Robotica (2014) volume 32, pp. 115–131. © Cambridge University Press 2013
doi:10.1017/S0263574713000726

Beacon selection and calibration for the efficient localization of a mobile robot

Jaehyun Park and Jangmyung Lee*

Department of Electrical Engineering, Pusan National University, Busan 609-735, South Korea

(Accepted June 17, 2013. First published online: August 2, 2013)

SUMMARY

This paper proposes a localization scheme using ultrasonic beacons in an unstructured multi-block workspace. Indoor localization schemes using ultrasonic sensors have widely been studied due to their low costs and high accuracies. However, ultrasonic sensors are susceptible to environmental noise due to the propagation characteristics of ultrasonic waves. In addition, the decay of ultrasonic signals over long distances implies that ultrasonic sensors are unsuitable for use in large indoor environments. To overcome these shortcomings of ultrasonic sensors, while retaining their advantages, a multi-block approach was devised by dividing an indoor space into several blocks with multiple beacons in each block. However, it is difficult to divide an indoor space into several blocks when beacons cannot be installed in a regular manner or when some new beacons are installed. To resolve this difficulty, a dynamic algorithm is needed to divide an indoor space into multiple blocks and to select suitable beacons. Therefore, this paper proposes a real-time localization scheme to estimate the position of a mobile robot independent of beacon locations and to estimate the position of a new beacon installed at an unknown position. A beacon selection algorithm was developed to select optimal beacons according to robot position and to set up sets of beacons for mobile robot navigation. By using the new beacon searching and calibration algorithm, a mobile robot is able to navigate in an unknown space without requiring the additional setup time needed to install new beacons. The performance of the proposed localization system was verified using real experiments.

KEYWORDS: Control of robotic systems; Mobile robots; Navigation; Robot localization; Sensor networks.

1. Introduction

With the development of robot-related technologies, such as microprocessors, sensors and computers, the practical uses of mobile robots have greatly increased. Real examples are provided by cleaning robots and tour guide robots operating in indoor environments, such as museums and exhibition halls. In outdoor environments, mobile robots have been developed for patrol and exploration. However, for service applications in different environments, position recognition, environmental perception, obstacle avoidance and path planning for autonomous navigation are essential. In particular, the precise localization of a mobile robot is one of the most important issues in the robotics field.

Localization schemes for mobile robots are classified into two categories, that is, relative and absolute position estimation. Relative position estimation is a dead reckoning method that utilizes odometry, gyroscopes and accelerometer sensors.[1,2] This method is straightforward and inexpensive to implement. However, accumulative errors become larger with as driving distance increases, and make precise localization more difficult.[3] Absolute position estimation obtained using global positioning system (GPS), lasers, vision, radio frequency identification (RFID) and ultrasonic sensors may resolve the problems of relative position estimation.[4–8] However, problems arising from the uncertainties associated with sensor information and the difficulties of installing sensors remain.

This paper uses ultrasonic sensors for the indoor localization of a mobile robot. Ultrasonic sensors are widely used, because they are inexpensive and easily controlled. However, ultrasonic sensors are

* Corresponding author. E-mail: jmlee@pusan.ac.kr; jangmlee@hanmail.net

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

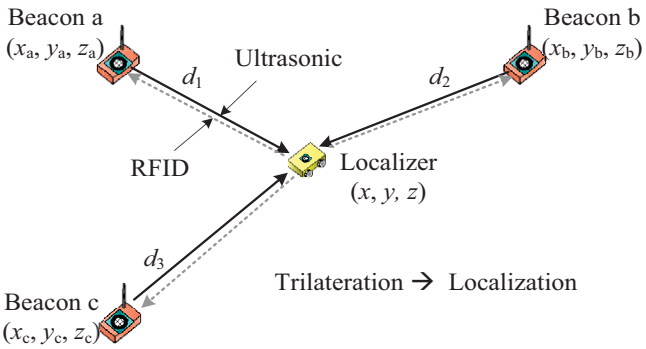


Fig. 1. (Colour online) Beacons in the iGS space.

susceptible to environmental noise resulting from the propagation characteristics of ultrasonic signals. Furthermore, because ultrasonic signals decay when they are transmitted over long distances,[9] they are unsuitable for use in large indoor environments. To overcome these shortcomings of ultrasonic sensors, while retaining their advantages, a new approach was proposed by dividing the indoor space into several blocks with multiple beacons in each block.[10] Ninety System Corp. developed the block ID recognition scheme to widen the operating area of mobile robots,[11] but the user needs to specify all coordinates of the active beacons and block IDs manually, which adds to the cost of setting up the working environment. Furthermore, when beacon placement is unstructured or additional beacons are installed, block repositioning and block division to specify beacon placement are difficult.

In this paper, a localization scheme is proposed for an unstructured beacon placement environment. Using this scheme, a mobile robot is able to estimate its position precisely using the beacon selection system independent of beacon placement situations. That is, the workspace of the mobile robot with the proper localization can be expanded by using a new beacon searching and calibration algorithm with the temporary beacon placed in an unknown position. Performance of the proposed algorithm has been verified through the comparisons with the conventional algorithms.

This paper is organized as follows: In Section 2, the indoor localization system (iGS) is introduced in detail, and in Section 3, the beacon selection algorithm in a multi-block environment is described. Section 4 describes the new beacon searching and calibration scheme, which provides the absolute position of a new beacon temporarily located at an unknown location. In Section 5, the effectivenesses and usefulnesses of the beacon selection and the new beacon search and calibration algorithms are verified by real experiments. Finally, Section 6 concludes this research work.

2. Multi-Block Localization Using iGS

2.1. Introduction to iGS (indoor GPS system)

For the indoor localization system, a mobile robot has been developed by the Intelligent Robot Laboratory of Pusan National University. The mobile robot is utilized as a platform for mobile robot navigation and its localization accuracy is $\pm$ 5 cm with the localization period of 200 ms. The basic operating space was defined as 5 m (wide) $\times$ 5 m (deep) $\times$ 2.5 m (high).

Figure 1 illustrates the navigational environment of the mobile robot in iGS space. There are at least three active beacon sensors in the room and one localizer on the mobile robot. The localizer calls a specific beacon to send ultrasonic signals, which are used by the localizer to measure distance to the beacon. When distance data from the three known beacon locations are available, the unknown location of the mobile robot can be estimated by trilateration.[12,13] The localizer uses an RFID signal to select a specific beacon. The selected beacon is requested to emit ultrasonic signals to provide the localizer with distance data. The distance from the localizer to the beacon can be easily obtained by multiplying the speed of the ultrasonic signal by its time of flight (TOF). Notice that effects of environmental temperature on the speed of the ultrasonic signal are compensated during the distance computation.

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

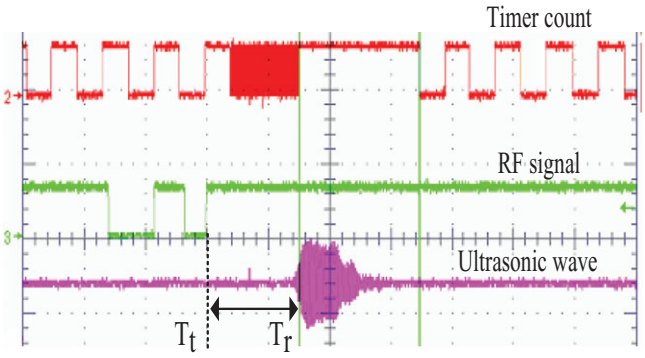


Fig. 2. (Colour online) TOF diagram.

