On the Kalman Filter Approach for Localization of Mobile Robots

Conference Paper *in* Advances in Intelligent Systems and Computing · September 2018

DOI: 10.1007/978-3-319-68855-8_12

CITATIONS
5

READS
170

4 authors, including:

Miroslav Mirchev
Ss. Cyril and Methodius University in Skopje
37 PUBLICATIONS **186** CITATIONS
SEE PROFILE

Lasko Basnarkov
Ss. Cyril and Methodius University in Skopje
57 PUBLICATIONS **336** CITATIONS
SEE PROFILE

On the Kalman Filter Approach for Localization of Mobile Robots

Kristijan Petrovski, Stole Jovanovski, Miroslav Mirchev, and Lasko Basnarkov

Faculty of Computer Science and Engineering,
Rudjer Boshkovikj 16, P.O. 393, 1000 Skopje, R. Macedonia
{petrovski.kristijan.1,jovanovski.stole}@students.finki.ukim.mk
{miroslav.mirchev,lasko.basnarkov}@finki.ukim.mk
http://finki.ukim.mk/

Abstract. In this work we analyze robot motion given from the UTIAS Multi-Robot Dataset. The dataset contains recordings of robots wandering in a confined environment with randomly spaced static landmarks. After some preprocessing of the data, an algorithm based on the Extended Kalman Filter is developed to determine the positions of robots at every instant of time using the positions of the landmarks. The algorithm takes into account the asynchronous time steps and the sparse measurement data to develop its estimates. These estimates are then compared with the groundtruth data provided in the same dataset. Furthermore several methods of noise estimation are tested, which improve the error of the estimate for some robots.

Keywords: robot localization · Extended Kalman Filter · noise estimation · real-world data

1 Introduction

In many research areas there are similar types of problems requiring some kind of localization in space, such as in robotics [1, 2], wireless sensor networks [3, 4], vehicle [5] and wildlife tracking [6] etc. Plenty of different approaches have been developed for solving these problems among which the Extended Kalman Filter (EKF) described in [7] has been one of the most employed, particularly in robot localization [1]. A good introduction to the Extended Kalman Filter is given in [8]. The EKF has been obtained by extending the applicability of the classical Kalman filter [9] to problems with a nonlinear model or measurement function. All varieties of the Kalman filter belong to the group of Bayesian approaches for localization, such as particle filters and multi-hypothesis tracking, surveyed in [10]. Although localization problems have been widely addressed there are still many aspects, particularly those which are application specific, that can be further analyzed.

An implementation of EKF applied on a real dataset of several robots measuring distances to several reference points is presented in [11]. There the authors develop a general method that utilizes a decentralized robot network, where

robots communicate state estimates of objects in the environment for purposes of localization. They obtain the same results as with a centralized network with the added benefit that the robots can use the Markov property to reduce memory, without taking into account the knowledge of other robots.

In this paper we provide a case study of the problem of localization using the Extended Kalman Filter applied on the same dataset as above, focusing on single robot localization. Due to the nature of data provided, several aspects of the problem are studied such as tackling asynchronous measurements at arbitrary timestamps and running the EKF in steps of varying duration. Methods that estimate the measurement and process noise are characterized, while the inherent bias of the measurement data is examined in detail and then corrected, as in [11]. These findings could be helpful in a more appropriate application of the EKF for localization problems and its further refinements.

The organization of the paper is the following. First, in Section 2 a presentation is given of the dataset used in the experimentation. Section 3 contains a description of the application of the localization method (EKF), while Section 4 shows techniques of determining the characteristics of noise from the data in order to incorporate them in the EKF. Section 5 presents an explanation of how the data is preprocessed and offers some numerical results for localization based on several parameters. Finally, in Section 6 we give some conclusions and discuss possible directions for future work.

