Constraint-Based Landmark Localization

Ashley W. Stroupe[1], Kevin Sikorski[2], and Tucker Balch[3]

[1] The Robotics Institute, Carnegie Mellon University,
Pittsburgh, Pennsylvania, USA
ashley@ri.cmu.edu
[2] Computer Science Department, University of Washington,
Seattle, Washington, USA
kws@cs.washington.edu
[3] College of Computing, Georgia Institute of Technology,
Atlanta, Georgia, USA
tucker@cc.gatech.edu

Abstract. We present an approach to the landmark-based robot localization problem for environments, such as RoboCup middle-size soccer, that provide limited or low-quality information for localization. This approach allows use of different types of measurements on potential landmarks in order to increase landmark availability. Some sensors or landmarks might provide only range (such as field walls) or only bearing measurements (such as goals). The approach makes use of inexpensive sensors (color vision) using fast, simple updates robust to low landmark visibility and high noise. This localization method has been demonstrated in laboratory experiments and RoboCup 2001. Experimental analysis of the relative benefits of the approach is provided.

1 Introduction

Localization is an essential component of many mobile robot systems. One of the most common approaches uses landmarks distributed throughout the environment to make estimates of position that are combined with a model of the robot's motion. Typically, sensing and odometry are combined using a Kalman-Bucy filter or a Monte Carlo approach. The popularity of landmark-based localization is due to its potential applicability to a large range of environments: indoor and outdoor, natural and artificial landmarks. Applying landmark localization to some environments because a challenge when few landmarks are available, when landmarks provide limited information, and when sensor measurements are noisy. It may also be difficult for dynamic environments when updates must occur quickly, as many approaches require complex updates. RoboCup middle-size soccer is an example of this type of environment.

The success of landmark-based localization relies on accurate sensing and adequate availability of landmarks. Traditional landmark navigation uses range and

bearing to any sensed landmark [1]. In indoor environments like RoboCup, walls can provide distance information without bearing in some cases. Examples include sonar and vision. In some vision systems, like ours, it may be impossible or impractical to determine bearing to the surface or identify specific points on the surface in order for a bearing to be useful. Goals, due to their large size and complex geometry, can provide useful bearing measurements but may produce unreliable distances. In outdoor environments, landmarks may be at large distances that cannot be measured accurately, yet they can still provide accurate bearing information. Bearing may also be available without accurate range in the event of partial occlusion of the landmark or oddly shaped landmarks. Some approaches have attempted to solve the problem of partial measurement availability by reducing the available measurements (such as always using only bearing) [2]. However, to make use of all available information, different types of landmarks must be accommodated.

The goal of this work is to provide a method of localization that is fast, mathematically simple, and takes advantage of different types of measurements on landmarks. We have attempted to expand the number of landmarks available to a robot by using only those measurements that are reliable for each landmark: range and bearing, range only, or bearing only. We have implemented the general method described here. Additionally, we applied our approach in our RoboCup 2001 middle-size team by developing a method for reliably using goals as bearing landmarks.

2 Background and Related Work

There are many approaches to landmark-based navigation. Prominent among these are triangulation, Kalman-Bucy filters, and Monte Carlo Localization (MCL) [1].

Triangulation depends on the range and bearing to landmarks and uses geometry to compute a single point that is most consistent with the current location [3]. This method is prevalent in human navigation approaches. Triangulation may be done with no weight given to previous positions, such as systems that use GPS, or may be incorporated with odometric information. Many robotic systems use approaches based on triangulation [4].

Kalman-Bucy filter approaches represent positions and measurements (odometry and sensing) probabilistically as Gaussian distributions. Position estimates are updated by odometry and sensing alternately using the property that Gaussian distributions can be combined using multiplication [5,6]. Typical implementations require range and bearing to landmarks and use geometry to determine the means of sensor-based position estimate distributions, though some implementations using only bearing exist [2]. While Kalman-Bucy filters are quite efficient, the additional computation of coefficients and covariance add to computation time; in some applications and on some platforms, reducing even this small time may be desirable.

Monte Carlo Localization is similar to Kalman-Bucy filter approaches, but represents the probabilistic distribution as a series of samples [7,8]. The density of samples represents the probability estimate that the robot is in a particular location. Distributions are updated by moving samples according to a probabilistic model of

odometry and then adding samples to regions consistent with sensing and dropping samples highly inconsistent with sensing. In one implementation, called Sensor Resetting Localization, the robot can reinitialize its position based on sensors when samples become highly improbable [9].