2.2. *Distance measurements and the calculation of absolute location*

The distance between the beacon and the mobile robot can be calculated by multiplying the speed of the ultrasonic signal by the measured TOF.[14]

Figure 2 shows an illustration of the starting time for the counter, T_t, and the arrival time of the ultrasonic signal at the localizer, T_r. The counter starts when the RF signal is transmitted and stops when the ultrasonic signal is detected by the localizer with a certain threshold value. The travel time of the ultrasonic signal, *TOF*, is represented as

$$TOF = T_r - T_t. \tag{1}$$

The distance between the localizer and an active beacon, d, can be obtained as by multiplying v and *TOF*.

$$d = v \cdot TOF. \tag{2}$$

Furthermore, the speed of the ultrasonic signal, v, as a function of environmental temperature and given by

$$v \cong 331.5\sqrt{\frac{T}{237}}\ [\mathrm{m/s}], \tag{3}$$

where T is the absolute temperature.[15]

When the absolute positions of the beacons are pre-specified as (x_a, y_a, z_a), (x_b, y_b, z_b) and (x_c, y_c, z_c), and the distances between the localizer and beacons are d_1, d_2 and d_3, the absolute position of the mobile robot, $P(x, y, z)$, can be represented by the following equations:

$$\begin{bmatrix} (x-x_a)^2 + (y-y_a)^2 + (z-z_a)^2 \\ (x-x_b)^2 + (y-y_b)^2 + (z-z_b)^2 \\ (x-x_c)^2 + (y-y_c)^2 + (z-z_c)^2 \end{bmatrix} = \begin{bmatrix} d_a^2 \\ d_b^2 \\ d_c^2 \end{bmatrix}. \tag{4}$$

From these equations, the absolute position of the mobile robot, $P(x, y, z)$, can be computed as

$$P = A^{-1}B, \tag{5}$$

where

$$A = 2\begin{bmatrix} x_b - x_a & y_b - y_a & z_b - z_a \\ x_c - x_b & y_c - y_b & z_c - z_b \\ x_a - x_c & y_a - y_c & z_a - z_c \end{bmatrix}, \quad P = \begin{bmatrix} x \\ y \\ z \end{bmatrix},$$

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

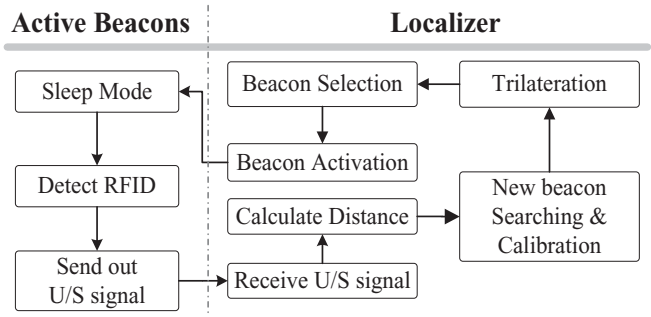


Fig. 3. The structure of the multi-block localization.

and

$$B = \begin{bmatrix} d_a^2 - d_b^2 - x_a^2 + x_b^2 - y_a^2 + y_b^2 - z_a^2 + z_b^2 \\ d_b^2 - d_c^2 - x_b^2 + x_c^2 - y_b^2 + y_c^2 - z_b^2 + z_c^2 \\ d_c^2 - d_a^2 - x_c^2 + x_a^2 - y_c^2 + y_a^2 - z_c^2 + z_a^2 \end{bmatrix}.$$

2.3. Multi-block localization system

Figure 3 shows the structure of the multi-block localization system proposed in this paper. A beacon selection algorithm is implemented to select optimal beacons, and the RFID transmitter in the localizer then sends trigger signals to activate the selected beacons. When an active beacon receives its ID, it sends an ultrasonic signal back to the localizer. The distance from the beacon to the mobile robot can be calculated by multiplying TOF by the speed of the ultrasonic signal, as shown in Eq. (2). If new beacons are installed, for example, because of a fault or an expansion of the workspace, the mobile robot can estimate the positions of these new beacons using the new beacon search and calibration algorithm.[18] A sleep mode is also used for active beacons, to save power when they are not in use. Initial positions of beacons are planned by the user and saved in the localizer's memory.

Figure 3 illustrates the structure of multi-block localization. The beacon selection process is described in Section 3 and the new beacon searching and calibration process in Section 4.

3. Beacon Selection Algorithm

Since the strength of an ultrasonic signal decays in inverse proportion to the square of the distance from a source,[16] one set of iGSs can only cover an area of about 5 m × 5 m (assuming that there are no obstacles in the room). When mobile robots must be localized globally, such obstacles, as well as decay problems, also need to be addressed. In these cases, the sensor space can be divided into multiple blocks to provide a solution for the global localization problem. In each block, a set of iGSs is installed and utilized for 'local' localization. In multiple blocks environments, it is important to choose the best beacons to optimize localization accuracy, because trilateration is affected by ranging errors, the geometrical arrangement of beacons and robot location. There are two possible beacon selection methods, one involves changing the block to update beacon information according to the robot's position, and the other method involves recognizing the boundary of the blocks to identify the block crossing situation.[17] However, either one of these two methods causes a problem whereby beacon information may not be updated because of a block recognition failure at a block boundary or because of malfunctioning beacons. Therefore, we propose an efficient algorithm for dealing with the problems of missing some beacons and losing a block boundary.

The beacon selection algorithm proposed in this paper is based on the distances between the mobile robot and beacons and the geometrical arrangement of beacons. In the case of iGS, which uses a threshold value to estimate distance, all beacons have almost the same detectable range. Thus, it is possible to select beacons according to distance from the mobile robot, when the robot's position, the

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

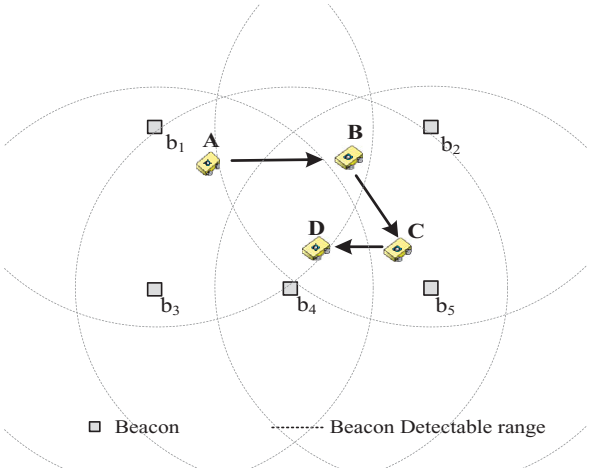


Fig. 4. (Colour online) An example of beacon selection.

positions of the beacons and their detectable ranges are known *a priori*. Figure 4 shows an example of beacon selection.

When a mobile robot is located at point A, beacons b_1, b_3 and b_4 are usable, whereas if the robot is located at point C, it employs b_2, b_4 and b_5 for localization. As in the above case, it is possible to select usable beacons according position from the robot. If the robot is located at point D, the beacons b_1, b_2, b_3, b_4 and b_5 can provide distance data to the mobile robot, but since three distance data are enough to obtain a location by trilateration, the mobile robot needs to select three among the five beacons.

To select the best three beacons, we propose a beacon selection algorithm that is derived using geometric dilution of precision (GDOP) analysis and distance sum (DS) analysis.

