

Range-Only SLAM for Robots Operating Cooperatively with Sensor Networks

Joseph Djughash, Sanjiv Singh, George Kantor and Wei Zhang

Carnegie Mellon University

Pittsburgh, Pennsylvania 15213

Email: {josephad, ssingh, kantor, weizhang}@ri.cmu.edu

Abstract—A mobile robot we have developed is equipped with sensors to measure range to landmarks and can simultaneously localize itself as well as locate the landmarks. This modality is useful in those cases where environmental conditions preclude measurement of bearing (typically done optically) to landmarks. Here we extend the paradigm to consider the case where the landmarks (nodes of a sensor network) are able to measure range to each other. We show how the two capabilities are complimentary in being able to achieve a map of the landmarks and to provide localization for the moving robot. We present recent results with experiments on a robot operating in a randomly arranged network of nodes that can communicate via radio and range to each other using sonar. We find that incorporation of inter-node measurements helps reduce drift in positioning as well as leads to faster convergence of the map of the nodes. We find that addition of a mobile node makes the SLAM feasible in a sparsely connected network of nodes.

I. INTRODUCTION

A moving robot can be localized given a map of landmarks and an onboard sensor to sense the landmarks. Conversely, given accurate localization of the sensor, it is possible to build a map of the landmarks. A question that has recently intrigued our community is how well it is possible to do both (localize and map) if neither exist a priori. Research in Simultaneous Localization and Mapping (SLAM) makes one of two assumptions. First, and commonly, that the moving sensor can measure both range and bearing to landmarks in its vicinity [1] or, that the moving sensor can detect only bearing to the landmarks [2]. Recently, there has been some attention to the complementary case, in which the moving sensor can only detect range to landmarks. In this case the “map” is the simply the location of the landmarks. Here we consider an extension of range-only SLAM to the case where the landmarks themselves are able to measure range to each other in the hope of improving both localization of the moving robot as well as the map of the landmarks.

Such a sensing modality could be very useful in the those cases in which the environment precludes measurement of bearing to landmarks such as underwater or inside buildings where line of sight does not go very far. This is especially the case if range can be measured through walls and common objects as is possible via range from timing radio signals. Emergency response, as with rescue operations in a burning building is a particularly compelling application. In this case, it is very important to be able to track firefighters in a potentially dark and smoky environment. We envision a future

scenario in which firefighters sporadically deploy small and simple nodes as they move around. An ad hoc network formed between the nodes that localize themselves and the moving firefighter. This function could be augmented by introducing robots that could intelligently position themselves to add links for localization and communication. In our system the nodes act as ranging beacons to provide beacon-to-beacon range within the network and therefore, in the context of this paper the terms beacon and node will be used interchangeably.

Here we present early results towards accomplishing such a scenario. We have developed algorithms specifically targeted towards the case where the network of landmarks (starting in an unknown configuration) is not completely connected. That is, each beacon can range only to a few other beacons. This is important because if the network is fully connected it is possible to use well known methods such as Multi-dimensional Scaling (MDS) [3] to determine network topology. For a less connected network but with enough connections to provide “rigidity” to the network, it is still possible to determine the map of the network [4]. However, in practice rigidity is not easy to achieve and measurement of ranges in between beacons of the network is not sufficient to localize beacons unless there is a high degree of connectivity in between beacons. We propose to introduce a moving beacon (carried by human or robot) that adds edges into the network and helps make the problem of determining network topology, better conditioned. No external positioning for the moving beacon is assumed although if a robot is used, it is possible to use onboard sensors to measure heading and distance travelled. As a complement, the network of landmarks can help localize the moving beacon without drift.

In this paper we discuss results with a robot operating among beacons that communicate with each other using radio and measure range to other beacons using sonar. While ranging with radio remains the eventual goal, practical implementations are not available yet. In the meanwhile, we use range from sonar instead of radio, the difference between the two being that radio penetrates walls while sonar does not. In this paper we compare five algorithms for localization and mapping. The first method extends the Kalman filter localization to estimate the location of the beacons simultaneously, without any prior knowledge of the beacon locations. In this case only the measurements in between the moving beacons and the stationary beacons are used. The second method incorporates beacon-

to-beacon range measurements into the online Kalman filter SLAM. The third method is an off-line batch method that takes all the data from a run and produces the best estimate of the robot path as well as the beacons. The fourth method is an online version that incrementally uses the robot as a virtual node to create rigid submaps that can be solved with MDS and patched together. We compare the five cases with data taken from a roving robot operating in a partially cluttered environment.