To resolve the difficulty in range measurements for outdoor environments, some work has been done using only bearing to localize [2]. In order to achieve pose estimates from bearing only, that work relies on a shape-from-motion type of formulation. This requires several frames and many simultaneously visible landmarks, and does not make use of range information when available. The SPmap approach allows use of different sets of measurements from different sensors, using all information from single sensors together, and takes into account landmark orientation [10]. This approach is based on Kalman-Bucy filters and thus requires covariance computation for position (and for landmarks for the mapping phase).

Our work has the mathematical simplicity of triangulation and uses all available landmark information, both bearing and range. Unlike many other approaches, landmarks that do not provide both range and bearing can be included without discarding useful information (as is range in bearing-only approaches). Bearing-only landmarks are incorporated in a single frame, rather than over several. Current implementation provides no estimate of position uncertainty, though the approach can be easily expanded to include it if computational demands permit.

3 Summary of Approach

The localization problem involves the use of odometry and measurements on landmarks to determine a robot's position and heading within an environment. Potential measurements on landmarks include range and bearing. The determination of a new estimate (position, heading, or pose) can be summarized as follows:

1. The robot estimate is updated after each movement based on the odometry model.
2. The robot senses available landmarks and computes an estimate consistent with sensing.
3. The estimates resulting from the movement update and from sensing are combined to produce a final estimate result.

Each type of measurement, bearing or range, provides an independent constraint on the position or heading of the robot. These constraints can either be a single-value constraint or a curve constraint. With single-value constraints, only one value of the position/heading is consistent with the constraint. With curve constraints, heading or position must be on a curve. Curves are typically circles or lines, but can be arbitrary.

For each sensor measurement of a landmark, a constraint is determined. A landmark may have multiple measurements, each of which will provide a separate constraint. For each measurement, the point consistent with the constraint that lies closest to the current estimate (position or heading) is assigned as the estimate for that measurement. Each constraint-based estimate is determined based on the original pose estimate; thus they are independent and can be combined in any order. The final

estimate (position or heading) is determined by combining the previous estimate and the estimate(s) from the sensor measurements.

In this work, uncertainty estimates are not required and all updates are performed using simple weighted averages. All sensor results are combined first and then this result is combined with the original estimate. In this way, order becomes irrelevant. Estimates in position and heading are alternately updated with a minimum time required between updates of the same type. This ensures independence of updates.

If uncertainty estimates are required, estimates can be expressed as probability distributions and integrated using probabilistic approaches.

4 Landmark Measurement Types

Three landmark measurement types are considered here: bearing to point, range to point, and range to surface. Each landmark provides one or more of these types. In each example, the robot has a pose estimate (light gray) (X_O,Y_O,θ_O) with actual pose (X,Y,θ) (dark gray). The landmark is at or passes through (X_L,Y_L). Range and bearing measurements are r and α, respectively. Sensor-based estimates are x', y', θ'.

4.1 Bearing to Point

A bearing to point measurement provides a linear constraint on position when heading is assumed known (Figure 1). This line passes through the landmark at (X_L,Y_L) at an angle of $\alpha+\theta_O$ (line slope is the tangent of this angle).

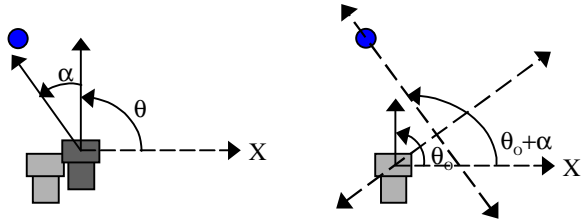

Figure 1. Left: A noisy bearing α is taken by a robot (true pose dark, estimate light). Right: Bearing and heading estimate create a line of possible consistent positions.

The point closest to the pose estimate is the intersection of the constraint line,

$$y = \tan(\alpha+\theta_O)x + Y_L - \tan(\alpha+\theta_O)X_L \quad (1)$$

and the line perpendicular to the constraint through the current position estimate:

$$y = \frac{-1}{\tan(\alpha+\theta_O)}x + Y_O + \frac{1}{\tan(\alpha+\theta_O)}X_O \quad (2)$$

This point can be computed as:

$$x' = \cos^2(\alpha+\theta_O)X_O + \sin^2(\alpha+\theta_O)X_L + \cos(\alpha+\theta_O)\sin(\alpha+\theta_O)(Y_O - Y_L) \quad (3a)$$

$$y' = \tan(\alpha+\theta_O)(x'-X_L) + Y_L \quad (3b)$$

with $y'=Y_O$ in the special case when $\alpha+\theta = \pm 90°$.

A single-value constraint on heading, when position is known, is also provided by bearing measurements (Figure 2).