2 Data Description

This paper uses the UTIAS Multi-Robot Cooperative Localization and Mapping Dataset [12]. The dataset is created by 9 runs of separate experiments. In each experiment 5 robots move randomly in a confined environment for a certain amount of time. They can perceive each other and 10 stationary landmarks for the purpose of localization. While moving, each robot collects groundtruth, control and measurement data:

- the groundtruth data records the robot's position and orientation (x, y, θ), and the landmark's positions (x, y) in the laboratory reference system. This data is obtained with a 10-camera Vicon motion system that collects data every 0.01s on average with an accuracy of 10^{-3}m.
- the control data is composed of records of the robot's forward and angular speed (v, ω). For each robot they are issued roughly every 0.015s.
- the measurement data consists of estimates of distances from the landmarks and other robots and the angle at which they are perceived ($\mathbf{z} = (d, \theta)$), as seen from the reference system located at the robot where the measurements are taken. These readings are sparser occurring every 0.2s on average.

Each of the 9 experiments has a different running time and different landmark positions. In some of them a certain number of obstacles are placed in the environment.

3 EKF Algorithm

The Kalman filter is a sequential Bayesian inference method generally applied in systems evolving in time, where the state estimate at the next moment is obtained in two steps. In the first step, referred as prediction, a state estimation is based on some model of the dynamics of the system. The second step is correction, because the estimate is improved using some measurements. The literature on the Kalman filter is abundant and the authors refer the novice reader to [8].

This section simply applies the Kalman filtering procedure given in the same work, by skipping the derivation and only focusing on the points that are typical for this work. This analysis uses the Extended Discrete Kalman Filter since the measurements are non-linear functions of the state variables. To create a model that produces trajectories comparable with the groundtruth, linear approximation can be applied, and then the following model of motion can be used

$$\begin{aligned} x_{k+1}^{-} &= x_k + v_k \cos\theta_n \Delta t_k, \\ y_{k+1}^{-} &= y_k + v_k \sin\theta_n \Delta t_k, \\ \theta_{k+1}^{-} &= \theta_k + \omega_k \Delta t_k, \end{aligned} \tag{1}$$

where the minus in the superscript means that the value is predicted, subscripts denote the number of iteration, while Δt_k is the time interval between two consecutive measurements, which as was said before is not constant. To shorten the notation one can put the state variables in a column vector $\mathbf{x} = (x, y, \theta)^T$ and thus obtain a simple version of the evolution model $\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k)$. The error of the estimate is the difference between the state vector and the corresponding vector obtained from the groundtruth data $\mathbf{x}_g = (x_g, y_g, \theta_g)^T$

$$\mathbf{e} = \mathbf{x} - \mathbf{x}_g. \tag{2}$$

Then, the prediction error covariance matrix of the Kalman filter-$\mathbf{P}$, which is the expectation of the product $\mathbf{e}\mathbf{e}^T$ evolves according to

$$\mathbf{P}_{k+1}^{-} = \mathbf{A}\mathbf{P}_k\mathbf{A}^T + \mathbf{Q}_k, \tag{3}$$

where $\mathbf{Q}_k$ is the model error covariance matrix and $\mathbf{A}$ is the system dynamics matrix

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & -v_k \sin\theta_k \Delta t_k \\ 0 & 1 & v_n \cos\theta_k \Delta t_k \\ 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

When the measurements arrive, the predicted state can be improved, which according to the second step in Kalman filtering is a linear combination of the predicted state and the difference between the measurements and their estimates

$$\mathbf{x}_{k+1} = \mathbf{x}_{k+1}^{-} + \mathbf{K}_k \left(\mathbf{z}_{k+1} - \mathbf{H}_k \mathbf{x}_{k+1}^{-} \right). \tag{5}$$

In the last equation the measurements at moment $k+1$ are packed in the vector $\mathbf{z}_{k+1}$, their estimates are the product $\mathbf{H}_k\mathbf{x}^-_{k+1}$, while $\mathbf{K}_k$ is the Kalman gain matrix that optimizes the correction. The matrix elements of $\mathbf{H}_k$ relate the measurement z_i with the state variables x_j with $H_{i,j} = \frac{\partial z_i}{\partial x_j}$. As given above, the measurements for a certain robot are distances from it to some landmark or other robot and the angle at which the other robot or landmark is estimated to be seen. Focusing on a robot denoted with index r, the distance to the landmark l is

$$d_{r,l} = \sqrt{(x_r - x_l)^2 + (y_r - y_l)^2}, \tag{6}$$

while in robot's coordinate system the landmark is estimated to be seen at angle

$$\theta_{r,l} = \arctan\left(\frac{y_l - y_r}{x_l - x_r}\right) - \theta_r, \tag{7}$$

