Third National Conference on Robotics
Melbourne 3-6 June 1990

Iterative Algorithm for Three Dimensional Autonomous Robot Localisation

Dr. L. Kleeman

Abstract

This paper[1] derives and analyses an algorithm for use in a low cost three dimensional ultrasonic localisation system under construction at Monash University. The system is intended for autonomous robot navigation in three dimensions, such as in under-water applications. At least four ultrasonic beacons are required to solve for the position in three dimensions. The system is an extension of a two dimensional version successfully designed and constructed by the author [1,2].

In this paper the algorithm for determining position is derived from first principles. The algorithm is suitable for an autonomous robot because the implementation requires no electrical connection to the vehicle. The error and convergence performance of the algorithm is presented with computer simulation results. The algorithm converges within a few iterations and exhibits good solution conditioning.

Keywords: Localisation, autonomous vehicle, three dimensional, ultrasonic, algorithm.

1. Introduction

The work presented in this paper was motivated by the recent successful design and construction of a *two* dimensional ultrasonic localisation system [1,2]. The two dimensional system uses a set of ultrasonic beacons at known positions in a plane in the environment. The beacons transmit ultrasonic chirps of 40 kHz in a fixed time sequence. On board an autonomous robot is an eight element ultrasonic receiver which monitors incoming chirps. From the fixed known time between beacon chirps and the speed of sound, which is calibrated for in the initialisation phase, arrival time differences of pulses are converted into distances differences to the beacons. The distance differences from three beacons allow solution for the robot's position in a plane by the intersection of hyperbolas. By employing six beacons, the effects of obstacles and reflections can be significantly reduced, with the bonus of calibration for the speed of sound.

The three dimensional system described in this paper uses similar beacon and receiver structures to the two dimensional system. However, the algorithm for solving for position is significantly different. It will be shown that *four* beacons are required to solve for a three dimensional position by the intersection hyperboloids. To produce a closed form solution is impractical, and an iterative technique is proposed which converges rapidly.

The system has applications in underwater maintenance of oil rig structures, pipelines, ships and deep sea installations. Sonar beacons could be employed underwater, where the speed of sound is approximately an order of magnitude faster than that of air.

Localisation systems reported in the literature have various disadvantages for autonomous operation. MELODI [3] is an ultrasonic system using on board angular detection of three different frequency transmitters in the environment. The system is susceptible to obstacles blocking beacons; accurate to only 1 part in 80; requires rotating

[1]Supported by a grant from Monash University Research Fund

arms on the receiver and is expensive. Another system [4] is impractical because it requires a distributed ultrasonic microphone to receive waves orthogonally from a spark gap transmitter. An ultrasonic system [5] for tracking robot manipulators provides accuracies of 0.5 mm in 2000 mm at a sample rate of 300 Hz. However, this system requires a high speed connection from the robot to the environment for synchronisation. No such link can be trailed behind an autonomous robot. The Loran-C [6] ship and aircraft navigation system is similar to the two dimensional system developed by the author, but uses different beacon patterns and transmission formats.

The paper is organised as follows. Section 2 reviews the beacon and receiver structure to be employed in the three dimensional localisation system. This gives the reader an appreciation of the data available to the position determination algorithm which is derived in Section 3. Section 4 gives an error estimate of the position given the measurement errors. In Section 5 the performance of the algorithm is evaluated and results presented from computer simulations. Conclusions and future work are presented in Section 6.

2. Beacon and Receiver Structure

This section describes the design and organisation of the beacons and the receiver on board the autonomous vehicle. The arrangement described here is designed to operate in air which is where the first 3D prototype will operate. The air transmitter and receiver circuitry has been built and tested. An underwater system will require sonar beacons and must also allow for the faster speed of propagation of sound underwater.

Since the robot aims to be autonomous, no wires between the robot and the environment can be allowed. This requirement complicates the algorithmic design since the times of transmissions cannot be determined in advance by the receiver. Also, the autonomy of the robot requires that the position information be produced on the robot and not in the environment. Thus, the receiver is located on the robot.

The localisation system consists of ultrasonic beacons located in the environment and an intelligent receiver on board the robot as shown in Fig. 1. The beacons send pulses in strict time sequence. The first beacon needs to be distinguished initially to allow the receiver to correctly identify which beacon corresponds to a received pulse. This is achieved by sending a double pulse from the first beacon with known pulse widths and separation. During initialisation or resynchronisation of the system, these double pulses are sought out. Note that there is no wire connection or communication, other than ultrasonic sound, between the beacons and receiver.