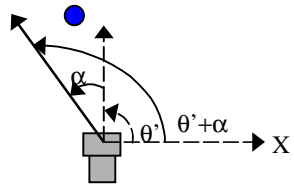

Figure 2. Using bearing measurement α and the angle determined by robot position estimate and landmark position, heading estimate θ' can be computed by subtraction.

Using position estimate, landmark position, and bearing measurement, the heading estimate can be determined as:

$$\theta' = \tan^{-1}\left(\frac{Y_L - Y_O}{X_L - X_O}\right) - \alpha \tag{4}$$

4.2 Range to Point

Range to a point provides a circular constraint on position (Figure 3). The constraint is centered at the landmark and is of radius r. Heading need not be known.

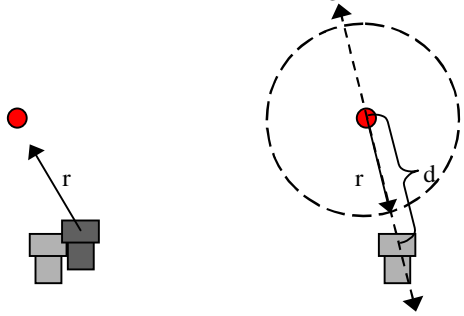

Figure 3. Left: A noisy range (r) is taken by robot (true pose dark, estimate light). Right: The range places the robot on a circle, radius r, around the landmark. The closest point on the constraint lies on the line from landmark to position estimate.

The closest point on the constraint is at the intersection of the circle and the line between landmark and position estimate. This is determined using similar triangles:

$$\frac{X_L - x'}{r} = \frac{X_L - X_O}{d} \Rightarrow x' = \frac{(X_O - X_L)r}{\sqrt{(X_L - X_O)^2 + (X_L - X_O)^2}} + X_L \tag{5a}$$

$$\frac{Y_L - y'}{r} = \frac{Y_L - Y_O}{d} \Rightarrow y' = \frac{(Y_O - Y_L)r}{\sqrt{(X_L - X_O)^2 + (Y_L - Y_O)^2}} + Y_L \tag{5b}$$

4.3 Range to surface

Range to a surface provides two-dimensional position constraints parallel to and on either side of the surface for position (Figure 4). Heading need not be known.

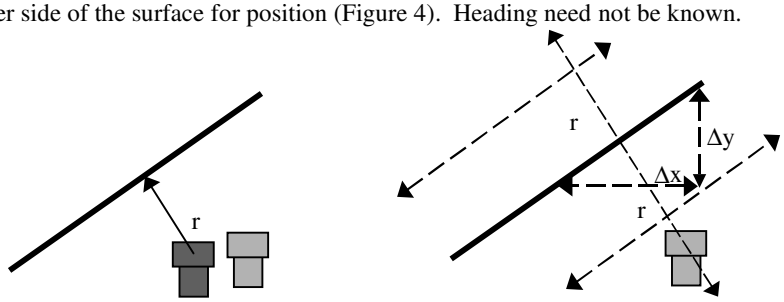

Figure 4. Left: The robot (true pose dark, pose estimate light) makes a noisy range measurement r on a wall. Right: The range places the robot on one of two lines parallel to the wall at that range (r,Δx,Δy), at the point closest to the estimate.

Given a wall projecting into the plane as a line through (X_L,Y_L) with slope m, two possible line constraints (on each side of the wall) are provided. These lines are shifted from the original by:

$$\Delta y = m\Delta x = \pm m\frac{r}{\sin(\tan^{-1}(m))} \tag{6}$$

resulting in the constraint lines:

$$y = mx + Y_L - mX_L \pm \frac{mr}{\sin(\tan^{-1}(m))} \tag{7}$$

The closest point on this constraint are computed as:

$$x' = mY_O + X_O - mY_L + m^2X_L \mp \frac{mr}{\sin(\tan^{-1}(m))} \tag{8a}$$

$$y' = mx' + Y_L - mX_L \pm \frac{mr}{\sin(\tan^{-1}(m))} \tag{8b}$$

Ambiguity is resolved either by eliminating one point as impossible (off the field, for example) or by choosing the result closer to the original estimate.

5 Implementation

The algorithm was implemented and applied in simulation and teams of real robots.

5.1 Software

The implementation of this localization method was integrated as part of *TeamBots* [13]. *CMVision* performs color analysis and color blob detection and merging on raw

camera images [14]. A landmark is represented by position, color, type of available measurements, minimum and maximum range and bearing at which landmark is reliable, and largest deviation allowed from expected range and bearing. Values are individually specified a priori for each landmark. In our experiments, all landmarks were predefined; however, the approach is amenable to adding mapped landmarks.