II. RELATED WORK

Localization with landmarks known a priori has been well known for centuries and has been used in robotics for a while [5]. The extension to SLAM has been of much interest in robotics because it offers the possibility of a robot operating in an area that it has not encountered before. As mentioned earlier, most work in SLAM assumes measurements that provide both range and bearing to landmarks in the scene. The advantage of such sensors is that single observations can be considered as having a Gaussian distribution and can be readily incorporated into a methods like Kalman filtering [1], [6]. An extension of these methods is use a particle filter to deal with complexity and sensitivity to failure in data association [7]. A different set of methods is used to deal with the case where only bearings to landmarks are measured. This form of SLAM, is often referred to as “Online Structure from Motion” or “Visual Odometry”. Instead of a scanning laser range finder, measurements are made using a moving camera where salient features are tracked a cross an image sequence [8]. Principally, the difference between the two forms is that the first deals with distributions (of the sensor measurements and landmark locations) in Euclidean space while Visual Odometry deals with the projections of the landmarks on the image plane.

A. Range Only SLAM

Recent work has extended SLAM algorithms to deal with the case that only range can be measured to landmarks. One great advantage with using the sensors such as active radio or sonar beacons is that the data-association problem that plagues SLAM in general is solved trivially because measurements are often tagged with the identity of the source of the signal. Several research efforts have shown how radio beacons [9] and sonar beacons [10] can be used to perform SLAM in outdoor environments. In these cases, the linearization of a single measurement as a Gaussian distribution is difficult because on its own, a noisy range measurement is best thought of an annular distribution. These methods go through an initial step to approximately locate the landmarks. The basic idea is to store the robot locations and measured ranges the first few times the landmark is encountered and then obtain an estimate of landmark position by intersecting circles on the plane. Once an estimate of a new landmark is produced, the landmark is added to the Kalman filter where its estimate is then improved along with the estimates of the other (previously seen) landmarks.

A recent effort has shown the use of low cost RFID sensors that provides very approximate information. RFID “tags” are located in the environment within a 90 degree cone [11].

Since the tags have limited range, each measurement only provides an indication (with varying probability) of where the tag might be in a large area. An evidence grid is used to accumulate the likelihood of tags existing in discrete cells and as well to locate the robot. Another effort has sought to localize a robot from existing wireless networks using signal strength as an indication of range [12].

B. Incorporation of node-to-node range measurements

A natural extension to range-only SLAM work consists of using a collection of beacons that can measure distances between one another in addition to measuring robot-to-beacon distances. In the case where every beacon can measure the range to every other beacon, classical multidimensional scaling algorithm (MDS) can solve the problem by constructing a matrix that contains the point-to-point distances between every pair of nodes (Euclidean distance matrix, EDM) and then finding the relative locations algebraically [3]. MDS provides a list of beacon coordinates that is unique to within an affine transformation and a reflection. In the case where some of the beacon-to-beacon distance measurements are unknown, it is still possible to find a unique solution provided that the graph with beacons as nodes and distance measurements as edges is rigid as described in [13]. In one approach, missing elements of the EDM have been filled in using nonlinear optimization [14]. Alternatively, it is sometimes possible to solve neighborhoods that have a complete or nearly complete EDM and then patch the neighborhoods together [15]. Robust quadrilaterals proposed in [4] make particularly good neighborhoods because they are robust to flip ambiguities that can be caused by measurement noise. The neighborhood patching approach has been combined with the use of GPS enabled anchor nodes to limit the propagation of neighborhood-matching error [16]. More recently, approximating the distance between nodes by the number of hops between them has been studied as a means of eliminating the need for anchors [17].

One important distinction of our work is that we incorporate moving beacons into the network localization formulation. By including carefully chosen range and odometry measurements from a beacon mounted on a mobile robot, we are able to uniquely determine beacon locations in cases where the graph composed of only fixed beacons is not rigid.

C. Hardware

Since hardware for ranging with radio through the walls remains an elusive goal for which commercially available equipment does not exist today, we have developed a sensor network that uses radio to communicate data and ultrasonic measurements to range in between beacons. These beacons enable a low-cost means of omni-directional ranging, suitable for localization in environments where GPS is not available. Compared with similar ultrasonic based localization systems such as the Cricket system [17] these beacons achieve a longer ranging distance of up to 15 m with an accuracy of 2cm. They are also more effective in detecting ultrasonic signals by using four pairs of ultrasonic sensors, making it able to detect

ultrasonic signal from any direction and reducing the number of beacons required by similar systems, which in turn leads to an improved update rate of the distance measurements.