GDOP is commonly used in GPS, which describes the effect of geometry on the relationship between measurement and position determination error.[17] The performance of the trilateration algorithm is affected by range errors, by beacon geometrical arrangement, and by robot location. A formal derivation of the GDOP begins with the linearization of the range measurement equation. The basic measurement relationship of the triangulation method can be represented as

$$y = \mathbf{H}x + \upsilon_\rho, \quad y = \mathbf{H}x + \upsilon_\rho, \tag{6}$$

where the additive white noise with variance σ^2 is denoted as $\upsilon_\rho \in N(0, \sigma^2)$. Notice that y represents the difference between the actual measured and the predicted ranges, x is a 3×1 vector representing the deviation from the nominal position and $\mathbf{H}$ is the linear connection matrix obtained by linearizing the nominal user position. It is assumed that range measurement errors are not correlated with the variance σ^2.

The least squares solution can be expressed as

$$\hat{x} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}y, \tag{7}$$

and error covariance can be represented as

$$E\langle(\hat{\mathbf{x}})(\hat{\mathbf{x}})^T\rangle = \sigma^2(H^T H)^{-1}. \tag{8}$$

Due to the properties of a linear connection matrix, it is necessary to focus only on the diagonal elements of

$$(H^T H)^{-1} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}. \tag{9}$$

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

From the GDOP definition[17] and utilizing (9), we obtain

$$GDOP = \sqrt{A_{11} + A_{22} + A_{33}}, \tag{10}$$

which can be utilized to evaluate beacon reliability. Notice that the best reliable beacon has the lowest GDOP value.

DS is an important measure in defining the reliability of the beacon data since the ultrasonic signals are susceptible to the environmental noises. Also, notice that the ultrasonic signal strength decays in proportion to the inverse square of distance. Therefore, the error of measuring distance based on threshold value increases with distance. Based on this observation, the sum of three Euclidean distance values from three positions is adopted to identify nearest beacons.

In a three-dimensional space, beacon to robot distance is calculated as

$$d_i = \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2}, \tag{11}$$

where x_i, y_i and z_i are the positions of beacons b_i, and x_r, y_r and z_r represent the position of the robot. The DS value of the three distance values is obtained as

$$DS = \sum_{i=1}^{n} d_i. \tag{12}$$

A low DS indicates that the beacons are located near to the robots. Therefore, the DS value is useful for determining which set of beacons may provide the localization of the mobile robot with the smallest position error.

Singularity problem occurs in the area where collinear beacons are configured or the area where there exists a common covering area of beacons. Beacons are attached on the wall or ceiling in the indoor environments, since most indoor environments consist of corridors and ceiling. In the corridor environment, beacons cannot be configured as optimal due to the geometrical restrictions such as Fig. 13. These cases may cause singularity problems. The singularity problems can be avoided by the suitable selection of beacons using GDOP. When the localization error becomes large by the singularity, the localization can be replaced by the dead-reckoning sensors on the mobile robot.

The GDOP values and DS values in the beacon selection algorithm are calculated from the positions of the mobile robot and the usable set of beacons. The most reliable subset of beacons has a low GDOP value, and this subset is selected using the beacon selection algorithm. At point D in Fig. 4, the three subsets of beacons, that is, subset 1(b_1, b_2, b_4), subset 2(b_1, b_3, b_5) and subset 3(b_2, b_4, b_5) have low GDOP values, and in this case, sets 1 and 2 will have greater positional errors based on robot to beacon distance considerations.

Based upon these observations, we propose a beacon selection algorithm that uses the level of trust, which incorporates GDOP and DS values. The beacon-selecting algorithm uses the level of the trust, f_i, for each subset beacon useable in the robot coverage area. The value of trust indicator f_i, is defined as

$$f_i(B_list, p_r) = Range\ error \cdot GDOP_i + k \cdot DS_i, \tag{13}$$

where *B_list* are the useable beacons in the robot's coverage area, *Range error* represents the distance measuring error of the ultrasonic signal and p_r is the position of the robot. *GDOP* and *DS* represent the GDOP and DS values of the *i*th beacon subset, respectively. The k is a weighting, which is used to balance the level of importance desired for the DS term. The value of trust indicator f_i depends on the values of GDOP and DS in each beacon subset.

A low f_i means a good geometry and a small positional error. Therefore, subset 3 was selected at point D for the localization. Figure 5 shows the results for beacons selected using the proposed beacon selection algorithm when the robot was located at A to D. Figure 6 summarizes the developed beacon selection algorithm.

A coverage check is used to construct the beacon list, B_List which shows the beacons to be utilized by the mobile robot considering the robot position and beacon coverage. If there are more

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

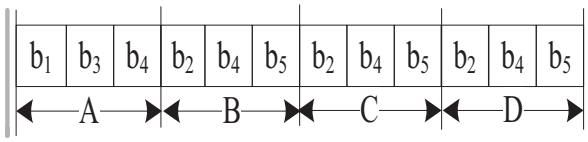


Fig. 5. Results of beacon selection algorithm.

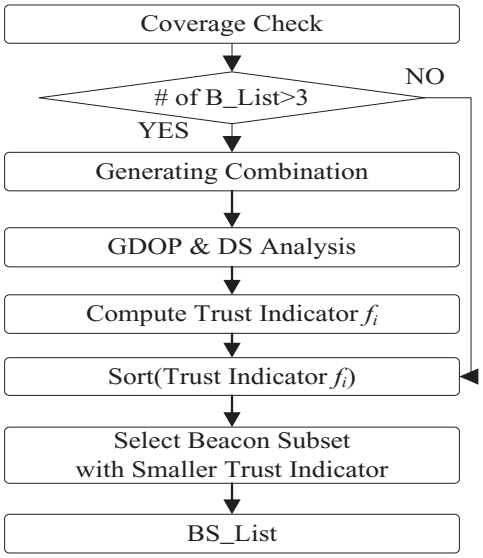


Fig. 6. Beacon selection algorithm.

than three beacons in the B_List, possible combinations of beacons are generated to select the best subset (with the smallest trust indicator value). The maximum number of B_List is seven, based on optimal placement of the beacons.[19,20] Normally, four and five beacons are available at a time for the mobile robot in this research.

To calculate the trust indicator f_i, GDOP analysis is performed to calculate GDOP values and DS analysis is used to calculate DS values for a given beacon subset. Then, the B_List is sorted in ascending order of trust indicator f_i and the beacon subset with the smaller trust indicator value is identified. Finally, the beacon selection list, BS_List, which contains the optimal geometric beacon locations, is returned for precise localization.