After odometry updates, sensor updates occur. There are four steps:

1. The robot obtains measurement data on all visible landmarks.
2. The robot computes each measurement's constraint.
3. The robot computes the closest point consistent with each constraint and combines these into a single sensor-based estimate X' by weighted average.
4. The final estimate is a weighted average of original and sensor-based estimates: $X_{k+1} = wX'+(1-w)X_k$.

Sensor updates are here illustrated in Figure 5 by a position update example. Position updates alternate with heading updates, which are computed similarly.

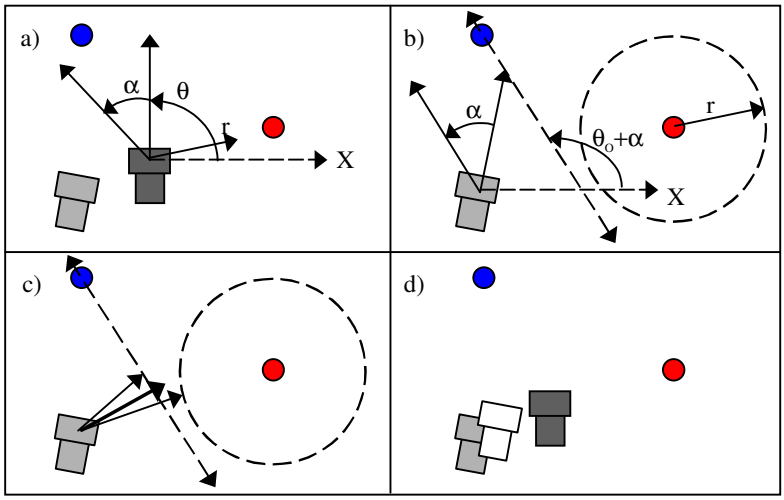

Figure 5. Position example. a) A robot (dark) with pose estimate (light) makes landmark measurements (one bearing, one range). b) Constraints matching sensors are computed. c) Closest constraint points are determined (thin arrows) and combined (wide). d) The final estimate (white) is a combination of former and sensor estimates.

5.2 Hardware Platform

The Minnow robot [15] is based on the Cye robot, an inexpensive and commercially available platform (Figure 6). The Cye consists of a drive section, with two-wheel differential drive, and a passive trailer. On the Minnow, a 700 MHz laptop communicates with the Cye's on-board processor via a serial link. High-level commands and image processing are implemented in C and Java on the laptop using *TeamBots*. Robots communicate using wireless Ethernet.

Figure 6. Two Minnow robots with colored block landmarks for localization.

A commercial USB camera provides images at 240 by 360 resolution. The camera is calibrated so that, at run-time, a selected image pixel is used to compute the ray from the camera to the point in space. Landmark ranges are determined from this ray by providing an intersection height, z; the point at which the ray intersects the given z value is the object's three-dimensional location (x,y,z). This provides range. Bearing is determined using the yaw angle of the ray directly or the angle produced by the position coordinates, atan(y/x). After color region processing using *CMVision*, pixels representing each region are chosen on which to compute range and bearing. Vision has not been implemented to determine distance to walls, but minimum distance obtained during a scan could be used (as the robot turns the camera, different sections of the wall would be visible, and a closest point could be found for each section). Vision cannot compute lines in the image, making bearing to walls unusable.

6 Quantitative Experimental Validation

To quantitatively evaluate this approach, a series of lab experiments were conducted.

6.1 Experimental Setup

Several small block-shaped objects were placed throughout an indoor environment that measures 7 by 5 meters (Figure 7). Each block was a different color that the robot's vision system was calibrated to recognize. Three experiments were conducted in which landmark measurements varied. Some landmarks were defined to provide range and bearing information while others were defined to provide only bearing.

A. Four landmarks providing range for position estimation and bearing for heading estimation;
B. Four landmarks providing range and bearing for position estimation and bearing for heading estimation;
C. Landmarks from B with four additional landmarks providing bearing for position and heading estimation.

Within the environment, the robot was commanded to follow a looping, piecewise-linear path defined by 12 waypoints (Figure 7). At each waypoint, the robot paused and reported its pose estimates from localization and from on-board odometry. During a pause, ground truth for pose was measured (position within 0.5 cm, heading within 1°). Each experimental run consisted of five loops (60 data points).

This experiment is intended to evaluate:

- The ability of this approach to localize a robot;
- The impact on performance resulting from the use of additional landmarks or measurements on landmarks that would not be available to approaches that rely on the same measurement(s) being available for all landmarks.