III. APPROACH

In this paper, we present a number of different range-only SLAM methods that are intended to be used with an ad hoc sensor network composed of a collection of range-finding beacons. We consider two distinct cases. In the first, the robot uses the network to measure the distances between itself and the beacons that are in fixed but unknown locations. In the second case, the robot also has access to beacon-to-beacon distance measurements taken by the network. In both cases, the objective is to maintain an estimate of robot position while simultaneously estimating the positions of all of the beacons.

A. Robot-to-Beacon Measurements

Here, we present the basic theory of Kalman filters as applied to the problem of localization and SLAM for the case where only robot to beacon measurements are available.

1) *Localization*: Let the robot state (position and orientation) at time k be represented by the state vector $q_k = [x_k, y_k, \theta_k]^T$ (see Fig. 1). The dynamics of the wheeled robot used in the experiments are best described by the following set of nonlinear equations:

$$q_{k+1} = \begin{bmatrix} x_k + \Delta D_k \cos(\theta_k) \\ y_k + \Delta D_k \sin(\theta_k) \\ \theta_k + \Delta \theta_k \end{bmatrix} + \nu_k = f(\hat{q}_k, u_k) + \nu_k, \quad (1)$$

where ν_k is a noise vector, ΔD_k is the odometric distance traveled, and $\Delta \theta_k$ is the orientation change. For every new control input vector, $u(k) = [\Delta D_k, \Delta \theta_k]^T$, that is received, the estimates of robot state and error covariance are propagated using the extended Kalman filter (EKF) prediction equations.

The range measurement received at time k is modeled by:

$$\hat{r}_k = \sqrt{(x_k - x_b)^2 + (y_k - y_b)^2} + \eta_k \quad (2)$$

where, \hat{r}_k is the estimate of the range from the beacon to the current state, (x_b, y_b) is the location of the beacon from which the measurement was received and η_k is zero mean Gaussian noise. The measurement is linearized and incorporated into the state and covariance estimates using the EKF update equations.

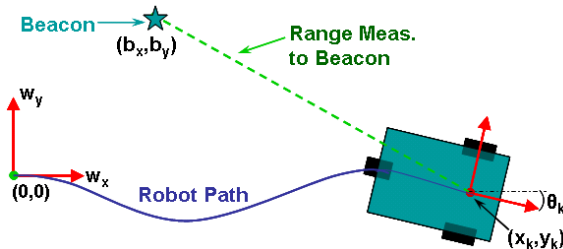


Fig. 1. Robot state and system setup.

2) *SLAM*: The Kalman filter localization formalism can readily be extended to the problem of SLAM, provided that initial estimates of the beacon locations are known. When no prior knowledge exists, it is necessary to perform some type of batch process to initialize beacon locations, which are then refined within the filter.

To extend the formalism to perform SLAM, we need only to include position estimates of each beacon in the state vector. Thus we get,

$$q_k = [x_k, y_k, \theta_k, x_{b1}, y_{b1}, \dots, x_{bn}, y_{bn}]^T \quad (3)$$

where n is the number of initialized beacons at time k . The motion model for the first three states is given by (1). The beacons do not move, so the motion model for the remaining states is trivial. The measurement model is the same as that stated in (2) with x_b and y_b replaced by x_{bi} and y_{bi} when the measurement is received from the i th beacon.

Beacon locations are initialized in an online method, such that initially there are no known beacons. We employ an approach similar to [18] that utilizes a two-dimensional probability grid to provide initial estimates of the beacon locations. The probability grid for each beacon is created using a “voting” scheme where pairs of prior range measurements from the given beacon “vote” for solutions that fit the pair. After generating this probability grid, the cell with the greatest number of votes contains (with high probability) the true beacon location. Once a subset of beacons is initialized, robot odometry measurements and range measurements from all initialized beacons are used to maintain a full state estimate (both robot and beacon locations) via EKF prediction and update steps.

B. Adding Beacon-to-Beacon Measurements

We now turn to the case where beacon-to-beacon range measurements are available in addition to the robot-to-beacon measurements used in the previous section. If each beacon can obtain measurements to enough of its neighbors, then it is theoretically possible to build a map using only beacon-to-beacon measurements. This notion of a *self localizing* sensor network is appealing and has received some attention in the recent literature. However, we have discovered that practical considerations such as limited maximum beacon range, obstacles that block line-of-sight, and sensor blind spots make it extremely difficult to achieve the level of interconnectivity necessary to perform self localization. To see why, consider Fig. 2a, which shows the range measurements available at the beginning of an experiment in a seven-node sensor network in a moderately cluttered environment. Note that nodes 24 and 29 can each get range measurements to only one neighbor, so clearly the available data is not sufficient to uniquely determine the positions of those nodes. In fact, most of the self-localization techniques described in the literature require cliques (fully interconnected subgraphs) of degree four or higher. As Fig. 2a shows, there is not a single clique of degree four before the robots begins its motion. For this reason, we have chosen to investigate means to incorporate beacon-to-beacon measurements into the SLAM problem rather than striving for self-localization. We present