where the last expression can be obtained with simple analytic geometry. The landmarks are assumed to have known positions (x_l, y_l). Then, the matrix elements $H_{r,l}$ for all robots r and for all measurements to the landmarks l can be obtained with straightforward calculus. The only non-zero elements are the following

$$\begin{aligned}
\frac{\partial d_{r,l}}{\partial x_r} &= \frac{x_r - x_l}{d_{r,l}}, \\
\frac{\partial d_{r,l}}{\partial y_r} &= \frac{y_r - y_l}{d_{r,l}}, \\
\frac{\partial \theta_l}{\partial x_r} &= \frac{y_l - y_r}{(x_l - x_r)^2} \cdot \frac{1}{1 + \left(\frac{y_l - y_r}{x_l - x_r}\right)^2}, \\
\frac{\partial \theta_l}{\partial y_r} &= -\frac{1}{x_l - x_r} \cdot \frac{1}{1 + \left(\frac{y_l - y_r}{x_l - x_r}\right)^2}, \\
\frac{\partial \theta_l}{\partial \theta_r} &= -1.
\end{aligned} \tag{8}$$

According to the Kalman filter theory the optimal Kalman gain at iteration k is

$$\mathbf{K}_k = \mathbf{P}^-_{k+1}\mathbf{H}^T_k \left(\mathbf{H}_k\mathbf{P}^-_{k+1}\mathbf{H}^T_k + \mathbf{R}_{k+1}\right)^{-1}, \tag{9}$$

where $\mathbf{R}_{k+1}$ is the measurement error covariance matrix.

The Extended Kalman Filter expects discrete concurrent control and measurement data, where each control reading is used to obtain *a priori* state estimate (Eq. (1)) and the concurrent measurement reading is used to obtain *a posteriori* state estimate (Eq. (5)). For the purpose of this dataset a constant time step could not be used as the data readings occur asynchronously. Furthermore, as the robots do not receive a measurement simultaneously with each control datum, the robot's position can not be updated until it receives a measurement reading. This means that the robot's *a priori* state estimate is updated

asynchronously as new control data come in, and when it obtains a measurement datum, that is used that to get *a posteriori* state estimate. As the control data arrives much more frequently, one can assume that for a measurement datum that updates the *a priori* state, there was a previous control datum shortly before it i.e. the control and measurement readings occurred at roughly the same time.

4 Noise Estimation Procedures

As is given, the Kalman Filter has unspecified values for the process and measurement noise in Eq. (3) and Eq. (9), respectively. These are unique to the environment and the robot's hardware, but as they are not available for the dataset a method needs to be developed that estimates them.

4.1 Estimating the Measurement Noise

The measurement noise quantifies the uncertainty of the robot's abilities to discern targets using various sensors. For this problem it can be calculated as the covariance matrix of the difference $\mathbf{e}_z = \mathbf{z} - \hat{\mathbf{z}}$ between the measurement estimate of a relative target's state $\hat{\mathbf{z}}$ obtained from the dataset, and the actual relative target's state $\mathbf{z}$, shown in [8].

Using relative range and bearing (d, θ) from the measurement data and the robot's actual groundtruth state (x, y, θ), estimated absolute target states can be calculated. The difference between these estimates and the actual absolute target states can be used for measurement noise computation (i.e. the difference between the positions the robot perceives the targets and the positions at which they are actually located). The covariance matrix of these differences can be used in Eq. (9).

However, one can expect that the distance noise level depends on the angle at which the robot located the target (for example if a robot located the target at an obtuse angle it might cause a bigger distance error, than if the target was centred in its field of view). This hypothesis is confirmed in Fig. 1, that shows the distance measurement error depending on the measurement angle. The scattered points correspond to all measurements for all robots for the first 8 experiments[1]. As expected the results show that the distribution of the distance measurement error is correlated with the measurement angle, with smaller distance errors occurring for measurements near the centre axis of the robot, and bigger distance errors occurring at the edges of the robot's field of view, resembling a parabolic curve.