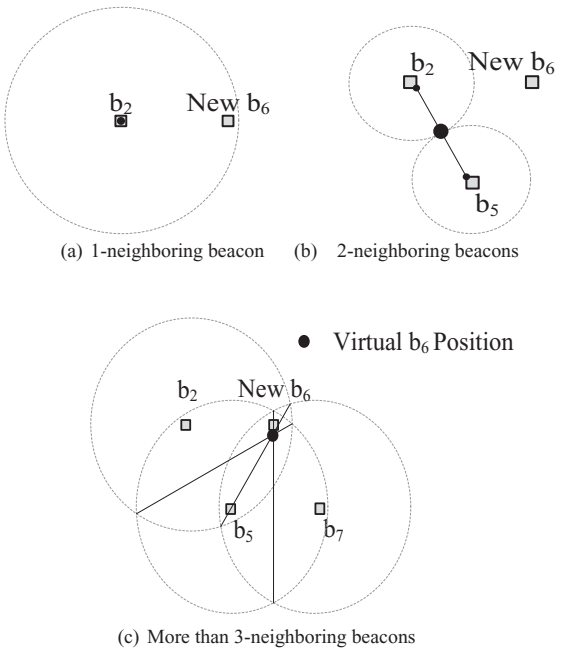
4. New Beacon Searching and Calibration

This section describes the new beacon search and calibration algorithm, which uses neighbouring beacons and dynamic auto-calibration to estimate the position of a new beacon in real time. When there are many corners and dynamics obstacles, it is not easy to estimate the precise locations of initial beacons. Therefore, the additional beacons can be required in the middle of navigation for the localization. If the navigation task is stopped to install the beacons, it is very time-consuming and is not cost-effective. The purpose of beacon searching and calibration algorithm is to estimate the beacon's position when a new beacon is installed in someplace temporarily without any *a priori* information on the beacon. It is useful to extend workspace or to cover some blocked area by using the additional beacons. It reduces the time to install new beacons and saves money.

4.1. Searching for new beacons

If there are new beacons in a multi-block workspace, the mobile robot may not recognize their positions, since it does not know when to initiate the auto-calibration algorithm. In order to solve this problem, we propose a new beacon searching algorithm which specifies the virtual position of a new beacon using neighbouring beacons.

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press



(a) 1-neighboring beacon (b) 2-neighboring beacons

(c) More than 3-neighboring beacons

Fig. 7. Examples of specifying the virtual position of a new beacon. (a) one neighbouring beacon. (b) two neighbouring beacons. (c) More than three neighbouring beacons.

If a new beacon is installed at an unknown position, the robot receives the IDs of the new beacon and neighbouring beacons from the user. In this situation, there are neighbouring beacons whose positions the mobile robot knows are shown in Fig. 7. The virtual position of the new beacon is obtained from the neighbouring beacons. If the mobile robot exists in the coverage area of the new beacon, auto-calibration is performed to estimate the position of the beacon and to determine if the new beacon can be used to receive ultrasonic signals. There are three methods of obtaining the virtual position of the new beacon. In the case where there is one neighbouring beacon, as shown in Fig. 7(a), the assumed position of the new beacon, b_6, is that of the neighbouring beacon, b_2. In the case of two neighbouring beacons, as shown in Fig. 7(b), the point of intersection of the two circles around the two neighbouring beacons of radius equal to 1/2 the distance between them is assumed to be the position of b_6. When there are more than three beacons, the position of a virtual beacon can be estimated as b_6 based on the least square method using all of the neighbouring beacons within the detectable range, as shown in Fig. 7(c).

When there are a number of new beacons, the robot calculates the position of each 'old' beacon, and then the positions of new beacons are estimated using the new beacon calibration procedure by assuming that the distance from the robot to the new beacon is the closest and that the mobile robot is in coverage area of the new beacon.

4.2. New beacon calibration

The new beacon calibration procedure is a method of estimating the position of a beacon based on auto-calibration. The principle of auto-calibration[18] is the inverse function of localization for current active beacons. The mobile robot measures distances from three known positions (referred to as seeds 1, 2 and 3) to the new beacon. The coordinates of the new beacon can then be calculated using these three distances.

The coordinates of the new beacon b_6, as shown in Fig. 8, in the sensor space, can be derived as follows:

$$P_6 = A^{-1}B, \tag{14}$$

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

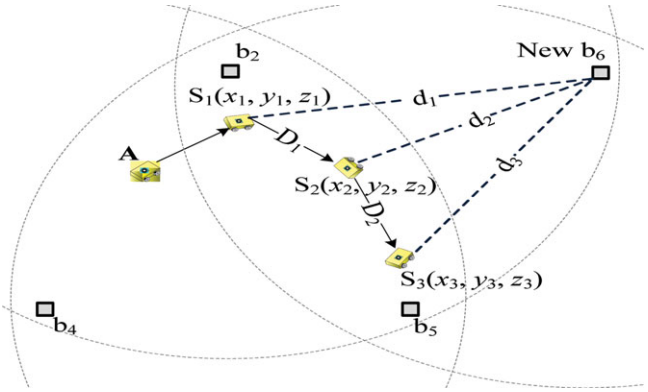


Fig. 8. (Colour online) Auto-calibration in an unstructured environment.

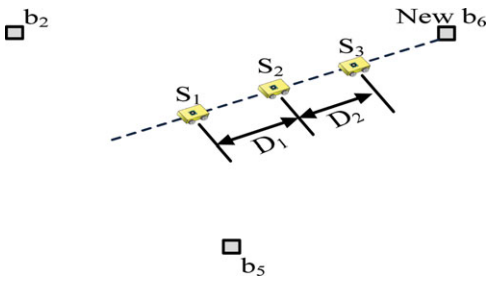


Fig. 9. (Colour online) Singularity problem caused by seed alignment.

```
Function Seed Allocation(new beacon)
 1:  seed number = 1;
 2:  while seed number != 3 do
 3:      seeds[seed number] = position calculation(BS_list);
 4:      if calculation distance(new beacon) == failure then
 5:          return failure;
 6:      end if
 7:      if seed number >= 2 then
 8:          Dn = distance calculation(seeds[seed number – 1], seeds[seed number]);
 9:          if Dn > distance threshold then
10:              seed number++;
11:          end if
12:          if seed number > 2 and singularity check(seeds) == failure then
13:              return failure;
14:          end if
15:      else
16:          seed number++;
17:      end if
18:  end while
19:  return success
```

Fig. 10. The seed allocation algorithm.

where

$$A = 2\begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_2 & y_3 - y_2 & z_3 - z_2 \\ x_1 - x_3 & y_1 - y_3 & z_1 - z_3 \end{bmatrix}, \quad P_6 = \begin{bmatrix} x_6 \\ y_6 \\ z_6 \end{bmatrix},$$

and

$$B = \begin{bmatrix} d_1^2 - d_2^2 - x_1^2 + x_2^2 - y_1^2 + y_2^2 - z_1^2 + z_2^2 \\ d_2^2 - d_3^2 - x_2^2 + x_3^2 - y_2^2 + y_3^2 - z_2^2 + z_3^2 \\ d_3^2 - d_1^2 - x_3^2 + x_1^2 - y_3^2 + y_1^2 - z_3^2 + z_1^2 \end{bmatrix}.$$

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

```
Function auto-calibration(seeds, new beacon)
 1:  while auto-calibration flag != success do
 2:      if(Seed Allocation(new beacon) == failure) then
 3:          return failure;
 4:      end if
 5:      P_new beacon = trilerteration(seeds, new beacon);
 6:      GDOP Analysis(seeds, P_new beacon);
 7:      if GDOP value < 10 then
 8:              auto-calibration flag = success;
 9:      else
10:          auto-calibration flag = failure;
11:     end if
12:  end while
13:  return true
```

Fig. 11. The auto-calibration algorithm.

```
1:  if new beacon confirm() == true then
2:      P_b virtual list = calculation virtual beacon position(new beacon list);
3:      while # of new beacon != 0 do
4:          # of close beacon, closest beacon = detect close beacon (coverage
check(P_b_virtual list));
5:          if detect beacon(closest beacon) == true then
6:              if auto-calibration(seeds, closest beacon) == success then
7:                  delete beacon(new beacon list, closest beacon);
8:                  add beacon(B_List, closest beacon);
9:                  # of new beacon - -;
10:         end if
11:     end if
12:   end while
13:  end if
```

Fig. 12. New beacon search and calibration algorithm.