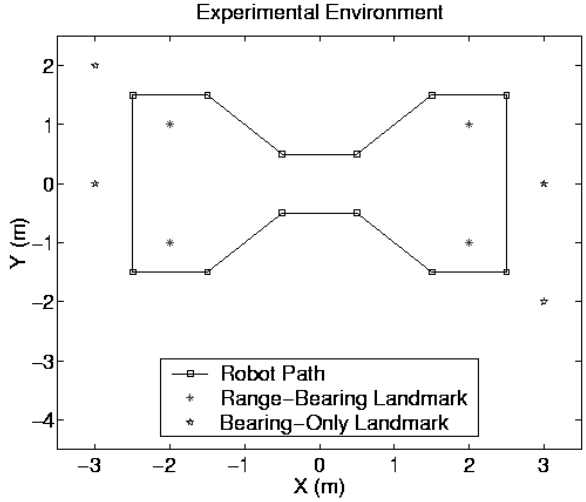

Figure 7. Experimental environment. Waypoints are shown along the robot's path. Landmarks differ by the type of measurements available for position updates.

6.2 Parameters

Sensor models and weights for combining estimates through averaging were determined experimentally. After calibration, it was empirically determined that range measurements produce errors on the order of 10-15% and bearing measurements produce errors on the order of 2°.

To update position or heading, first all sensor information is combined. If more than one measurement is used for a single landmark, the estimate from each measurement is weighted equally to determine that landmark's final estimate; in a more general approach, these could be weighted by individual measurement certainty. For combining all estimates from all landmarks, individual estimates are weighted (w) by an estimate of landmark quality ($1/\sigma^2$, where σ^2=10% of range).

The final result is a weighted average of previous estimate (updated for movement) and the sensor-based estimate. For position updates, the sensor-based estimate is

weighted based on the best (closest) landmark; if the closest landmark falls below a threshold, w = 0.2 is used and otherwise w = 0.05 is used. The threshold of 3.75 meters was experimentally determined based on range accuracy across distances. For heading estimates, which are less affected by changes in range to landmarks, w = 0.5 if more than half of the landmarks were visible and w = 0.2 otherwise.

6.3 Experimental Results

In each experiment, the total error in position, $\sqrt{\Delta x^2 + \Delta y^2}$, and heading was computed at each waypoint. From this the maximum and mean errors were obtained. Additionally, the standard deviation of absolute error in x and y were found. In Figure 8, the total error at each waypoint is plotted for the best odometry result and the worst localization test (4 landmarks with range only for position estimation). While position estimation initially has error greater than odometry, the magnitude of these errors does not increase unbounded over time as it does for odometry.

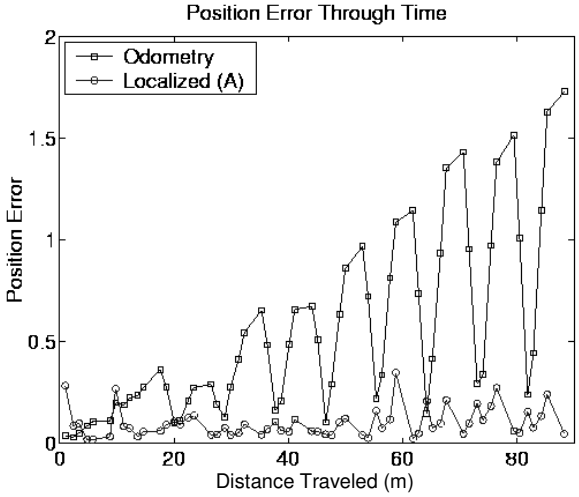

Figure 8. Experimental results for best odometry run and worst localization run. Total position error (m) as a function of distance (at each waypoint) is shown. Odometry errors grow with time while localization errors remain roughly constant.

Peaks in localization error occur when landmarks are furthest away from the robot. The cyclical nature of odometry error, with minima at loop ends, is most likely due to predominance of heading drift. Such errors may temporarily cancel, producing artificially accurate data.

The results for pose estimation in all experiments are shown in Table 1. Mean and maximum errors are shown with standard deviation in x and y position error.

This method of localization reduces absolute position error (mean, maximum, and standard deviation) and eliminates odometry drift. While the method for updating heading did not vary (except by number of landmarks for C) heading results

improved as a result of better position estimates: heading updates use position (Equation 4), thus position improvement leads to heading improvement. In these experiments, each addition of landmarks or measurement types to existing landmarks further improves performance.