For this reason a method is proposed, which takes care to change the distance measurement noise estimate depending on the measurement angle. First the measurement data is segmented into 150 equal width bins, based on the measurement angle, then for each bin the average and variation of the errors is calculated. To further smooth the data a second degree polynomial line[2] is fitted

[1] The 9-th experiment is excluded, as it has different environment conditions

[2] We use a regression ridge model for smoothing

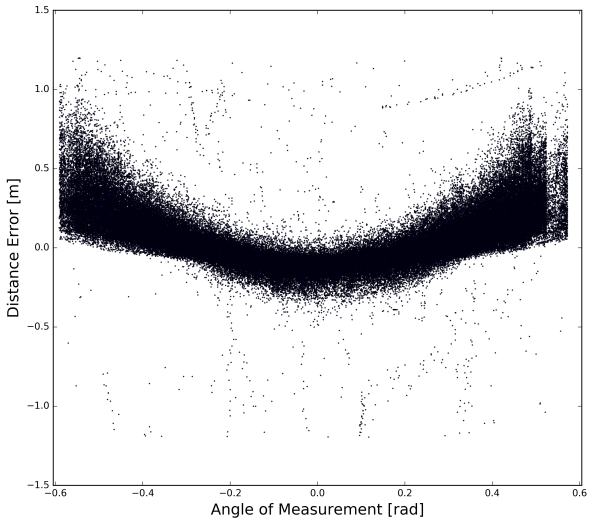

Fig. 1. Distance error versus the measurement angle. Every robot roughly has a field of view spanning in the range of -0.6 to 0.6 radians. For the purposes of this plot one percent of extreme values were removed.

over these values, and then finally these smoothed values are used for the estimation. The obtained results are shown in Fig 2. As a result, for each measurement that belongs to a particular bin the EKF can use separate values. The variation per each bin is used as the covariance noise estimate in Eq. (9). Furthermore, the Kalman filter expects the noise to have a mean of 0, while in Fig. 1 the measurement distance error is shown to be positively biased. For this reason Eq. (5) is amended, with the addition of the bin average of the measurement distance error, which causes the measurement distances to be centred around zero. The results obtained from applying this method are shown in Section 5.

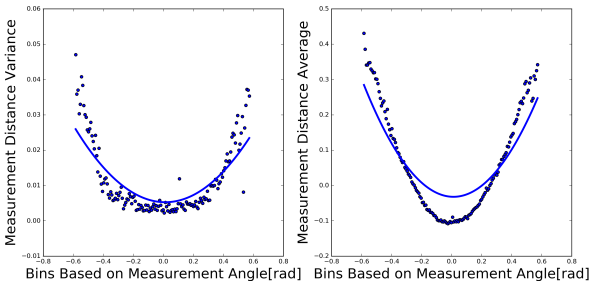

Fig. 2. This figure shows the variance and average of the measurement distance error from all the bins based on measurement angle, additionally a polynomial line of degree 2 is drawn to smooth the data.

4.2 Estimating the Process Noise

The process noise represents the difference between the system evolution according to the model and its real counterpart. In Kalman filter theory it is quantified with the covariance matrix of the error $\mathbf{e} = \mathbf{x}_k - \hat{\mathbf{x}}_k$ between the actual state of the robot $\mathbf{x}_k$ and its version obtained by evolving the previous actual state for one step by the model $\hat{\mathbf{x}}_k = \mathbf{f}(\mathbf{x}_{k-1})$.

To estimate the process noise one can run the whole experiment and calculate the average error between the robot's estimated state (1) obtained using the above model, and the true groundtruth value (x, y, θ) provided by the dataset, which should give a good estimate for the actual process noise.

This can be achieved by iterating through the control data. Every control reading is applied to the groundtruth robot position at the moment[3] of the previous control. Subsequently, the difference is calculated between that state and the groundtruth robot state at the time of the new control. These differences serve as a mean to estimate the error between the system evolution derived using control readings, and the actual robot motion calculated using groundtruth data. The process noise estimate is then computed as the covariance matrix of these differences (used in Eq. (3)). Moreover, there are several scopes of data that one can use to compute the covariance. The estimate can be obtained using differences from data readings of a single robot or more broadly data readings from the whole experiment or even differences coming from the entire dataset (this is further discussed in Section 5).