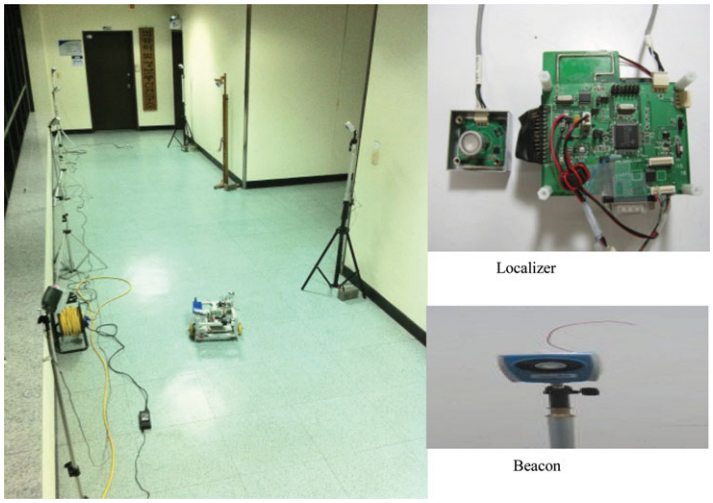


Fig. 13. (Colour online) The experimental environment.

Notice that d_1 is the distance from the mobile robot at Seed 1 to new beacon b_6, d_2 is the distance from the mobile robot at Seed 2 to new beacon b_6 and d_3 is the distance from the mobile robot at Seed 3 to new beacon b_6. From Eq. (14), the coordinates of new beacon b_6 can be obtained.

When the robot is moving into an unstructured environment, it calibrates the locations of a new beacon using the auto-calibration procedure. The conventional method[8,11] of selecting seeds selects those for which the mobile robot's new positions are greater than a given distance threshold, D_n, in Eq. (7), in order to avoid the selection of seeds that are too close. In addition, the singularity problem in practical situations, that is, when seeds are in a straight line relative to the new beacon b_6, as shown

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

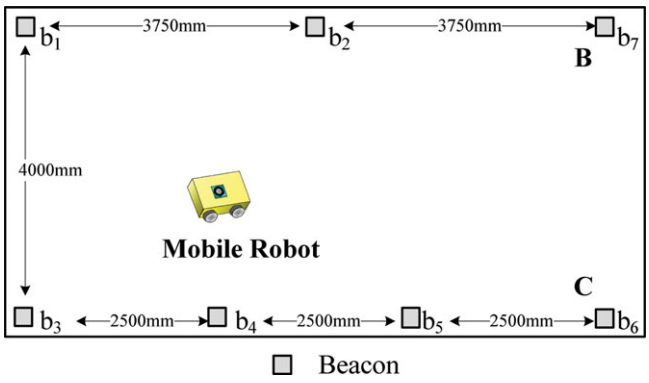


Fig. 14. (Colour online) Beacon deployment.

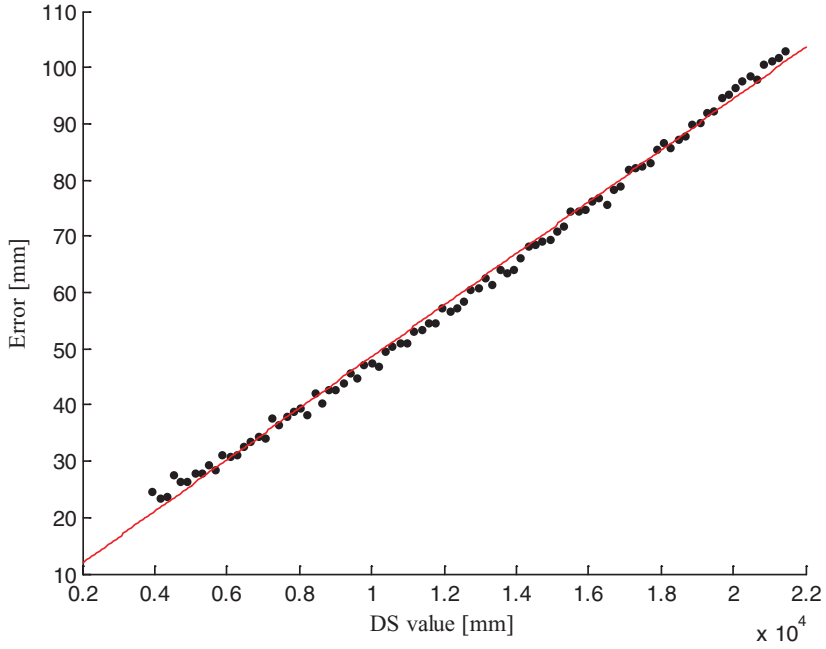


Fig. 15. (Colour online) The relationship between positional error and DS value.

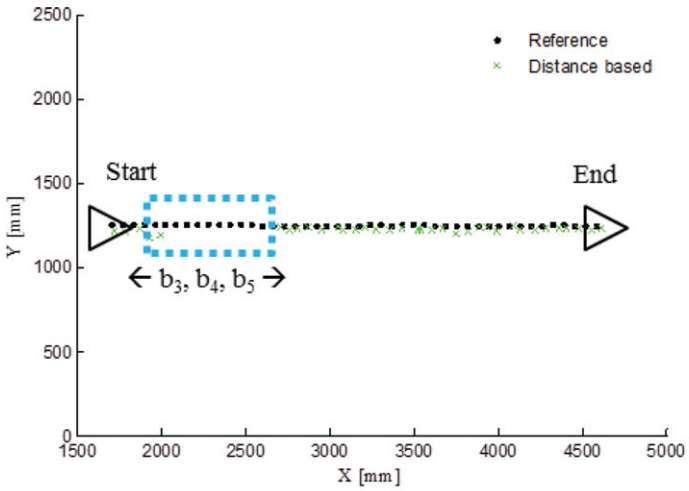
in Fig. 9, is solved by ensuring that the minimum singular value of the matrix is non-zero.

$$D_n = \sqrt{\{x_{n+1} - x_n\}^2 + \{y_{n+1} - y_n\}^2}\,, \tag{15}$$

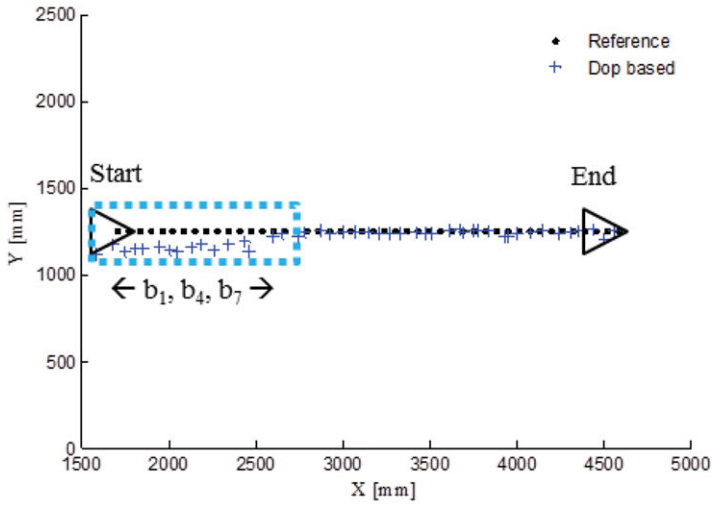
where $n = 1$ and $n = 2$. The seed allocation algorithm is described in Fig. 10 in detail.