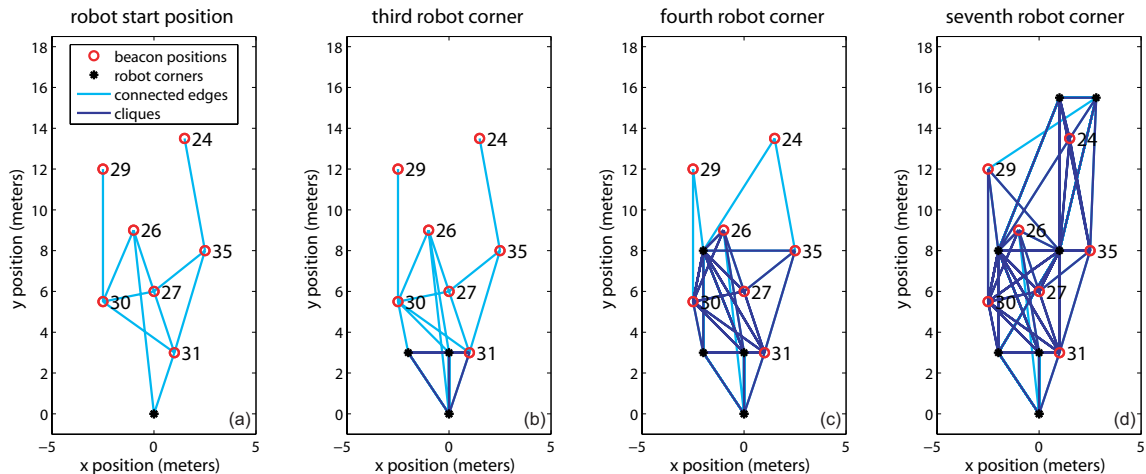


Fig. 2. This figure shows how adding virtual nodes can make the self-localization problem solvable. (a) depicts the graph at the beginning of the experiment. No cliques exist at this point, so it is impossible to localize any of the beacons. (b) the first clique appears at the third robot corner, nine more cliques are created. (d) at the seventh corner, there is enough information in the graph to uniquely localize every beacon.

three methods: two that extend the Kalman-filter-based SLAM algorithm presented in the previous section and one that uses ideas from the area of self-localizing networks to incrementally build a map by treating locations where the robot receives robot-to-beacon measurements as a virtual beacon locations.

1) *Extending SLAM*: One approach to incorporating beacon-to-beacon measurements into the localization and mapping algorithm is to simply use those measurements to update the states of the beacons that create the measurement. This can be done with a simple extension to the EKF SLAM method that uses the robot-to-beacon measurements as discussed in the previous section. All that is required is a modified measurement model that includes beacon-to-beacon as well as robot-to-beacon measurements. The estimation process then proceeds via a standard application of the EKF. Additionally, beacon-to-beacon measurements can be incorporated into the initialization step in a straightforward manner.

2) *Improving the SLAM Map*: A second method of incorporating the beacon-to-beacon measurements into the SLAM solution is to use post processing step to improve the map that results from a SLAM run. This can be posed as an optimization problem where the cost function is the Mahalanobis distance from the SLAM map added to the sum of the squares of beacon-to-beacon range measurement innovations. Let (\hat{q}, P) denote the Kalman filter SLAM estimate at the end of a run, where P is the Kalman covariance matrix for the state \hat{q} . Let (\hat{q}^M, P_M) denote the portions of the \hat{q} and P that correspond to the beacon locations, and let q_i^M be the part of the state that corresponds to the (x, y) location of the i th beacon. Define the set of available beacon-to-beacon distance measurements to be \mathcal{R} where $r_{ij} \in \mathcal{R}$ denotes the range measurement between beacons i and j . Finally, let σ_b^2 be the covariance of the error in the range measurements. Then we can improve the SLAM map by finding the q^M that minimizes the cost function

$$J(q^M) = (q^M - \hat{q}^M)^T P_M^{-1} (q^M - \hat{q}^M) + \sum_{r_{ij} \in \mathcal{R}} \frac{(r_{ij} - \|q_i^M - q_j^M\|)^2}{\sigma_b^2}.$$

3) *Self-Localization with Virtual Nodes*: Consider a robot moving through the environment that receives a measurement at a particular location. We can imagine placing an immobile virtual node at that location and including that virtual node together with the corresponding measurement as part of the beacon network. The distances between virtual nodes placed in sequence can be estimated using robot odometry, and these distances can be included as edges in the network as well. Virtual nodes placed at robot locations where two or more range measurements are likely to “rigidify” the network by increasing the likelihood of the existence of the fully-connected cliques that are necessary for self-localization. Specifically, in our experiments, the robot was able to receive multiple range measurements at each corner due to the fact that it executed the corners by turning in place. As a result, we propose a method of mapping the beacon network that adds the robot corners as virtual nodes then uses self-localization techniques to solve the map.