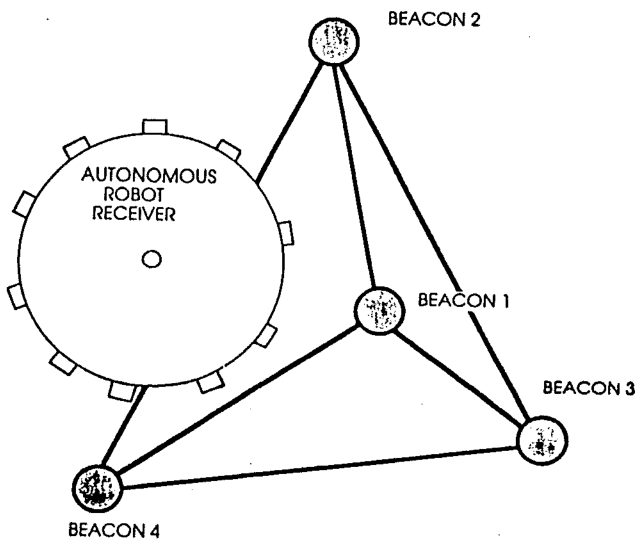

Fig. 1 Overview of Localisation System.

The beacons' pulse sequence is controlled by a central beacon control module that sends signals to each beacon on a common 4 wire telephone cable. Each beacon circuit is identical, with its sequence number set by DIP switches. Fig. 1 shows 4 beacons, however this number may be varied up to 16 (prototype hardware limitation). Each beacon contains circuitry to sense its turn to transmit and an asynchronous start 40 Khz oscillator to drive an ultrasonic transmitting device.

The transmitter control module is responsible for generating signals to correctly sequence the firing of the beacons. In Fig 2, the timing of beacon firing is shown for the 4 beacon configuration to be employed in the prototype. Note that beacon 1 is identified by the receiver detecting a double pulse. Each pulse corresponds to a chirp of 40 Khz ultrasonic sound.

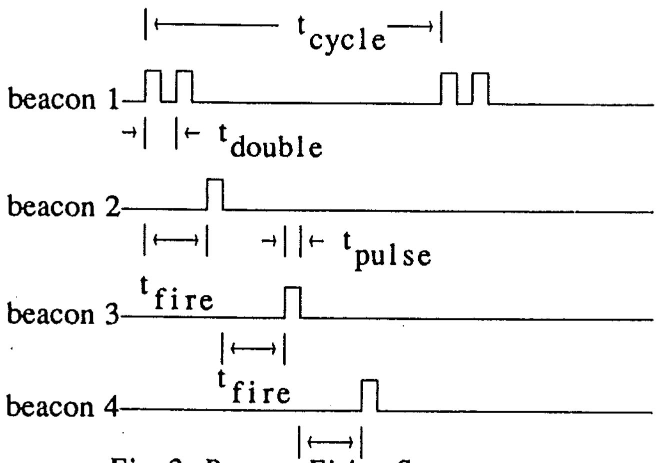

Fig. 2 Beacon Firing Sequence.

The receiver consists of ultrasonic receiving devices placed evenly on a sphere. The signals from the devices are amplified and filtered before being analysed by a microprocessor as shown in Fig. 3. The microprocessor extracts times of arrival and identity of pulses. Distance is determined from the speed of sound. The position can then be calculated, as described below, from data which are the distances to beacons plus an unknown offset.

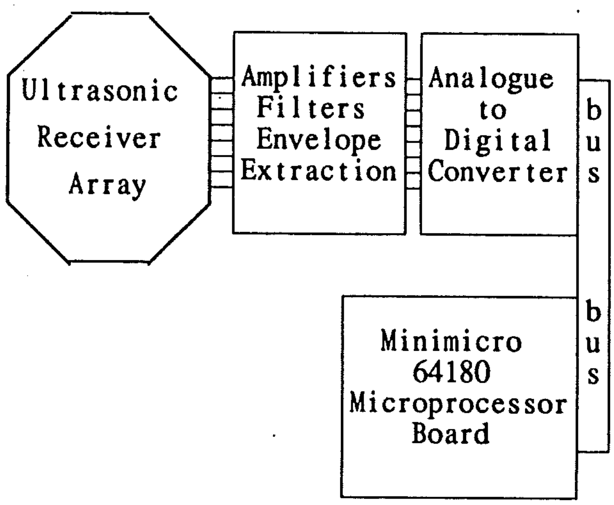

Fig. 3 Receiver Block Diagram.

The receiver array should be approximately spherical so that the same distance offset applies in any direction. A constant offset in measurements from all beacons has no effect on the final result, since the solution is based only on differences in distance to the beacons. The polarity of each receiver should be the same to prevent errors of half a wavelength or 4mm.

3. Three Dimensional Position Determination Algorithm

In this section we derive a three dimensional position determination algorithm which can be used with the data available from the hardware described in the previous section. The distance data obtained from beacons cannot be used directly since it contains an unknown offset. To eliminate this offset, differences in distances to beacons are used. The locus of points defined by a constant difference in distance to two beacons in a plane is a hyperbola. To obtain the three dimensional locus of points, the plane is rotated about a line connecting the two beacons. The rotation does not change the distances to the beacons. The resulting surface is a *hyperboloid.* It has the property that for all points P on its surface, $\|\mathbf{Pb}_1\| - \|\mathbf{Pb}_2\|$ is constant, where $\mathbf{Pb}_1$ is the displacement vector from the beacon b_1 to the point P, and $\| . \|$ is the Euclidean norm. Given data from two beacons, we know our solution lies on the hyperboloid defined by the measured difference in distance.