Table 1: Pose Estimation Experimental Results

	Odometry (Best)	Localization (Range Only)	Localization (Range-Bearing)	Localization (Range-Bearing + Bearing Only)
Mean Position Error	0.5592m	0.0994m	0.0719m	0.0627m
$\sigma_{\Delta x}$	0.4378m	0.0822m	0.0578m	0.0562m
$\sigma_{\Delta y}$	0.5624m	0.0920m	0.0620m	0.0482m
Max Position Error	1.7275m	0.3459m	0.2326m	0.1652m
Mean Heading Error	29.6°	5.1°	2.3°	1.7°
Max Heading Error	42°	11°	5°	4°

7 RoboCup Middle-Size Soccer Implementation

A bearing-only approach was used to localize the robots for our middle-size RoboCup 2001 team, the CMU-Hammerheads. The most critical aspect of applying this approach to RoboCup is the accurate sensing of field landmarks.

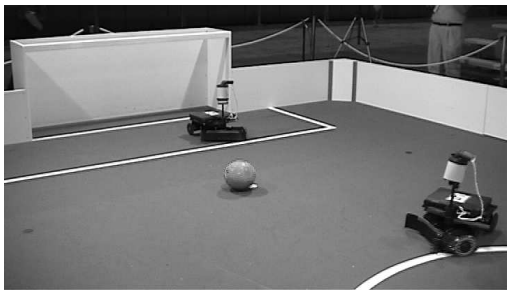

Figure 9. RoboCup middle-size field. Note that left side of goal is occluded by field wall and goal wall. The right side is visible but partially occluded by the goalie.

7.1 Landmark Sensing

For robots with only visual sensing, the RoboCup middle-size environment provides a challenge in accurately measuring landmark position and ranges. Features available on the field consist of:

- Small black squares on the walls. Walls may not be present in the future. The small squares are not visible from many points on the field.
- Field walls. Ambiguous landmarks which (as a surface) cannot alone provide accurate pose.
- Field lines. Complex methods of line fitting are required to accurately position the robot with respect to field lines.
- Goals. Goals are large and brightly colored and thus easy to sense (Figure 9). However, it is difficult to get a range to a large and concave object, as some parts may be closer than others or may be occluded.

Field goals, easily identified visually from any distance, are the obvious landmark choice. Due to the size and concavity of the goal, it is often difficult to determine which point on the goal is being observed or to get an accurate range measurement. In the example shown in Figure 10, the edge of the detected goal represents the front edge of the goal (left robot) or a variable point on the back of the goal (right robot), resulting in inconsistent ranges and inconsistent landmark reference position. However, the angle to the front corners can be reliably determined, even when not directly observable. As shown, even in the instance when the back of the goal is detected, this angle is coincident with the angle to the front corner.

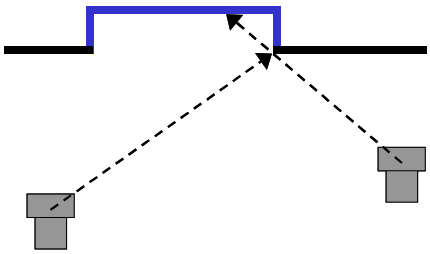

Figure 10. The angle from robot to visible edge of the goal is measured to the same point on the goal from all robot positions, despite the concavity. Range varies.

As goal corner angles can be reliably detected, they are amenable to use as bearing-only landmarks. The angles to each edge of the goal can be independently determined, providing four landmarks on the field.

A second difficulty that arises frequently in RoboCup middle-size soccer is occlusion (Figure 9). With up to 8 robots on the field, two of which are specifically tasked with blocking the goals, occlusion of part of the goal is common. It was necessary to determine when the edge of the goal was visible in order to known when the landmark was measured accurately.

In order to deal with occlusion, we took advantage of the color-coded aspects of the RoboCup world. When the sensed goal is observed adjacent to another color (such as robot, robot marker, ball, etc), occlusion is present. Points on the edge of the goal that are adjacent to the occlusion are thrown out. Once valid edge points have been selected, the median bearing is computed. Only the top ten valid points are used for efficiency and to minimize occlusion, which occurs at ground level. The median

is used to reduce sensitivity to noise. It is this value that is used as the bearing to the landmark. If no valid points are found, the landmark is not used.

7.2 Parameters

Using bearing only, resulting heading estimates are more accurate than position estimates. Small errors in heading and visual angle to landmarks produce small errors in heading estimate but can produce large errors in position at larger distances from the landmark. Thus, a smaller weight, a, is assigned to sensor estimates for position than for heading when combining with the previous estimate. For both heading and position, a larger weight was used when more landmarks were visible (2 or more). For heading, w = 0.5 or 0.2. For position, w = 0.2 or 0.1.