For example, at the beginning of the experiment (Fig. 2a) the graph that results from the available data has no cliques of degree four or greater, making it impossible to uniquely determine the positions of any of the nodes. When the robot turns the third corner, the first fully connected clique appears (Fig. 2b). Multidimensional scaling (MDS) is used to determine the relative locations of the clique nodes. Note that this clique contains three virtual nodes, including the robot starting location. Two of the nodes together with robot odometry data are sufficient to uniquely determine the global position and orientation of the clique. Additionally, the three virtual nodes can be used together to determine the correct handedness of the clique, allowing us to resolve the flip ambiguity. As a result, the location of beacon 31 is uniquely determined and added to the map.

As the robot proceeds through subsequent corners, new virtual nodes and ranges are added to the graph. At each corner, the graph is searched for new cliques. When it is possible to resolve the ambiguities in orientation and handedness, new cliques are patched into the existing map by minimizing the sum-of-squares distances between shared nodes. Fig. 2c shows that nine more cliques of degree four

are created and merged into the map at the fourth corner. As the robot proceeds through the map, more and more cliques are generated and patched in until eventually all of the beacons are mapped. Fig. 2d shows the state of the graph when the robot reaches the seventh corner in our example, at which point enough cliques have been generated to provide a unique solution for all of the beacon locations.

IV. EXPERIMENT

We have developed sensor network that uses radio to communicate data and ultrasonic measurements to range in between beacons. Each beacon consists of a microcontroller, a radio transceiver and four pairs of ultrasonic sensors (See Fig. 4) [19]. Using ultrasound pulses to measure the distances between each other, each beacon in turn sends out a radio signal followed by an ultrasonic pulse train. Other beacons that receive both signals convert the time difference between receiving the radio signal and the ultrasonic signal to a distance measurement by multiplying with the speed of sound. The beacons automatically formulate an ad hoc multi-hop wireless network to share the range information. They employ a distributed architecture in which every beacon will have all the distance measurements for all the beacons in the network, eliminating the requirement of a central node for the coordination. Every beacon can be both sender and receiver. The beacons coordinate the pinging sequence such that ultrasonic pulses from multiple beacons do not interfere with one another.

A. Noise Characteristics

In our ultrasonic-based ranging system, there are mainly two kinds of noises that affect the performance of the range estimation. One is the multipath effects caused by the reflections of the ultrasonic pulses, yielding false results when they are captured by an ultrasonic transducer. To reduce the multipath effects and also achieve 360 degree ranging capability, our ranging sensor uses four pairs of ultrasonic transducers facing four directions. The first ultrasonic transducer that captures a valid ultrasonic signal will inform the sensor to ignore the ultrasonic signals captured by any of the other three ultrasonic transducers, which are probably caused by the multipath effects. Another kind of noise comes from the beacon wrongly correlating the RF data from one sender with the ultrasonic pulses from other sources, ultrasonic pulses sent out by another sensor, environmental noise, etc. In our approach, we have implemented a sentry-based scheduling scheme such that in every time slot, only one beacon in the system can send out ultrasonic pulses so that the ultrasonic pulses sent out by the beacons will not interfere with each other.

Our experiments have shown that our beacons significantly reduce these noises, and accurately and reliably estimate range between each other. From the four experiments, we collected 20,014 inter-beacon range estimations by the beacons. Out of these estimations, we observed only 6 error estimations.

To evaluate the precision of the system, we analyze the ranging performance between two beacons, and compare it with the range measurement using a tape measure. The

result is illustrated in Fig. 3. This shows that the sensors accurately estimate the distance between the two beacons, and the difference between the average range estimation and the manually surveyed range measurement is 0.04 m (under ideal circumstances, where there are no multi-path effects on the sensors, the range error is approximately 0.02 m).

