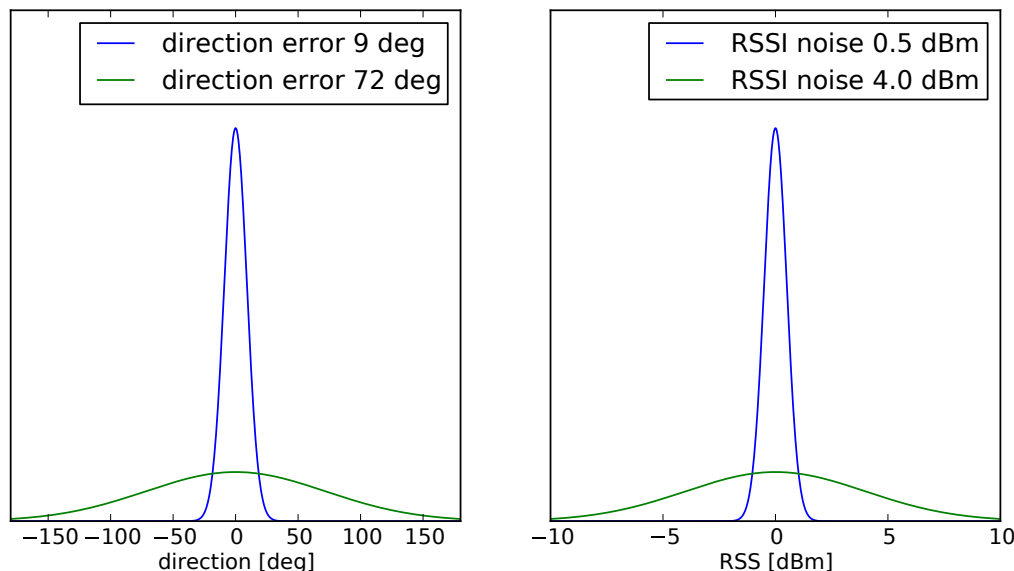


Figure 3: Shown are the two most extreme noise distributions from the parameter study for the two kinds of noise respectively. Remarkable is the most wide bearing angle distribution for the direction measurements which spreads from  $-150^\circ$  to  $150^\circ$  rendering a measured direction almost worthless.

During our parameter study we performed 1000 simulations for each noise configuration in order to be able to make statistically sound statements. Thus we performed a total of 64000 simulation runs. Every one of these simulations resulted in a closed loop independent of the amount of noise.

From examination of results of single runs we know that the robot can get stuck in narrow cavities between nodes as discussed above and depicted in Fig. 2. This figure shows a rather short episode where the robot does one additional loop before being able to escape. Depending on the actual node placement and noise realization, the robot can get stuck for an extended period of time going in circles. Nevertheless, in each and every of the simulation runs the robot could escape after some time thanks to the exploratory qualities noise brings into the system.

## 5.2 Robustness

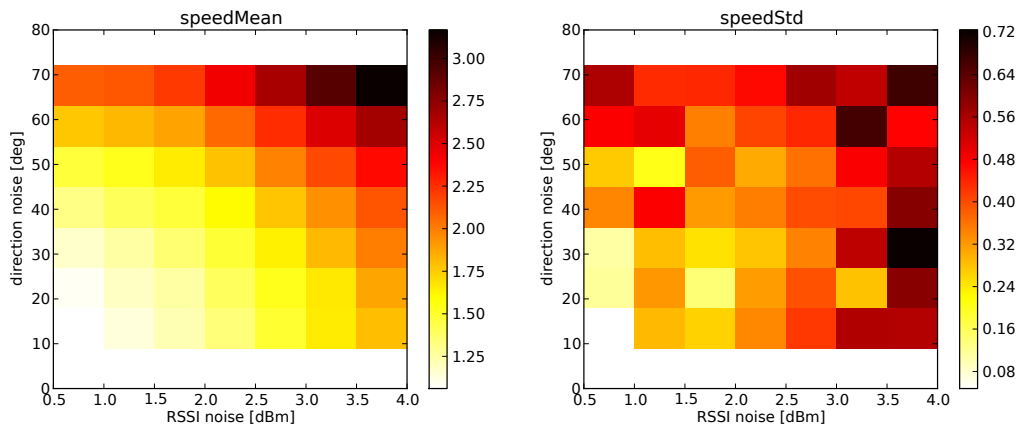
Node placement as well as both noise components are random allowing statistical statements about robustness. We ran 1000 simulation runs per noise configuration and for each run, i.e., for each node configuration, we executed the algorithm once without noise as a baseline and once with noise in order to study the effect of noise.