7.3 Performance

It is difficult to quantify performance at RoboCup, as ground truth data cannot be obtained during play. Qualitatively, the use of this localization method greatly improved performance during practice and play, as observed primarily in the "go home" behavior. Without localization, the robots would achieve greater than 1-meter error in position and 20-degree error in heading within a few minutes of play. With localization, pose was typically maintained with less than 0.5m and a few degrees error. Occasionally, greater errors would arise briefly due to the high percentage of time when landmarks were occluded by the many robots on the field. Accuracy of 0.5m is adequate for the soccer application, which is highly reactive and large-scale.

7.4 Simulation Experiments

Difficulties in generating quantitative results on the soccer field, including not having a real field in our lab, necessitated simulation. The TeamBots simulator was used with block landmarks positioned at goal corner locations. To simulate reduced landmark visibility (occlusion), field of view was limited to 5 meters (about half the field). Simulation parameters for noise standard deviations were: right and left wheel velocities = 5%, vision range = 5%, vision angle = 5º (slightly higher than actual). Robots followed the path of Figure 7 for five 5-loop trials. Table 2 shows results comparing performance using only landmark bearing to that using range and bearing.

Results indicate that even with only bearing information, the robot is consistently able to remain relatively well localized with frequent (though not constant) landmark observations; errors do not continuously grow over time as in odometry. Eliminating range information slightly reduces performance in position (increasing magnitude of the largest outliers) and does not adversely affect heading estimation. The magnitude of errors agrees with the qualitative results obtained by observing the robot on the real mid-size field over the short time represented by the 5-loop simulation.

Table 2: Pose Estimation Simulation Results: Soccer Field

	Odometry	Localization (Bearing)	Localization (Range-Bearing)
Mean Position Error	0.2536m	0.1714m	0.1068m
$\sigma_{\Delta x}$	0.1057m	0.0929m	0.0500m
$\sigma_{\Delta y}$	0.1852m	0.0886m	0.0558m
Max Position Error	1.0353m	0.5500m	0.2613m
Mean Heading Error	4.0262°	2.2328°	2.0938°
$\sigma_{\Delta t}$	2.9382°	1.8450°	1.6631°
Max Heading Error	15.7897°	8.5425°	8.4449°

8 Discussion

This localization method, using a weighted average, keeps robot navigational error bounded within the limits of sensor accuracy when provided with adequate landmarks. Initial position estimates may show slightly higher error than pure odometry. Noise in sensing leads to immediate small estimate errors while odometry errors accumulate over time. However, the localization method presented prevents drift evident in odometry and maintains a consistent, bounded level of error.

The ability to use different types of measurements on landmarks may expand the number of landmarks available to a robot and thereby improve localization performance, as shown in these experiments. Successful bearing-only approaches require several frames and many landmarks to compute pose. Our approach incorporates bearing-only landmarks in a single-frame update. Range information is accommodated when available and few concurrently visible landmarks are required.

If only range measurements are used, no corrections can be made on robot heading. If only bearing measurements are used, this approach cannot solve all "lost robot" problems. The linear constraints on position can only provide corrections to position within limits. When heading estimates are highly incorrect, the line constraint falls far from truth and provides very erroneous results. An example is in Figure 11; the constraint pulls the robot's estimate further from the actual position.

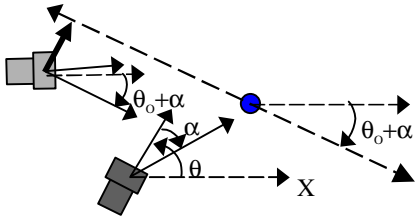

Figure 11. A robot measures landmark bearing from it's actual location (dark). The robot's erroneous pose estimate (light) with a correct bearing produces an erroneous constraint line which pulls the position estimate further away from truth (wide arrow).

If both bearing and range measurements are available, from the same or different landmarks, this approach can converge to the correct answer from any initial condition. While no formal proof is offered, it has been experimentally determined (in simulation and on real robots) that if three landmarks can provide each type of measurement, the robot is able to eventually recover from any errors. If fewer measurements are available, the ability to recover may be limited.

This simple approach to localization is useful when computational time is extremely limited. In some applications with some platforms, such as the Minnow in RoboCup, saving even the small amount of additional time required to compute covariance and proper Kalman-Bucy updates might be desirable. However, when computational constraints are relaxed, this approach may provide a useful method for generating probabilistic distributions for Bayesian or Monte Carlo updates even with noisy and limited information.

9 Conclusions

We provide a fast, efficient, method of mobile robot localization that takes advantage of the different types of measurements made on landmarks to improve landmark availability. This method prevents odometry drift. With adequate landmarks, it can recover from being lost. While performance improves with additional measurements, few simultaneously visible landmarks are required for reasonable updates.