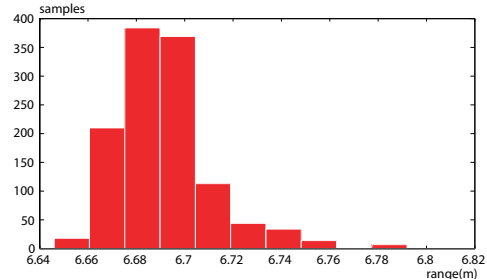


Fig. 3. Histograms of a typical set of range measurements between two sensor beacons. The surveyed range measurement between the two beacons, measured using tape measure, is 6.65 m. The observed shift in the histogram is due to multi-path within the environment and errors in manual surveying. This level of accuracy is constant across ranges from 2 m to 15 m.

We then examine the performance of the range estimations by the beacons as a function of displacement by comparing the range estimations between pairs of beacons to their true ranges. We see that most of the beacons can accurately and reliably estimate ranges between each other with a standard deviation of less than 0.04 m. Range measurements between 21 pairs of beacons, with measurements ranging from 2 m to 10 m, mostly had errors less than 0.05 m. In the case of one of the beacon pairs we observed a standard deviation of 0.4 m. Closer experimentation shows that out of 341 range measurements between these two beacons, 335 of them are close to the true range of 7.75 m, and 6 of them are close to 5.15 m. In our application, such gross errors can be easily filtered out.

B. Experiments Conducted

We carried out our experiments on a Pioneer 1 robot from ActivMedia. It is equipped with a beacon placed on top of the robot at about 1 ft above the floor (See Fig. 4). The robot was controlled by an operator who drove the robot using a wireless link. The robot was driven around within a large indoor area with partial clutter. There was no ground truth available for the exact path taken by the robot, however, the robot's intended path was surveyed. The true robot path differs somewhat from the intended path, however, it is close enough for the purposes of judging the validity of the localization and SLAM algorithm. In addition to the beacon that was placed on the robot, several other beacons were placed around the environment on top of boxes, 1 ft above the floor. The locations of these beacons were accurately surveyed to allow proper evaluation of the accuracy of our SLAM mapping results.

V. RESULTS

For each of the five methods described above, the mean absolute cross track error (XTE), standard deviation and the RMS error in the resulting maps (beacon locations) are presented in Table I.



Fig. 4. Robot used for our experiments shown along with a range sensing beacon mounted on top. The robot has wheel encoders that measure distance traveled and estimate heading. Neither estimate is good but the heading estimate is particularly poor.

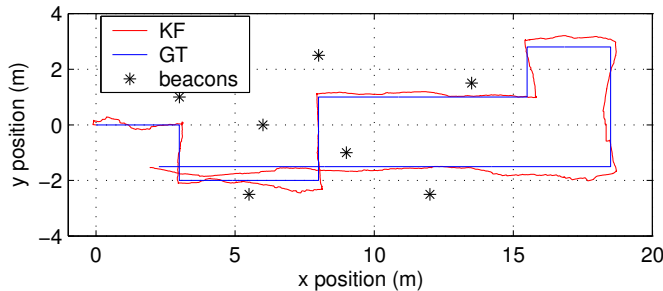


Fig. 5. Localization using Kalman filter (KF) and the reference ground truth (GT). In this case the location of the beacons is known a priori. The robot starts at (0,0) and ends at (2.5, -1.75), travelling 45.85 m

A. Localization with Robot-to-Beacon Measurements

Here we present the results of performing localization on a dataset acquired by driving a robot through a cluttered environment (See Fig. 5). The numerical errors shown in Table I (Column 1) represent the error of the estimated path when compared with the “intended” travel path. This therefore implies that the magnitude of the error terms represent only an exaggerated approximation of the real errors. However, even without an accurate ground truthing system to help calculate the path errors, the localization results that we present here can be used to judge the improvements and accuracy of the SLAM results we will present next.

B. SLAM with Robot-to-Beacon Measurements

The results that were acquired from performing SLAM on the dataset are presented in Fig. 6. A simple affine transform (AT) is performed on the final beacon locations estimates in order to re-align the locally accurate solution into a global coordinate frame. From the SLAM output, we are able to extract an estimate of the network topology. The map that was acquired from SLAM can then be fed back into the localization algorithm to generate numerical errors that could reasonably be compared to the results from performing localization on the “true”/surveyed beacon locations. As can be observed from the numerical results, Table I (Column 2), the SLAM algorithm provides an accurate network topology

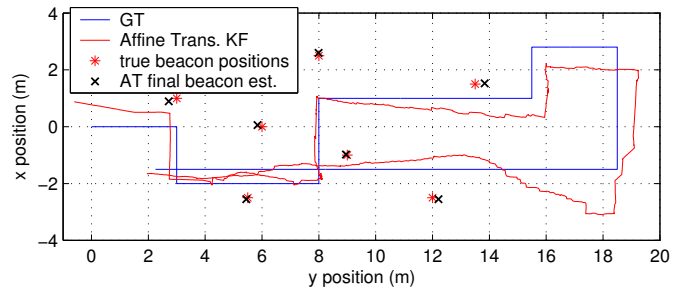


Fig. 6. Kalman filter SLAM results used to perform localization (on the same dataset). The location of the beacons is completely unknown at the start of SLAM. The path and beacon estimates shown include an affine transform that re-aligned the local solution into a global coordinate frame.