However, the auto-calibration algorithm is affected by geometrical arrangement of seeds and the location of the new beacon, since geometry causes the positional errors. Therefore, the new beacon calibration algorithm is used to select optimal seeds using D_n and the GDOP value, in order to solve the seed arrangement problem. When the robot recognizes a new beacon using the new beacon searching algorithm, it choose those seeds satisfying D_n, and then, if the GDOP value of the seeds is less than 10, which indicates that seeds can be used for calculations, the coordinates of the new beacon are obtained by auto-calibration. Therefore, the mobile robot is able to navigate in an unstructured environment using the new beacon. The auto-calibration algorithm, which involves GDOP analysis, is introduced in Fig. 11. If the seed allocation algorithm finishes successfully, the position of the new beacon will be calculated using seed positions. Then, if the GDOP value from GDOP analysis is less than 10, the position of the new beacon is stored.

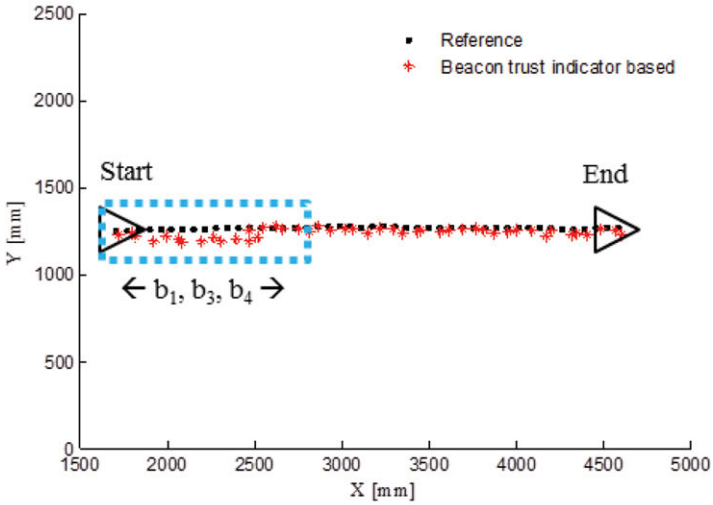
https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press



(a) Distance based trajectory



(b) DOP based trajectory



(c) Beacon trust indicator based trajectory

Fig. 16. (Colour online) The straight trajectory comparison. (a) Distance-based trajectory. (b) DOP-based trajectory. (c) Beacon-trust-indicator-based trajectory.

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

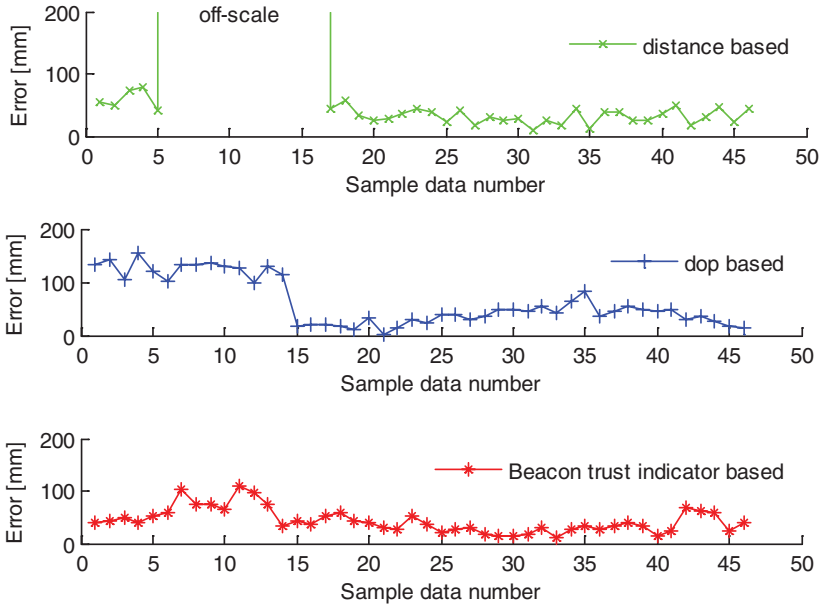


Fig. 17. (Colour online) Positional errors along a straight trajectory.

Figure 12 summarizes the new beacon search and calibration algorithm developed in the multi-block workspace. If information of the new beacon is transmitted to the mobile robot by the user, calculating the virtual position of the beacon is initiated, and subsequently, the auto-calibration algorithm is executed using the closest beacon. Finally, the closest beacon is deleted from the new beacon list and added to the B_list used by the beacon-scheduling algorithm.

5. Experiments

The mobile robot used in this paper was developed with DC motors with embedded encoders and a dsPIC30F4012 from Microchip controller. The experimental iGS is composed of the localizer on the mobile robot and beacons on walls or ceilings, as shown in Fig. 13. The localizer system uses a TMS320C2406A from TI for system control, a rfPIC12F675F from Microchip for RF signal control and a AT40-10P (40 kHz) from Nippon Ceramic for the ultrasonic signal receiver. The beacon system uses a MSP403F1101A from TI for system control and an AR40-10P (40 kHz) for the transmission of ultrasonic signals. Estimating the distance from the beacon to the receiver takes 40 ms per beacon. The localization time is 200 ms when using four beacons and the average error is about 50 mm. Experiments were performed in an indoor environment, in which eight beacons were deployed at the vertices (x, y) (units: mm) of the workspace, to prove the effectivenesses of the algorithms proposed in this paper. Further information is represented author's personal webpage (https://sites.google.com/site/mobilerobotlocalization/home).

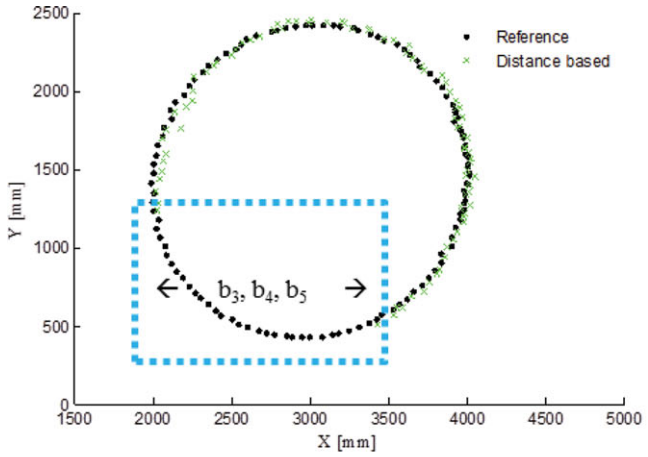
5.1. Testing of the beacon selection algorithm

Figure 14 depicts the deployments of beacons and the mobile robot. Seven beacons were deployed in the workspace. Before the experiment began, *Range error* and k in Eq. (13) were determined. *Range error* was set at 22.5 mm, which was the average value for the 7 beacons, and then k was calculated. *Range error* · *GDOP* means positional error, and therefore, k is obtained by the relationship between the positional error and the DS value. Figure 15 shows the relationship between positional error and DS value. The black dot in Fig. 15 indicates positional error versus DS value, and the red line shows the best linear fit, k calculated using

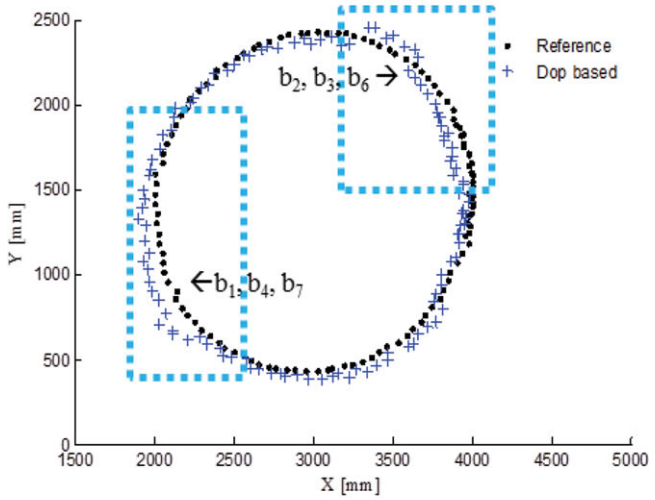
$$k = p_1 + \frac{p_2}{DS}, \tag{16}$$

where $p_1 = 0.0045$ and $p_2 = 2.6636$ by the result in Fig. 15.