Incorporating the additional information from another beacon, b_3, narrows the locus of solutions to a three dimensional curved line given by the intersection of the hyperboloids defined by (b_1,b_2) and (b_1,b_3). Note that the hyperboloid defined by (b_2,b_3) adds no extra information and therefore must intersect in the same line.

A fourth beacon is required to uniquely specify the position. The hyperboloid defined by (b_1,b_4) intersecting the line defines the position.

The algorithm derived to solve for the position is iterative. Each iteration is obtained by linearising the hyperboloids about the last iteration position. This is achieved by replacing the hyperboloids with their tangent planes.

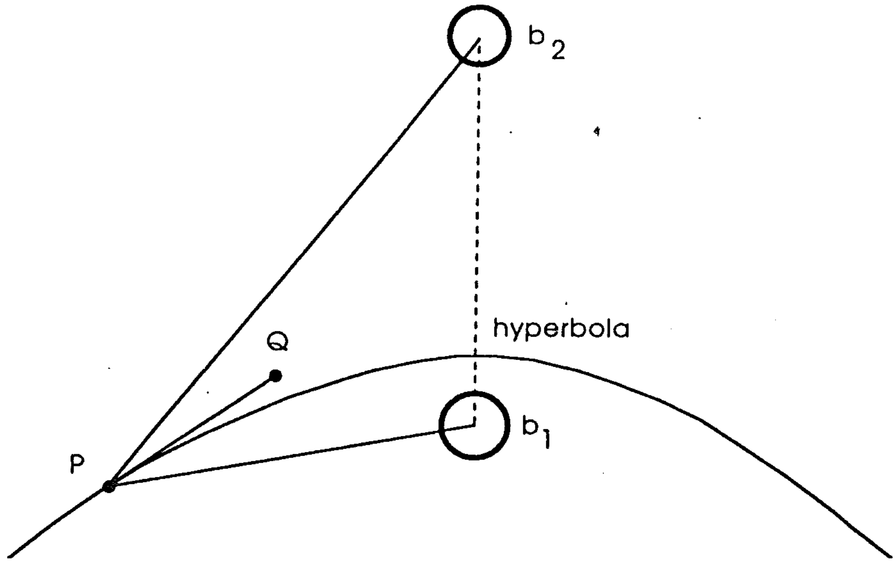

Fig. 4 Receiver and Beacon Geometry.

Consider two beacons b_1 and b_2 as shown in Fig. 4, where P is the estimated solution point to be updated. The vector $\mathbf{PQ}_{12}$ is the tangent to the hyperbola of constant difference in distance to b_1 and b_2 in the plane Pb_1b_2. In [2], it is shown that $\mathbf{PQ}_{12}$ bisects the angle $\angle b_1Pb_2$ giving

$$\mathbf{PQ}_{12} = \frac{\mathbf{u}_1 + \mathbf{u}_2}{2} \tag{1}$$

where $\text{Unit}\left[\mathbf{V}\right] = \frac{\mathbf{V}}{\| \mathbf{V} \|}$ and $\mathbf{u}_1 = \text{Unit}\left[\mathbf{Pb}_1\right]$

The tangent plane to the hyperboloid passing through P has a normal vector, $\mathbf{N}_{12}$, which is perpendicular to $\mathbf{PQ}_{12}$ and lies in the plane Pb_1b_2. The normal to the plane Pb_1b_2 is given by the vector cross product $\mathbf{Pb}_1 \times \mathbf{Pb}_2$. Therefore

$$\mathbf{N}_{12} = \text{Unit}\left[\left[\mathbf{u}_1 \times \mathbf{u}_2\right] \times \mathbf{PQ}_{12}\right] \tag{2}$$

Substituting (1) in (2), removing scalars within Unit operations and rearranging using the vector identity:

$$(\mathbf{A} \times \mathbf{B}) \times \mathbf{C} = \mathbf{B}\,(\mathbf{C} \bullet \mathbf{A}) - \mathbf{A}\,(\mathbf{B} \bullet \mathbf{C})$$

(where $\bullet$ is the vector dot product) gives:

$$\begin{aligned}\mathbf{N}_{12} &= \text{Unit}\left[(\mathbf{u}_2 - \mathbf{u}_1)(1 + \mathbf{u}_2 \bullet \mathbf{u}_1)\right] \\ &= \text{Unit}(\mathbf{u}_2 - \mathbf{u}_1)\end{aligned} \tag{3}$$

The direction of the tangent to the line of intersection of two hyperboloids defined by Pb_1b_2 and Pb_1b_3, $\mathbf{V}_{123}$, is the line of intersection of the tangent planes defined by normals $\mathbf{N}_{12}$ and $\mathbf{N}_{13}$. Looking edge on to both planes, as in Fig. 5, it is clear that

$$\mathbf{V}_{123} = \text{Unit}\left[\mathbf{N}_{12} \times \mathbf{N}_{13}\right] \tag{4}$$