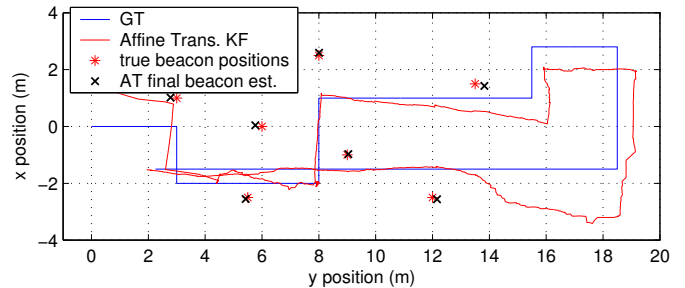


Fig. 7. Kalman filter SLAM (with inter-beacon measurements) results used to perform localization (on the same dataset). The location of the beacons is completely unknown at the start of SLAM. The path and beacon estimates shown are after an affine transform that re-aligns the local solution into a global coordinate frame.

that produce comparable results in localization.

C. SLAM with Beacon-to-Beacon Measurements

The results for the case where we modified the simpler Kalman filter SLAM algorithm to incorporate the inter-beacon measurements within its online estimation process is shown in Fig. 7. The estimated path and beacon locations have been affine transformed to align the locally accurate solution into a global map for proper evaluation. Upon acquiring the beacon locations from the SLAM algorithm, we then feed the resultant map into the localization algorithm to acquire numerical results that can be compared with the results from the other methods presented in this paper (See Table I, Column 3).

D. Improving the Map from SLAM

We now turn to the results of performing an optimization on the network map by the SLAM algorithm using only the robot to beacon range measurements. By observing the result of this approach, we can evaluate the benefits of post-processing the data with performing an “online” estimation to incorporate the inter-beacon measurements with data from real sensors. After having performed the batch process on the map from the Kalman filter SLAM algorithm, we then again fed the resultant map into the localization algorithm to generate the results presented in Fig. 8 and Table I (Column 4).

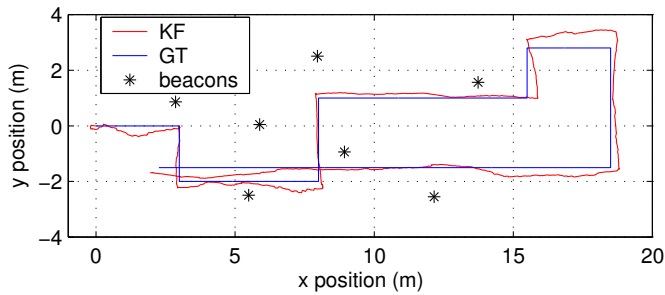


Fig. 8. The results from performing localization on the map from Kalman filter SLAM after the offline optimization step. The map from Kalman filter SLAM is used as initial conditions for the optimization process.

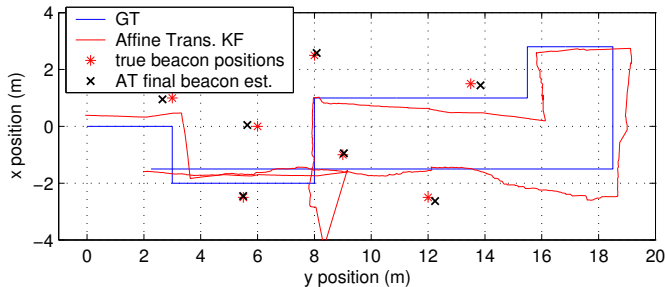


Fig. 9. Results from Kalman filter SLAM, with beacons initialized through the self-localization method with virtual nodes. The path and beacon estimates shown include an affine transform that re-aligns the local solution into a global coordinate frame. The sudden jumps seen in the Kalman filter result is an artifact that can be observed when the filter switches its behavior to not only use odometric information but also any range measurements it receives from any initialized beacon.

E. SLAM using Self-Localization with Virtual Nodes (VIR. N.)

As partially completed maps are generated (using the self-localization method with virtual nodes), the locations of the beacons are initialized into the Kalman filter SLAM algorithm, which then refines their estimates to produce a final map of the network (See Fig. 9). This resultant map is again fed into the localization algorithm to generate the numerical errors, Table I (Column 5), that can be compared with the results of the other approaches.