5 Results

The dataset contains many robots that are purported to move randomly during their experiment. However after some analysis one can find that some robots got 'stuck' and stopped moving for a period of time.

There are several robots with similar movement in the dataset. For these robots the state estimate error remains constant for a long period, which does not produce realistic results when evaluating error statistics.

In order to gain meaningful results, we consider only the robots that kept moving randomly for the whole duration of their experiment. In addition the 9-th experiment that contained obstacles in the environment is discarded from the analysis, as it had different environmental conditions from the other 8 experiments.

The analysis of this paper is focused on a single robot trying to localize itself inside its environment, using measurements coming only from landmarks. A general algorithm is developed, which allows experimentation with different robots from the dataset. We then study several scenarios:

- For the process noise estimate (p.n.) one can use different scopes to estimate the noise:

[3] Linear interpolation on the groundtruth data is used, so there are estimates of the robot state at any moment.

  - using data from each robot run, to estimate the process noise in its run
  - using data averaged over the whole experiment, i.e. all 5 robots that operated concurrently in the environment
  - using data averaged over the whole dataset, i.e. all 8 experiments

  The variance of error of the needed scope of data is used in Eq. (3).
- The measurement noise estimate (m.n.) can also be calculated using different scopes for estimation. The variance of the needed scope of data is used in Eq. (9). Otherwise, the algorithm can also use the equal width bin method as explained in Subsection 4.1.
- In this dataset robots perceive their environment by periodically taking pictures and then processing them to obtain relative range and bearing to recognized targets (each object in the environment has a bar-code that uniquely identifies it [12]). This means that at some distinct time a single robot potentially has measurements to several landmarks. For the case where a robot has concurrent landmarks measurements, the algorithm can choose to include all of them, or just use a single landmark measurement, in Eq. (5).

For each of these scenarios the algorithm provides an error calculated from the average difference between the estimated robot's state and the actual groundtruth robot state. Since the groundtruth and control data from the dataset are not synchronous, the state estimations occur at different moments than the groundtruth data. To obtain a valid error estimate, the algorithm uses linear interpolation to compute groundtruth data for the moments when the estimations are calculated. We then focus on the results coming from the whole dataset and results when including only the top 5 robots of the dataset (the robots which are best localized by the algorithm). For each error estimate the algorithm calculates the absolute average error, and the standard deviation of the error. The results are summarized in Tables 1 and 2.

The tables demonstrate a fairly big difference between the top 5 robots and the rest. Because of this a further direction of research might be into visualizing the robot motion and determining reasons why these robots are localized better.

Concerning the measurement noise, the equal width bin method performs good on the dataset, mainly because it takes into account both the changing variance based on the measurement angle, and the unsymmetrical measurement distance error. The method is worse when using smaller scopes of data for estimation. However it is unrealistic that one can obtain specific estimates when operating in a real environment. Moreover, the method actually performs best when focusing on the top robots of the dataset (Table 2). Further analysis could be directed into finding methods that use different segmentation techniques and better smooth the data in the different segments.

For the landmark option, as expected, the algorithm gets better results when using multiple landmarks. This is especially true when focusing on the whole dataset, however, for the top 5 robots this option does not make a big change in the error. More investigation is needed into the choice of landmark measurements used, perhaps only including the closest and most centred landmarks will give better results.

Table 1. Error statistics (absolute average and standard deviation) for different parameters, described at the beginning of this section. This data comes from the **first 8 experiments** in the dataset.

Different scenarios	**avg.; sd. of position er.[m]**	**avg.; sd. of angle er.[rad]**
p.n. from whole dataset	0.1506; 0.3798	0.1177; 0.2599
p.n. from one experiment	0.1513; 0.3805	0.1179; 0.2602
p.n. from single robot	0.1512; 0.3799	0.1180; 0.2603
m.n. with equal width bins	0.1586; 0.4384	0.1455; 0.3471
m.n. from single robot	0.1506; 0.3798	0.1177; 0.2599
m.n. from one experiment	0.1521; 0.3752	0.1180; 0.2570
m.n. from whole dataset	0.1652; 0.3971	0.1416; 0.3246
multiple landmarks	0.1506; 0.3798	0.1177; 0.2599
single landmark	0.1598; 0.3947	0.1225; 0.2696

Table 2. Error statistics (absolute average and standard deviation) for different parameters, described at the beginning of this section. This data comes from the **5 most precisely localized robots** in the dataset.