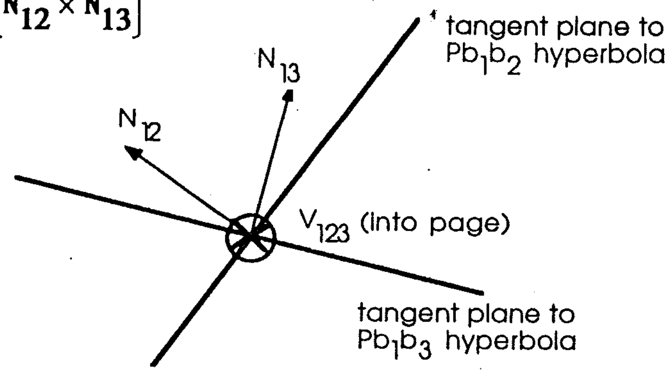

Fig. 5 Edge on View of Intersecting Tangent Planes.

Adding a fourth beacon will define the position. The question is how far should we move along $\mathbf{V}_{123}$ to satisfy the measured difference in distance to b_1 and b_4. If we move δr, then the change in difference in distance to b_1 and b_4, δd_{14} is given by

$$\delta d_{14} = \delta r_{14}\, \mathbf{V}_{123} \bullet (\mathbf{u}_1 - \mathbf{u}_4) \tag{5}$$

Therefore

$$\delta r_{14} = \frac{\delta d_{14}}{\mathbf{V}_{123} \bullet (\mathbf{u}_1 - \mathbf{u}_4)}$$

$$= \frac{\delta d_{14}}{\mathrm{Unit}(\mathbf{g}_{12} \times \mathbf{g}_{13}) \bullet \mathbf{g}_{14}} \qquad (6)$$

Where $\mathbf{g}_{12} = \mathbf{u}_1 - \mathbf{u}_2$. We can do the same analysis for δr_{13} and δr_{12}. To produce the distance iteration vector, $\boldsymbol{\delta r}$, we can add together the three displacements since each changes only one of d_{14}, d_{13} or d_{12} and leaves the other two fixed.

$$\boldsymbol{\delta r} = \delta r_{14}\mathbf{V}_{123} + \delta r_{12}\mathbf{V}_{134} + \delta r_{13}\mathbf{V}_{142} \qquad (7)$$

Substituting (6), (4) and (3) in (7) and applying the vector identity

$$(\mathbf{A} \times \mathbf{B}) \bullet \mathbf{C} = (\mathbf{B} \times \mathbf{C}) \bullet \mathbf{A} = (\mathbf{C} \times \mathbf{A}) \bullet \mathbf{B}$$

gives:

$$\boldsymbol{\delta r} = \frac{\left[\delta d_{12}(\mathbf{g}_{13} \times \mathbf{g}_{14}) + \delta d_{13}(\mathbf{g}_{14} \times \mathbf{g}_{12}) + \delta d_{14}(\mathbf{g}_{12} \times \mathbf{g}_{13})\right]}{(\mathbf{g}_{12} \times \mathbf{g}_{13}) \bullet \mathbf{g}_{14}} \qquad (8)$$

We need three components δr_{12}, δr_{13} and δr_{14} to give the three degrees of freedom necessary to solve for an arbitrary position given a starting point. We add $\boldsymbol{\delta r}$ to our estimated position and recalculate a new $\boldsymbol{\delta r}$ based on our new position. The iterative procedure terminates when $\|\boldsymbol{\delta r}\|$ is sufficiently small.

The author has shown that equation (8) can be rewritten in a form that is symmetric with respect to the indices 1,2,3 and 4. That is, the iterative algorithm is not dependent on choosing beacon 1 as a reference as may initially appear to be the case in (8).

4. Estimation of Solution Error given Measurement Error

Suppose we have a maximum measurement error of ε in each of d_{14}, d_{13} and d_{12} (where $d_{ij} = d_i - d_j$ and d_i is the distance from the robot to beacon i). The maximum solution error, E, given that our iterative technique converged, is derived again by linearising about the solution point.

$$E = \varepsilon \,\mathrm{MAX}\left\{ \frac{\left[(\mathbf{g}_{13} \times \mathbf{g}_{14}) \pm (\mathbf{g}_{14} \times \mathbf{g}_{12}) \pm (\mathbf{g}_{12} \times \mathbf{g}_{13}) \right]}{(\mathbf{g}_{12} \times \mathbf{g}_{13}) \bullet \mathbf{g}_{14}} \right\} \qquad (9)$$

where the maximum is over the $\pm$ and in terms of the $\|.\|$ operator. These vectors are all available directly from the last iteration of the position estimation algorithm, so require no extra calculation.