This method takes advantage of different types of measurements that may be available with different types of landmarks. Unlike approaches that preclude use of valid measurements to eliminate noisy measurements, we use all viable data and complete updates in a single frame. Cooperative localization may be performed by using teammates as landmarks or updating with measurements made by teammates.

Our approach can be used in the simple manner described, using weighted averages, or can be used to provide input for generating samples for Monte Carlo localization or a Gaussian distribution for Kalman-Bucy filter localization. We recognize that a statistical approach such as these may provide more accurate pose estimates. However, combination by weighted average is extremely simple and fast computationally, which is an important feature for real-time, dynamic domains such as RoboCup.

The described method provides a simple method of localization that is applicable to RoboCup middle-size soccer, despite many difficulties in achieving accurate measurements on visual landmarks in the RoboCup middle-size environment. This is accomplished by determining a means by which the angles to goal corners can be reliably measured and used as bearing landmarks.

A probabilistic implementation with collaboration (based on [11] and [16]) has been implemented in simulation and will be applied to our RoboCup legged team in 2002. For curve constraints, Gaussian distributions are aligned tangent to the curve with means computed in the same manner as the approach described here. For single value constraints, Gaussian distributions are centered at this value. The sensor model determines variances of these Gaussian distributions. The probabilistic updates use

Gaussian multiplication [12] to compute mean and covariance. This type of approach was used for the CMU 2001 legged team [11]. Updates could also be done using Kalman-Bucy filter updates, (also approximating estimates with Gaussian distributions). Similarly, samples representing arbitrary distributions can be generated for Monte Carlo Localization using the constraint and sensor models.

Acknowledgements

The authors wish to acknowledge the contributions of Jim Bruce, Scott Lenser, Hans Moravec, and Manuela Veloso. We also wish to acknowledge DARPA's Mobile Autonomous Robot Software and Control of Agent Based Systems programs and the Northrop Grumman Corporation for support of this research.

References

1. Borenstein, J. Everett, H.R., Feng, L.: Where am I? Sensors and Methods for Autonomous Mobile Robot Positioning. Technical Report, U Michigan (1996)
2. Deans, M., Hebert, M.: Experimental Comparison of Techniques for Localization and Mapping using a Bearings Only Sensor. Proc ISER '00. IEEE (2000)
3. Graydon, D., Hanson, K. (eds.): Mountaineering: The Freedom of the Hills. 6th ed, (1997)
4. Rekleitis, I., Dudek, G., Milios, E.: Multi-Robot Collaboration for Robust Exploration. Proc 2000 IEEE ICRA (2000)
5. Smith, R.C., Cheeseman, P.: On the Representation and Estimation of Spatial Uncertainty. International Journal of Robotics Research. 5 (1998) 5:56-68
6. Leonard, J., Durrant-Whyte, H.: Mobile Robot Localization by Tracking Geometric Beacons. IEEE Transactions on Robotics and Automation, 7 (1991) 3:376-382
7. Dellaert, F., Fox, D., Burgard, W., Thrun, S.: Monte Carlo Localization for Mobile Robots. Proc IEEE ICRA. IEEE (1999)
8. Fox, D., Burgard, W., Thrun,S.: Markov Localization for Mobile Robots in Dynamic Environments. Journal of Artificial Intelligence Research. 11 (1999)
9. Lenser, S., Veloso, M.: Sensor Resetting Localization for Poorly Modeled Mobile Robots. Proc IEEE ICRA, 2000. IEEE (2000)
10. Castellanos, J., Montiel, J., Neira, J., Tards, J.: The SPmap: A probabilistic Framework for Simultaneous Localization and Map Building. IEEE Transactions on Robotics and Automation. 15 (1999) 5:948-952
11. Uther, W., Lenser, S., Bruce, J., Hock, M., Veloso, M.: CM-Pack'01: Fast Legged Robot Walking, Robust Localization, and Team Behaviors. In: RoboCup-2001: Robot Soccer World Cup V, A. Birk, S. Coradeschi, S. Takodoro (eds.), Springer-Verlag (2002)
12. Stroupe, A.W., Martin, M.C., Balch, T.: Distributed Sensor Fusion for Object Position Estimation by Multi-Robot Systems. Proc IEEE ICRA, 2001. IEEE (2001)
13. Balch, T.: TeamBots. http:// www.cs.cmu.edu/~trb/TeamBots.
14. Bruce, J.: CMVision Realtime Color Vision. http://www-2.cs.cmu.edu/~jbruce.
15. Balch, T.: The Minnow Project. http://www-2.cs.cmu.edu/~coral/minnow.
16. Stroupe, A., Balch, T.: Collaborative Probabilistic Constraint-Based Landmark Localization. Submitted to Proc IROS 2002.