Different scenarios	**avg.; sd. of position er.[m]**	**avg.; sd. of angle er.[rad]**
p.n. from whole dataset	0.0471; 0.0632	0.0540; 0.0833
p.n. from one experiment	0.0484; 0.0646	0.0561; 0.0855
p.n. from single robot	0.0490; 0.0658	0.0572; 0.0866
m.n. with equal width bins	0.0471; 0.0632	0.0540; 0.0833
m.n. from single robot	0.0558; 0.0752	0.0417; 0.0752
m.n. from one experiment	0.0607; 0.0801	0.0442; 0.0767
m.n. from whole dataset	0.0706; 0.0881	0.0603; 0.0876
multiple landmarks	0.0471; 0.0632	0.0540; 0.0833
single landmark	0.0477; 0.0635	0.0560; 0.0846

6 Conclusions and Future Work

This paper presents a study on localization of robots based on real data. As it can be expected with such data there are certain problems pertaining to synchronizing the input data and uncertain robot's movements. However the results demonstrate several findings.

The error is very much dependent on the quality of the robot's motion through the environment with the best 5 localizations of robots achieving around 3 times better results, as compared to all the other robots in the dataset (Tables 1 and 2). When the robots view multiple targets and keep perceiving landmarks for a longer time they perform better. To achieve even better results, aspects such as the unsymmetrical measurement distance and the changing measurement variance need to be included in the computation.

This paper provides a solid foundation for further analysis in cooperative multi-robot localization, which can be performed using the same dataset as it also contains robot-to-robot measurements. Cooperative multi-robot approaches have been described in [13] and [14] using distributed EKF. In such a multi-robot scenario robot-to-robot and robot-to-landmark measurements need to be combined, which have different characteristics as the former accumulates error on its multi-hop path, while the latter is prone only to single hop measurement noise. Therefore, the localization could be performed better using a modified version of the EKF that allows weights to be given to the different types of measurements.

References

1. J.J. Leonard, and H.F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *Robotics and Automation, IEEE Transactions on*, 7(3):376–382, 1991.
2. S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 128(1–2):99–141, 2001.
3. D. Niculescu. Positioning in ad hoc sensor networks. *Network, IEEE*, 18(4):24–29, 2004.
4. G. Mao, B. Fidan, and B.D.O. Anderson. Wireless sensor network localization techniques. *Computer networks*, 51(10):2529–2553, 2007.
5. H. Li, and F. Nashashibi. Cooperative multi-vehicle localization using split covariance intersection filter. *Intelligent Transportation Systems Magazine, IEEE*, 5(2):33–44, 2013.
6. P. Juang, H. Oki, Y. Wang, M. Martonosi, L.S. Peh, and D. Rubenstein. Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet. *ACM Sigplan Notices*, 37(10):96–107, 2002.
7. G.L. Smith, S.F. Schmidt, and L.A. McGee. *Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle.* National Aeronautics and Space Administration, 1962.
8. Greg Welch and Gary Bishop. An introduction to the kalman filter. *Department of Computer Science, University of North Carolina*, 2006.
9. R.E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1):35–45, 1960.
10. D. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello. Bayesian filtering for location estimation. *IEEE pervasive computing*, (3):24–33, 2003.
11. Keith Yu Kit Leung. *Cooperative localization and mapping in sparsely-communicating robot networks.* PhD thesis, University of Toronto, 2012.
12. Keith YK Leung, Yoni Halpern, Timothy D Barfoot, and Hugh HT Liu. The utias multi-robot cooperative localization and mapping dataset. *The International Journal of Robotics Research*, 30(8):969–974, 2011.
13. S.I. Roumeliotis and G.A. Bekey. Distributed multirobot localization. *Robotics and Automation, IEEE Transactions on*, 18(5):781–795, 2002.
14. R. Madhavan, K. Fregene, and L.E. Parker. Distributed cooperative outdoor multirobot localization and mapping. *Autonomous Robots*, 17(1):23–39, 2004.