5. Performance of the Iterative Hyperboloid Intersection Algorithm

In order to study the convergence properties of the algorithm, a hypothetical beacon pattern was chosen as shown in Fig. 6.

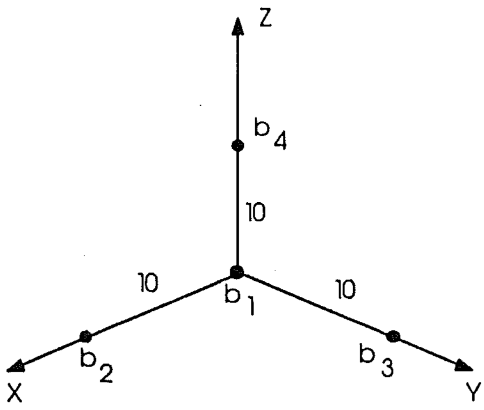

Fig. 6 Simulation Beacon Pattern.

Convergence is assured in practice provided the iteration points all stay within the positive octant of the coordinate system. Once the coordinates become negative, the tangent plane to the hyperboloid points away from the solution causing divergence. The speed of convergence is illustrated by the plotting the distance from the goal point versus the iteration number. In Figs. 7–9 random start points were chosen by a uniform distribution on each axis normalised to 3 units from the goal point. Four goal points are shown with differing convergence rates. Figs. 7,8,9 show the maximum, minimum, and mean errors of 1000 convergence runs. Five orders of magnitude error improvement can be seen in all cases after 4 iterations. Each iteration requires 4 square roots, 13 divisions, 33 multiplications and 47 additions or subtractions, all single precision floating point. Each iteration takes 300 μsec on a T800 20 MHz Transputer, 433 μsec on 33 MHz 80386/80387 and 55 msec on 9 MHz Hitachi 64180 8 bit microprocessor.

Fig. 10 shows the performance for random start points and goal points. The goal points were chosen uniformly within the cube with diagonal (2,2,2),(12,12,12) and start points in a random direction 3 units from the goal point. The maximum error plot on the first iteration appears to diverge initially, but this is in fact due to the iteration point moving out of an ill conditioned position near an axis.

From the results, it is clear that only a few iterations are necessary to converge. The starting point in a real system will usually be a good approximation to the position for slowly moving vehicles.

Table 1 shows the maximum length standardised error vectors for various goal points. To evaluate actual errors these vectors are scaled by the difference in distance error ε. The largest error vector occurs from a goal point (30,30,30) which is a large distance from the beacons. The variation in the iteration 5 error in Figs. 7–10 is due to differing error vectors and rounding errors.

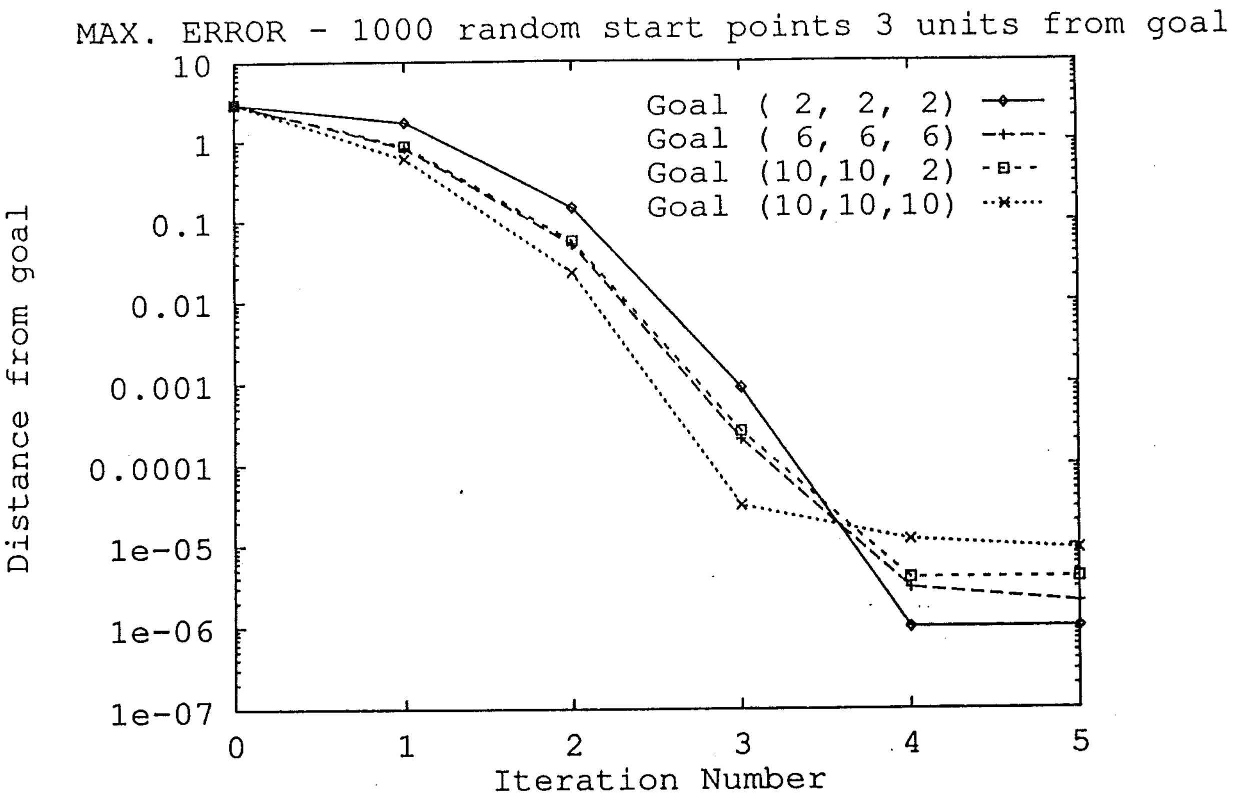
MAX. ERROR - 1000 random start points 3 units from goal
Goal (2, 2, 2)
Goal (6, 6, 6)
Goal (10,10, 2)
Goal (10,10,10)
Distance from goal
10
1
0.1
0.01
0.001
0.0001
1e-05
1e-06
1e-07
0
1
2
3
4
5
Iteration Number