VI. SUMMARY

We have compared five methods on localization and mapping using sensors that are able to measure range. We show how the basic SLAM algorithm is able to locate the map of the nodes to the point that localization from that point on produces results that are on par with localization with exactly surveyed nodes. Experimental results validate the hypothesis

TABLE I

THE MEAN ABS. (μ) CROSS TRACK ERROR (XTE), STANDARD DEVIATION (σ) AND RMS ERROR OF VARIOUS METHODS (IN METERS).

	LOC.	SLAM ₁	SLAM ₂	IMP. MAP	VIR. N.	
XTE	μ	0.143	0.189	0.184	0.160	0.189
	σ	0.110	0.160	0.148	0.139	0.161
RMS	–	0.177	0.171	0.124	0.224	

that incorporation of inter-node measurements helps reduce drift as well as helps the map of the nodes converge faster (Figure 6 and 7). We show that a batch method that incorporates all (robot-node and inter-node) range measurements produces the best results in network localization as well as robot localization. The final method demonstrates that mobile beacons can make the network self-localization problem tractable in situations where node-to-node connectivity is low. This method is not as accurate as the others, but it is more amenable to distributed implementation on the sensor network.

REFERENCES

- [1] J. Leonard and H. Feder, "A computationally efficient method for large-scale concurrent mapping and localization," in *Robotics Research: The Ninth International Symposium*. Snowbird, UT: Springer Verlag, 2000, pp. 169–176.
- [2] R. Szeliski and S. B. Kang, "Recovering 3d shape and motion from image streams using nonlinear least squares," *Journal of Visual Communication and Image Representation*, vol. 5(1), pp. 10–28, March 1994.
- [3] I. Borg and P. Groenen, *Modern multidimensional scaling: theory and applications*. New York: Springer, 1997.
- [4] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM Press, 2004, pp. 50–61.
- [5] J. Leonard and H. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," in *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, Washington D.C., USA, May 1991.
- [6] J. Guivant, F. Masson, and E. Nebot, "Simultaneous localization and map building using natural features and absolute information," *Robotics and Autonomous Systems*, 2002.
- [7] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, and E. Nebot, "Fastslam: An efficient solution to the simultaneous localization and mapping problem with unknown data association," *Journal of Machine Learning Research*, 2004.
- [8] J. Folkesson, P. Jensfelt, , and H. Christensen, "Vision SLAM in the measurement subspace." in *Proceedings Intl Conf. on Robotics and Automation 2005*, Barcelona, ES, April 2005.
- [9] J. Djugash, S. Singh, and P. Corke, "Further results with localization and mapping using range from radio," in *Proceedings of FSR2005*, Cairns, Australia, August 2005.
- [10] P. Newman and J. Leonard, "Pure range-only sub-sea slam," in *Proceedings of IEEE Conference on Robotics and Automation*, Boston, USA, September 2003.
- [11] D. Hahnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose, "Mapping and localization with RFID technology," in *Proceedings of IEEE Conference on Robotics and Automation*, 2004.
- [12] A. M. Ladd, K. E. Bekris, A. Rudys, L. E. Kavvaki, D. S. Wallach, and G. Marceau, "Robotics-based location sensing using wireless ethernet," in *Proceedings of the 8th annual international conference on Mobile computing and networking*, 2002.
- [13] B. Hendrickson, "Conditions for unique graph realizations," *SIAM Journal of Computing*, vol. 21, no. 1, pp. 65–84, 1992.
- [14] B. Hendrickson, "The molecule problem: exploiting structure in global optimization," *SIAM Journal on Optimization*, vol. 5, pp. 835–857, 1995.
- [15] S. Capkun, M. Hamdi, and J.-P. Hubaux, "GPS-free positioning in mobile ad-hoc networks," in *HICSS*, 2001. [Online]. Available: citeseer.ist.psu.edu/article/capkun01gpsfree.html
- [16] L. Doherty, K. S. J. Pister, and L. E. Ghaoui, "Convex position estimation in wireless sensor networks."
- [17] N. B. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller, "Anchor-free distributed localization in sensor networks," MIT LCS, Tech. Rep. TR-892, April 2003.
- [18] E. Olson, J. Leonard, and S. Teller, "Robust range-only beacon localization," in *Proceedings of Autonomous Underwater Vehicles, 2004*, 2004.
- [19] W. Zhang, J. Djugash, and S. Singh, "Parrots: A range measuring sensor network," Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-06-05, in progress.