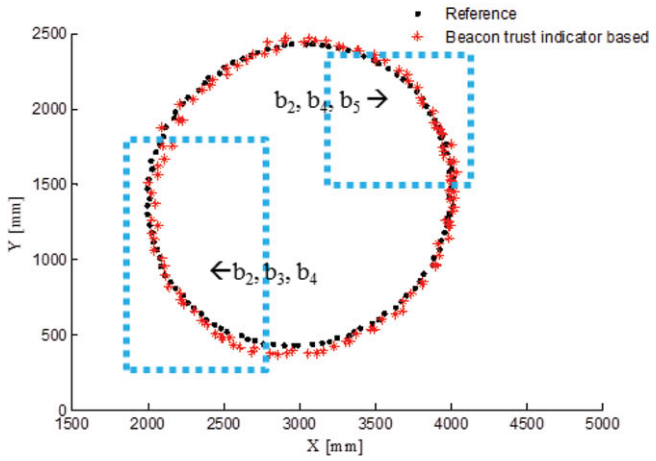
https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press



(a) Distance based trajectory



(b) GDOP based trajectory



(c) Beacon trust indicator based trajectory

Fig. 18. (Colour online) The circular trajectory comparison. (a) Distance-based trajectory. (b) GDOP-based trajectory. (c) Beacon-trust-indicator-based trajectory.

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

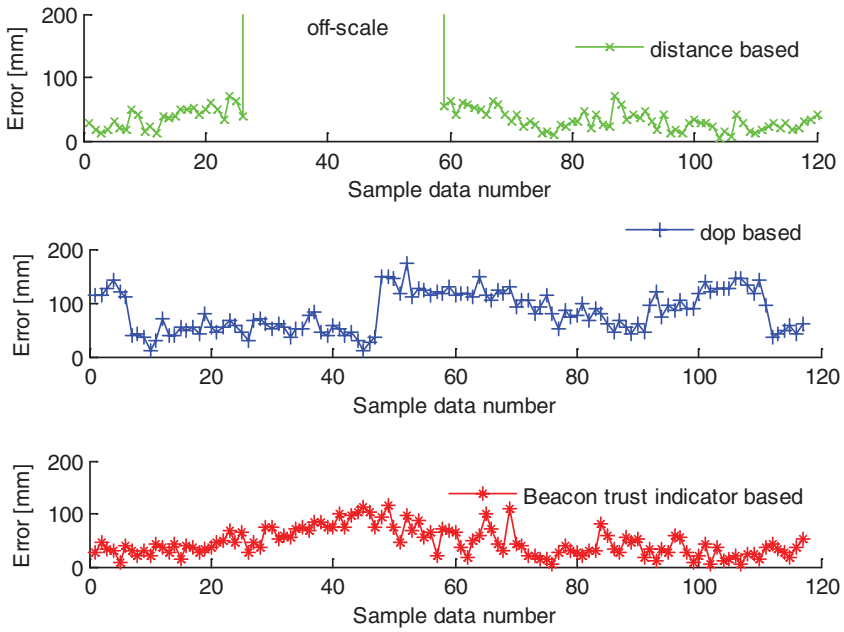


Fig. 19. (Colour online) Position errors along a circular trajectory.

Two different experiments were conducted to investigate the performance of the multi-block application and to compare the trajectories of the mobile robot. To compare the performance of position estimation, distance and GDOP-based beacon selection algorithms are used. The distance-based beacon selection algorithm estimates the robot position using the three nearest beacons, whereas the GDOP-based beacon selection algorithm estimates the robot position using the three beacons with the lowest GDOP values. In order to compare the effectiveness of the three-beacon selection algorithm, straight and circular trajectories were used.

Figures 16(a)–(c) show the trajectory obtained using the three-beacon selection algorithm. In Fig. 16(a), trajectory loss occurred near the start point, since beacons b_3, b_4 and b_5 were used. These beacons cause a singularity problem because they are arranged along with X-axis. The GDOP-based beacon selection algorithm in Fig. 16(b) showed a large error at the start point, where the b_1, b_4 and b_7 beacons had the lowest GDOP values, on the other hand, the b_7 beacon had large distance error since it was located further away than the other beacons. On the other hand, the beacon-trust-indicator-based selection algorithm produced positioning results with a lower position than the other algorithms because it considers both distance and GDOP.

The position errors of the algorithms are summarized in Fig. 17. The distance-based algorithm produces a large error shown as 'off-scale' due to singularity problem when the beacons are arranged side by side. The average error of the GDOP algorithm was 69.39 mm and of the beacon trust indicator algorithm was 42.95 mm.

Figures 18(a)–(c) illustrate the results obtained for a circular trajectory of the robot, which showed similarities with straight trajectory results. More specifically, the distance-based selection algorithm encountered a singularity problem and the GDOP-based selection algorithm had a large error due to the use of distanced beacon. However, the beacon-trust-indicator-based selection algorithm produced accurate position estimations. Figure 19 shows position errors along a circular trajectory. The average error of the GDOP-based algorithm was 84.75 mm, whereas that of the beacon-trust-indicator-based algorithm was 46.33 mm, indicating that the robot well determined its position in a wide multi-block environment using the beacon-trust-indicator-based selection algorithm.

5.2. Experiment in a new beacon environment

Experiment 3 was performed to compare the conventional auto-calibration algorithm and the new beacon searching and calibration algorithm. A new beacon was installed at an unknown position

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

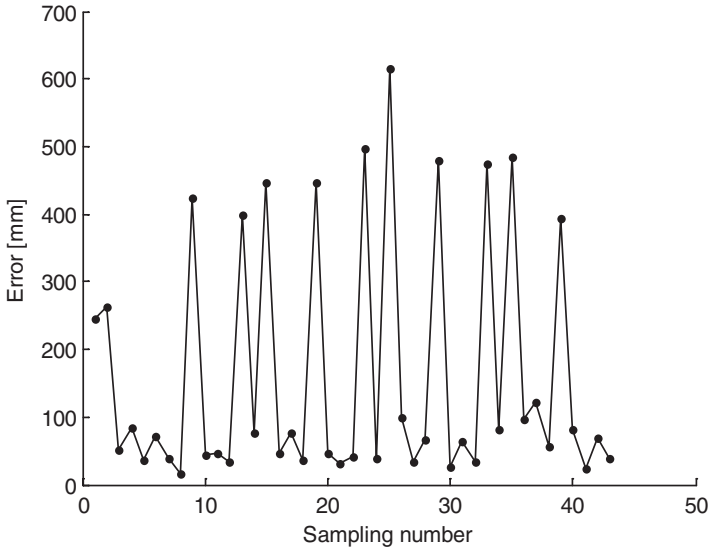


Fig. 20. Coordinate estimation errors of the new beacon using conventional auto-calibration.