Fig. 7

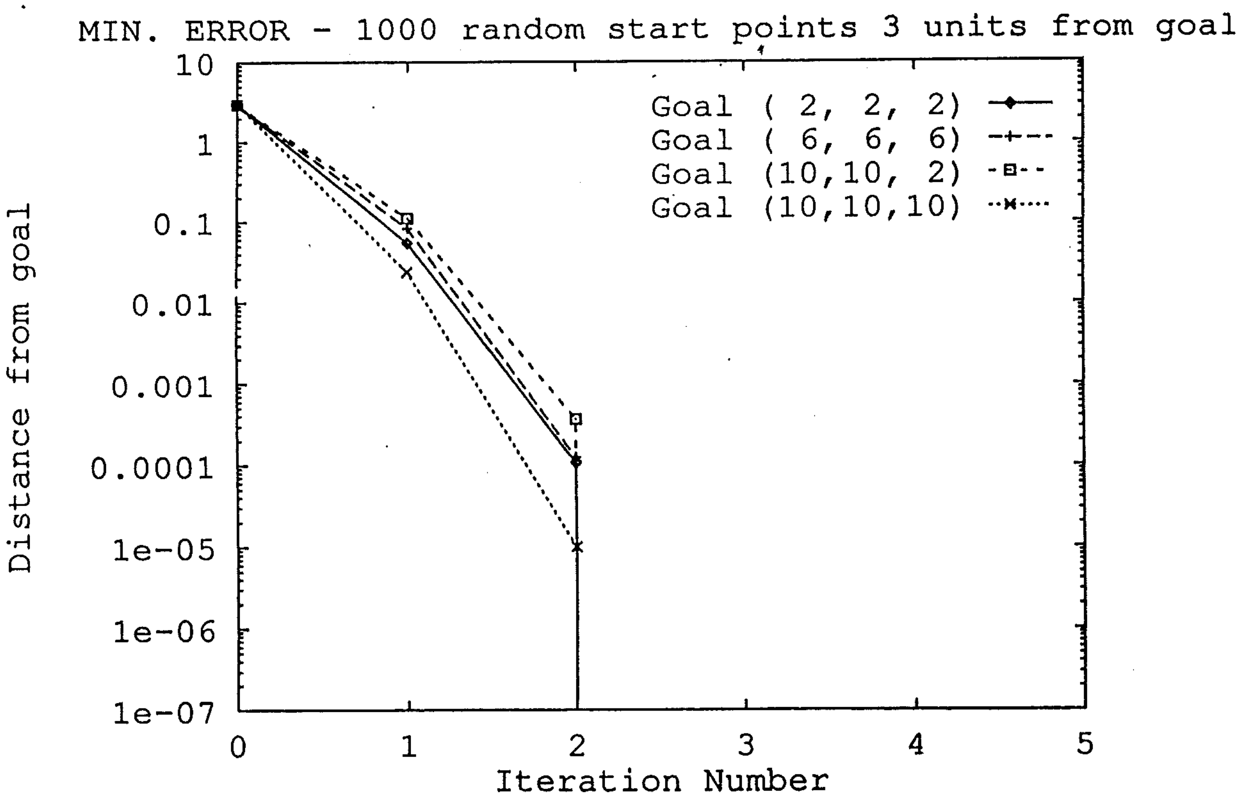
MIN. ERROR - 1000 random start points 3 units from goal
Goal (2, 2, 2)
Goal (6, 6, 6)
Goal (10,10, 2)
Goal (10,10,10)
Distance from goal
10
1
0.1
0.01
0.001
0.0001
1e-05
1e-06
1e-07
0
1
2
3
4
5
Iteration Number