As metrics we choose the number of nodes which were in communication range of the robot and the time it took to complete one loop in relation to the respective numbers generated by the run without noise. Thus, the two metrics are fractions of the number of visible nodes and relative algorithm speed.

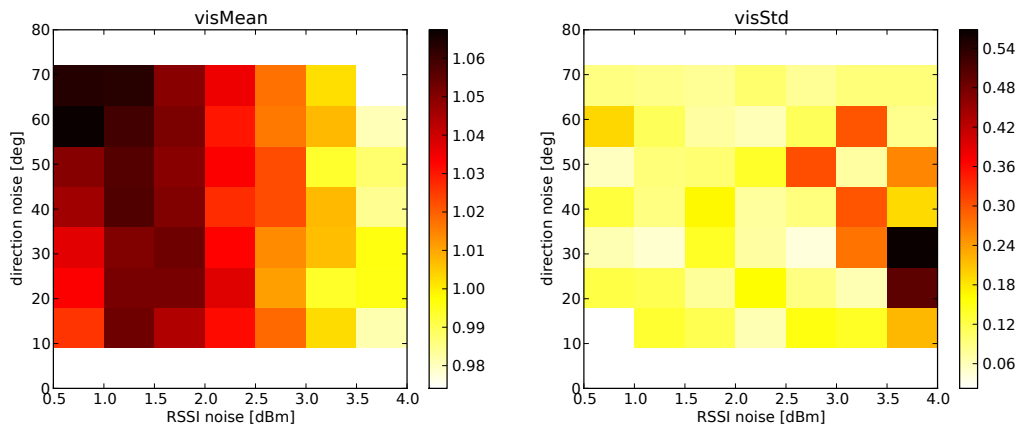
The results of these simulations are depicted in Fig. 4. The plots show mean and standard

deviation of both metrics, respectively.

The relative speed of the algorithm decreases as a function of noise where the increase is independent of the kind of noise. The standard deviation shows roughly the same behavior. This is to be expected since more noise in the sensory input also means more noise in the movement of the robot leading to more jitter and a longer effective travelled distance.



(a) Speed metric of the simulation data. Depicted are mean and standard deviation.



(b) Fraction of visited nodes metric of the simulation data. Depicted are mean and standard deviation.

Figure 4: Results of the parameter study simulations. Depicted are the two metrics speed 4(a) and fraction of visited nodes 4(b) with their mean and standard deviation respectively as a function of the two kinds of noise. For each of the noise configurations (direction bearing angle noise and RSSI noise) 1000 simulations were used as basis for each distribution.

The fraction of visited nodes shows a more interesting behavior. This metric is largely independent of the direction noise and only a function of noise in the RSSI measurements. The standard deviation seems to be rather independent of both. For low RSSI noise levels this metric is even greater than one, meaning that more nodes were discovered than the noise-free run did. For larger noise levels, this metric again drops below one. There seems to be some optimal small noise which in a way encourages exploration while degrading speed only in a minimal way.

Remarkable is how much noise the algorithm actually can tolerate. A standard deviation of  $72^\circ$  means that very little information is actually transported in the sensory data (as can be seen in Fig. 3). The results further suggest that even higher — but unrealistic — levels of RSSI noise could be tolerated.

## 6 Summary

We presented an algorithm to explore the outline of a wireless network for a mobile robot and the simulation framework in which this algorithm was evaluated. We did a parameter study in simulation and were able to confirm the expected characteristics of the algorithm.

### 6.1 Conclusion

The algorithm is very robust against noise as expected. The performance naturally degrades with increasing noise but its effectiveness is not affected. Small amounts of noise in the RSSI measurements, which control the distance to the network nodes, can even be benefiting because they increase exploratory behavior of the robot.

This very basic and computationally cheap algorithm can cope with very noisy measurements and work with minimal information. Neither complex distance estimations nor odometry or self-localization are necessary to complete this task. Algorithms such as the one presented which try to work with as little explicit information as possible are best suited for a future world with cheap autonomous robotic applications.

### 6.2 Future Work

The next step building on this algorithm will be to localize the network nodes. By exploiting the sensorimotor interaction, this may be achieved without actual complex localization via RSSI measurements but using the direction data gathered while outlining the network. Combining this information with odometry of some kind could facilitate some rough localization of network nodes which can be precise enough for a number of applications.

Enabling a robot to integrate itself into a wireless network in an autonomous way is our long-term goal of this work. Further algorithms have to be developed or existing ones extended taking more realistic physical scenarios like non-line-of-sight and obstacles into account and maybe even exploiting these effects.