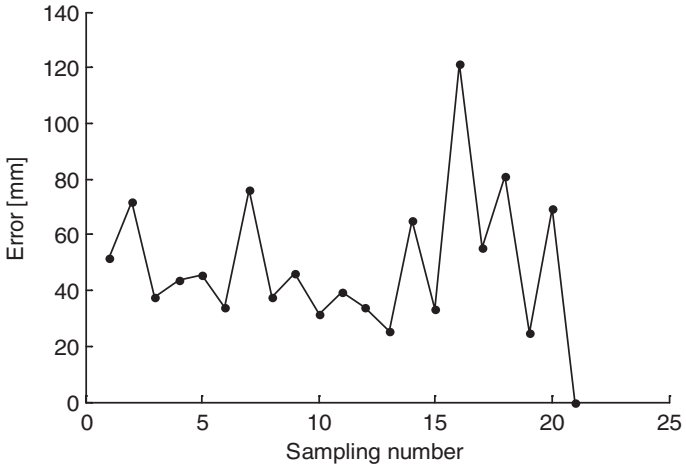


Fig. 21. Coordinate estimation errors of the new beacon using the new beacon searching and calibration algorithm.

(Fig. 14). The real coordinates of the new beacon were (1,827 mm, 13 mm), which were obtained using the auto-calibration procedure with the robot stationery. Figures 20 and 21 show the coordinate estimation errors of the new beacon for the conventional auto-calibration algorithm and the new beacon searching and calibration algorithm. In the case of the conventional auto-calibration algorithm, large positional errors occurred due to the arrangement of seeds. However, when the optimal set of beacons was selected and the calibration algorithm was used, the estimation accuracy is improved. The average error in Fig. 20 was 159.95 mm and in Fig. 21 was 60.49 mm. Throughout this experiment, the new beacon searching and calibration algorithm was found to more effectively calibrate the position of new beacons than the conventional calibration method.

6. Conclusions

Beacon selection and calibration algorithms have been developed to localize a mobile robot efficiently in a large workspace. Ultrasonic-sensor-based localization systems have been widely used for this purpose, because they are inexpensive, accurate and easy to control. However, because ultrasonic signals decay rapidly with distance from the source, they are not used for localization in large indoor environments. In this paper, to overcome these shortcomings of ultrasonic sensors, a new localization

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press

algorithm was developed to cope with multiple beacons in multiple blocks in wide workspaces. A beacon selection algorithm, which uses the trust indicator (derived by GDOP and DS analysis) of a beacon subset, was developed for the real-time localization of the mobile robot. In addition, by using the new beacon searching and calibration algorithm, the mobile robot was able to navigate in unknown space after installing an additional beacon. The effectiveness of this localization system was verified but performing real robot navigation experiments. The beacon selection and calibration-algorithm-based localization system can be applied to location-based services in various fields, and to robot positioning systems.

Acknowledgements

This research was supported by the Special Environment Navigation/Localization Robot Technology Research Center support program of the Korean Ministry of Knowledge Economy supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2011-(C7000-1001-0004)).

References

1. H. Myung, H. K. Lee, K. Choi and S. W. Bang, "Mobile robot localization with gyroscope and constrained Kalman filter," *Int. J. Control Autom. Syst.* **8**(3), 667–676 (2010).
2. J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Trans. Robot. Autom.* **12**, 869–880 (1996).
3. C. C. Tsai, "A localization system of a mobile robot by fusing dead-reckoning and ultrasonic measurements," *IEEE Trans. Ind. Electron.* **47**(5), 1399–1404 (1998).
4. T. S. Abuhashim, M. F. Adbedl-Hagez and M. A. Al-Jarrah, "Building a robust integrity monitoring algorithm for a low cost GPS-aided-INS system," *Int. J. Control Autom. Syst.* **8**(5), 1108–1122 (2010).
5. D. Amarasinghe, G. K. I. Mann and R. G. Gosine, "Landmark detection and localization for mobile robot applications; a multisensor approach," *Robotica* **28**(5), 663–673 (2010).
6. N. M. Kwok, Q. P. Ha, S. Huang, G. Dissanayake and G. Fang, "Mobile robot localization and mapping using a gaussian sum filter," *Int. J. Control Autom. Syst.* **5**(3), 251–268 (2007).
7. I. Parra, M. A. Sotelo, D. F. Llorca and M. Ocaña, "Robust visual odometry for vehicle localization in urban environments," *Robotica* **28**, 441–452 (2010).
8. S. S. Han, H. S. Lim and J. M. Lee "An efficient localization scheme for a differential-driving mobile robot based on RFID system," *IEEE Trans. Ind. Electron.* **54**(6), 1–8 (2007).
9. C. C. Tsai, "A localization system of a mobile robot by fusing dead-reckoning and ultrasonic," *IEEE Trans. Instrum. Meas.* **47**, 1399–1404 (1998).
10. W. Qinhe and H. Hashimoto, "Fast Localization of Multi-Targets in the Intelligent Space," *Proceedings of the Annual Conference SICE*, Tagawa, Japan (2007) pp. 264–269.
11. D. G. Seo and J. M. Lee, "Localization Algorithm for a Mobile Robot Using iGS," *Proceedings of the 17th International Federation of Automatic Control World Congress*, Seoul, Korea (2008) pp. 742–747.
12. D. E. Manolakis, "Efficient solution and performance analysis of 3D position estimation by trilateration," *IEEE Trans. Aerosp. Electron. Syst.* **32**, 1239–1248 (1996).
13. F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Trans. Robot.* **21**, 93–101 (2005).
14. B. Barshan, "Fast processing techniques for accurate ultrasonic range measurements," *IOP J. Meas. Sci. Technol.* **11**, 45–50 (2000).
15. W. S. Eom and J. M. Lee, "Ubiquitous Positioning Network of a Mobile Robot with Active Beacon Sensors," *Proceedings of the International Conference on Circuits/System, Computers and Communications*, Busan, Korea (2007) pp. 255–256.
16. S. Y. Yi and B. W. Choi, "Autonomous navigation of indoor mobile robots using a global ultrasonic system," *Robotica* **22**, 369–374 (2004).
17. D. E. Manolakis, "Efficient solution and performance analysis of 3-D position estimation by trilateration," *IEEE Trans. Aerosp. Electron. Syst.* **32**, 1239–1248 (1996).
18. S. B. Kim, J. M. Lee and I. O. Lee, "Precise Indoor Localization System For a Mobile Robot Using Auto Calibration Algorithm," *Proceedings of the 13th International Conference on Advanced Robotics*, Jeju, Korea (2007) pp. 635–640.
19. T. Sun, L. Chen, C. Han and M. Gerla, "Reliable Sensor Networks for Planet Exploration," *Proceedings of IEEE International Conference Networking, Sensing and Control*, Tuscon AZ, USA (2005) pp. 816–821.
20. M. Y. Nam, M. A. Sabbagh, J. E. Kim, M. K. Yoon, C. G. Lee and E. Y. Ha, "A real-time ubiquitous system for assisted living: Combined scheduling of sensing and communication for real-time tracking," *IEEE Trans. Comput.* **57**(6), 795–808 (2008).

https://doi.org/10.1017/S0263574713000726 Published online by Cambridge University Press