Fig. 8

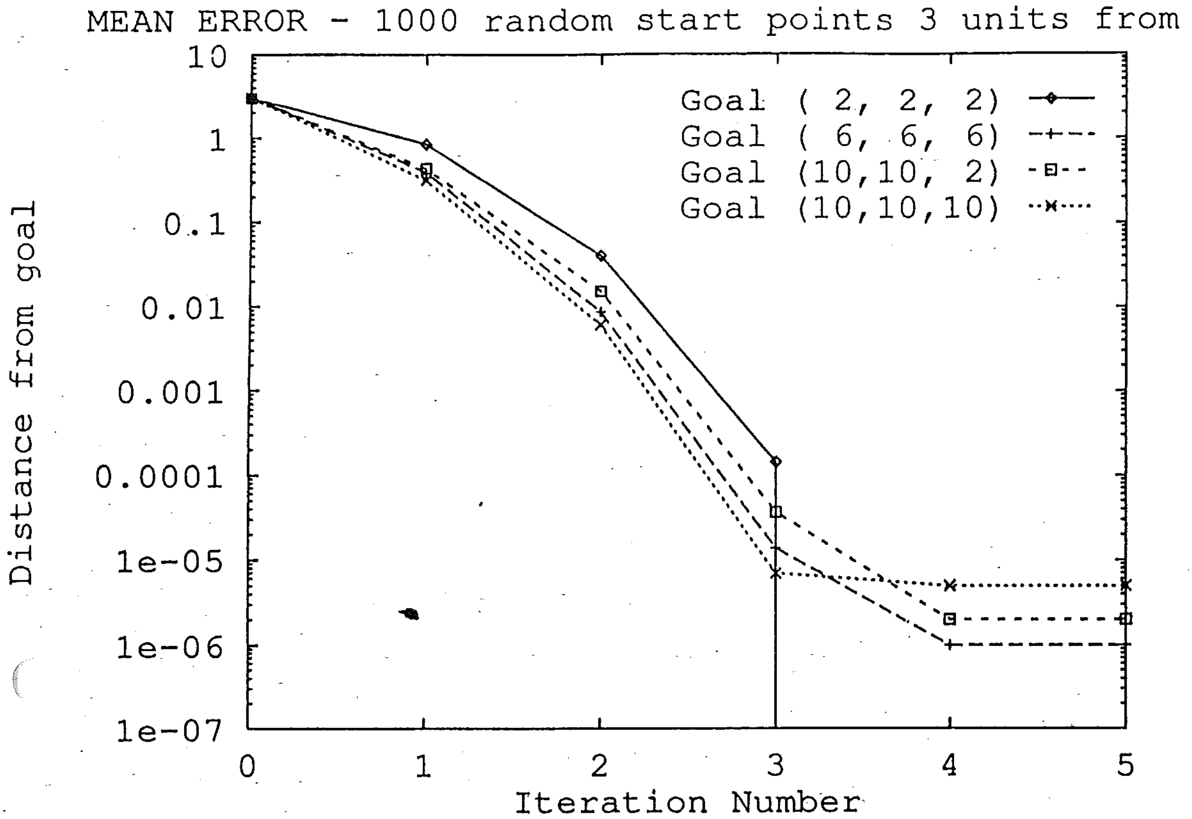
MEAN ERROR - 1000 random start points 3 units from goal
Goal (2, 2, 2)
Goal (6, 6, 6)
Goal (10,10, 2)
Goal (10,10,10)
Distance from goal
Iteration Number
10
1
0.1
0.01
0.001
0.0001
1e-05
1e-06
1e-07
0
1
2
3
4
5