**Acknowledgements:** We thank everybody who contributed to the success of this project. This includes the complete Lehrstuhl Kognitive Robotik and the DFG graduate research training group METRIK, which also funds one of the authors.

## References

- [BH12] C. Blum, V. V. Hafner. An Autonomous Flying Robot for Network Robotics. *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*, pp. 1–5, May 2012.
- [DFK11] J. Derenick, J. Fink, V. Kumar. Localization using ambiguous bearings from radio signal strength. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. Pp. 3248–3253. Sept. 2011.  
[doi:10.1109/IROS.2011.6094538](https://doi.org/10.1109/IROS.2011.6094538)
- [DGS09] K. Dantu, P. Goyal, G. Sukhatme. Relative bearing estimation from commodity radios. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*. Pp. 3871–3877. May 2009.  
[doi:10.1109/ROBOT.2009.5152689](https://doi.org/10.1109/ROBOT.2009.5152689)
- [DRGW10] K. Daniel, S. Rohde, N. Goddemeier, C. Wietfeld. A communication aware steering strategy avoiding self-separation of flying robot swarms. In *Intelligent Systems (IS), 2010 5th IEEE International Conference*. Pp. 254–259. July 2010.  
[doi:10.1109/IS.2010.5548367](https://doi.org/10.1109/IS.2010.5548367)
- [EAM07] E. Elnahrawy, J. Austen-Francisco, R. Martin. Adding Angle of Arrival Modality to Basic RSS Location Management Techniques. In *Wireless Pervasive Computing, 2007. ISWPC '07. 2nd International Symposium on*. Feb. 2007.  
[doi:10.1109/ISWPC.2007.342648](https://doi.org/10.1109/ISWPC.2007.342648)
- [FK10] J. Fink, V. Kumar. Online methods for radio signal mapping with mobile robots. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. Pp. 1940–1945. May 2010.  
[doi:10.1109/ROBOT.2010.5509574](https://doi.org/10.1109/ROBOT.2010.5509574)
- [Gol05] A. Goldsmith. *Wireless Communications*. Cambridge University Press, 2005.
- [HAK<sup>+</sup>09] D. Han, D. Andersen, M. Kaminsky, K. Papagiannaki, S. Seshan. Access Point Localization Using Local Signal Strength Gradient. In Moon et al. (eds.), *Passive and Active Network Measurement*. Lecture Notes in Computer Science 5448, pp. 99–108. Springer Berlin Heidelberg, 2009.  
[doi:10.1007/978-3-642-00975-4i\\_10](https://doi.org/10.1007/978-3-642-00975-4i_10)
- [HLZF10] S. Hauert, S. Leven, J.-C. Zufferey, D. Floreano. Communication-based Swarming for Flying Robots. In *Proceedings of the Workshop on Network Science and Systems Issues in Multi-Robot Autonomy, IEEE International Conference on Robotics and Automation*. IEEE International Conference on Robotics and Automation ICRA. Ieee Service Center, 445 Hoes Lane, Po Box 1331, Piscataway, Nj 08855-1331 USA, 2010.
- [Hun07] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering* 9(3):90–95, 2007.

- [JOP<sup>+</sup>] E. Jones, T. Oliphant, P. Peterson et al. SciPy: Open source scientific tools for Python. 2001–.  
<http://www.scipy.org/>
- [MF10] F. Mattern, C. Floerkemeier. From the Internet of Computers to the Internet of Things. In Sachs et al. (eds.), *From Active Data Management to Event-Based Systems and More*. Lecture Notes in Computer Science 6462, pp. 242–259. Springer Berlin Heidelberg, 2010.  
[doi:10.1007/978-3-642-17226-7\\_15](https://doi.org/10.1007/978-3-642-17226-7_15)
- [MM09] B. Milic, M. Malek. NPART - node placement algorithm for realistic topologies in wireless multihop network simulation. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*. Simutools '09, pp. 9:1–9:10. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, 2009.  
[doi:10.4108/ICST.SIMUTOOLS2009.5669](https://doi.org/10.4108/ICST.SIMUTOOLS2009.5669)
- [PAK<sup>+</sup>05] N. Patwari, J. Ash, S. Kyperountas, I. Hero, A.O., R. Moses, N. Correal. Locating the nodes: cooperative localization in wireless sensor networks. *Signal Processing Magazine, IEEE* 22(4):54 – 69, July 2005.  
[doi:10.1109/MSP.2005.1458287](https://doi.org/10.1109/MSP.2005.1458287)
- [Sha51] C. E. Shannon. Presentation of a maze-solving machine. In *Proceedings of the Eighth Conf. of the Josiah Macy Jr. Found. Cybernetics*, pp. 173–108. 1951.