Fig. 9

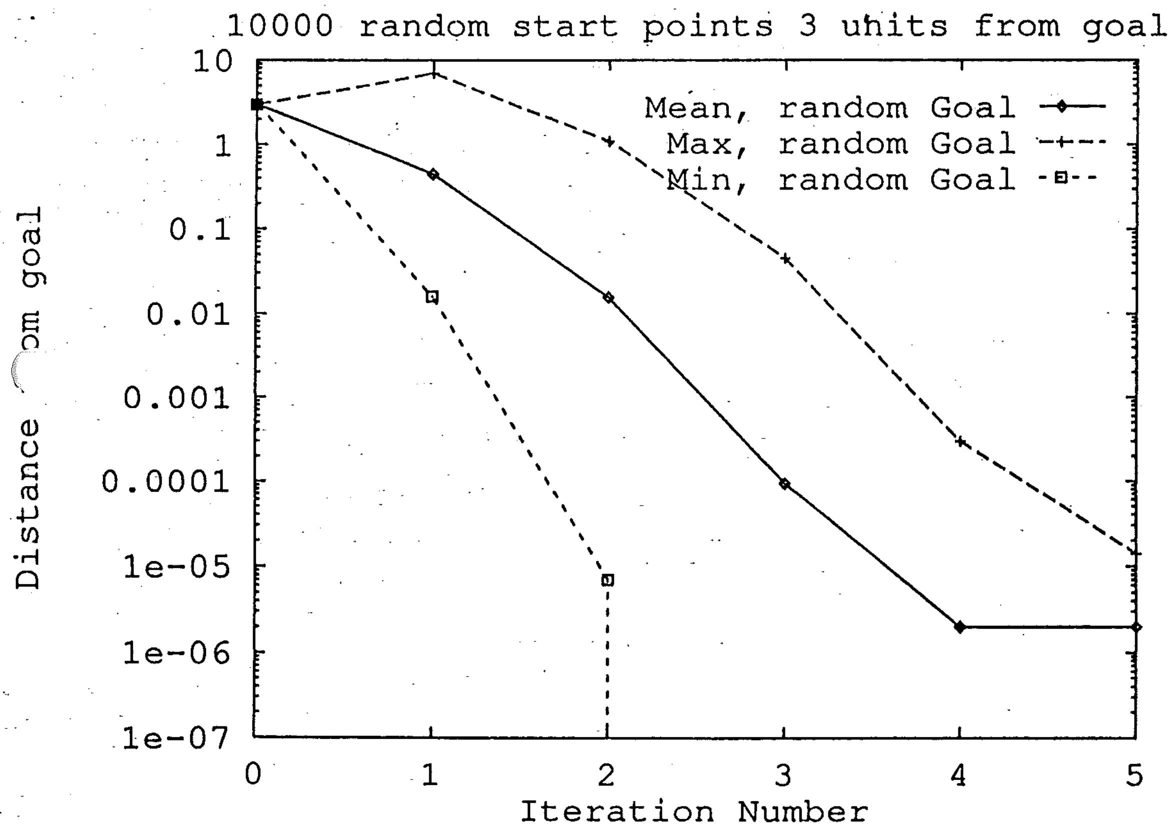
10000 random start points 3 units from goal
Mean, random Goal
Max, random Goal
Min, random Goal
Distance from goal
Iteration Number
10
1
0.1
0.01
0.001
0.0001
1e-05
1e-06
1e-07
0
1
2
3
4
5

Fig. 10

TABLE 1

Maximum Length Error Vectors:

Goal			Error Vector		
(0.20	0.20	0.20)	(−0.78	1.18	−0.78)
(2.00	2.00	2.00)	(0.98	−0.72	−0.72)
(0.10	2.00	2.00)	(1.50	−0.48	−0.48)
(0.10	0.10	5.00)	(1.38	1.38	−0.50)
(5.00	5.00	5.00)	(−0.87	−0.87	−0.87)
(1.00	10.00	10.00)	(2.45	−2.24	−2.24)
(10.00	10.00	10.00)	(−3.15	−3.15	−3.15)
(20.00	20.00	20.00)	(−15.29	−15.29	−15.29)
(30.00	30.00	30.00)	(−37.81	−37.81	−37.81)
(1.00	20.00	30.00)	(0.42	−19.93	−32.22)

6. Conclusions and Future Work

An iterative algorithm for a three dimensional localisation scheme has been presented. Convergence is obtained after only a few iterations. The algorithm also has the advantage that the final coordinate errors can be estimated in terms of the measurement errors. The system has been designed for autonomous vehicles and requires no umbilical cord nor communication systems apart from the ultrasonic transmissions.

The paper describes a four beacon system, however if more beacons are employed greater immunity to spurious pulses, obstacles and reflections, together with calibration for the speed of sound, could be obtained as was achieved in a two dimensional system [1,2]. The system is in the process of being implemented in air and future versions are anticipated for underwater. An underwater system is anticipated to operate faster than an air based system due to the higher speed of sound.

7. Acknowledgement

I wish to acknowledge the assistance of my colleague Dr. Greg Cambrell in the simplification of equations (3) and (8), and for his work in formulating alternative derivations (not included here) for the iterative technique presented in this paper.

REFERENCES

[1] L. Kleeman, "Ultrasonic autonomous robot localisation system", IEEE International Conference Intelligent Robots and Systems '89 Tsukuba, JAPAN, pp. 212–219 Sept. 89.

[2] L. Kleeman, "Ultrasonic autonomous robot localisation system," Monash University, Dept. Electrical and Computer Systems Engineering, Technical Report MECSE 89–8, 1989.

[3] T. Arai and E. Nakano, "Development of measuring equipment for location and Direction (MELODI) using ultrasonic waves," Trans. ASME Journal of Dynamic Systems, Measurement and Control, Vol.105, September 1983, pp. 152–156.

[4] W. E. Moritz, P. L. Shreve and L. E. Mace, "Analysis of an ultrasonic spatial locating system," IEEE Trans. on Instrumentation and Measurement, Vol. 25, No.1 March 1976, pp. 43–50.

[5] M. Dickinson and A. S. Morris, "Coordinate determination and performance analysis for robot manipulators and guided vehicles," IEE proceedings Pt. A, No.2, February 1988, pp. 95–98.

[6] R. L. Frank, "Current developments in Loran-C," Proceedings of the IEEE, Vol 71, No.10, October 1983, pp. 1127–1139.