Source Localisation with a Bernoulli Particle-Filter-Based Bearings-Only Tracking Algorithm

Jennifer L. Palmer
Ricardo Cannizzaro
Branko Ristic
Thomas Cheah
Chatura Nagahawatte
John Gilbert
Sanjeev Arulampalam
Defence Science & Technology Group
506 Lorimer St, Fishermans Bend, VIC 3207 Australia
Jennifer.Palmer@dsto.defence.gov.au; Ricardo.Cannizzaro@dsto.defence.gov.au;
Branko.Ristic@dsto.defence.gov.au; Thomas.Cheah@dsto.defence.gov.au;
Chatura.Nagahawatte@dsto.defence.gov.au; John.Gilbert@dsto.defence.gov.au;
Sanjeev.Arulampalam@dsto.defence.gov.au

Abstract

Locating the source of a chemical or acoustic emitter in an unknown environment with noise-corrupted detector signals is a challenging task undertaken regularly in emergency-response operations. Such missions can endanger personnel, thus the use of autonomous systems would be highly advantageous. In this research, an approach for autonomous bearings-only tracking (BOT) is trialled in a proof-of-concept experiment relying on a bright light source and a sensor constructed from photodiodes and carried atop a radio-controlled ground vehicle. The goal is to test the algorithm's performance with real-world, noise-corrupted data. A Bernoulli particle-filter algorithm, designed to tolerate cluttered signals, is used to estimate the location of the light source from low-resolution measurements of its bearing. Data is collected with a variety of navigational patterns in an environment that occasionally produces false detections. The algorithm yields accurate estimates of the location of the light source (on average, 22 cm from a source with a 16-cm horizontal aperture); however, in some cases, false detections result in poor estimation of the source position and large uncertainties. Future research will aim to address the limitations of the current trial and to implement the algorithm on an autonomous platform.

1 Introduction

Unmanned systems have proven useful in a variety of military and civilian applications, including emergency response, humanitarian assistance, and disaster relief [Anon., 2012]; and systems with advanced autonomous capabilities could significantly reduce the risks to personnel in unknown or un-cleared areas for intelligence, surveillance, and reconnaissance missions, as well as more specialised operations, such as contaminant-source localisation or search and rescue/recovery. Autonomous systems may eventually supplement or replace highly trained personnel in the most dangerous parts of these safety-critical, labour-intensive tasks.

For use in missions to detect and locate an emitter of, for example, a chemical, biological, or radiative (CBR) contaminant, autonomous systems must be capable of performing searches without continual human intervention. The conditions in which emergency-response missions are performed are varied and often present significant navigation and path-planning challenges for autonomous systems. For example, in cluttered urban terrain, including inside buildings, data from global navigation satellite systems and line-of-sight communications signals are unreliable (or unobtainable).

This paper describes the demonstration of a bearings-only-tracking (BOT) algorithm that may eventually be executed on mobile air and ground platforms during counter-CBR missions, or on autonomous undersea platforms searching for a downed aircraft by sensing its acoustic beacon. BOT strategies rely on noise-corrupted bearing measurements to estimate the kinematics (position and velocity) of a target/source with a purely radial dispersion (e.g., a radiative source or a chemical diffusing in a region without wind). Common defence applications employ a variety of active and passive sensing techniques and include submarine tracking with passive sonar and radar tracking of aircraft [Arulampalam*, et al.*, 2007]. The success of BOT algorithms relying on passive sensors depends on the sensor's motion having a non-zero derivative at least one order higher than that of the target [Yan *et al.*, 2014]. For example, to track a target with a constant, non-zero velocity, a seeker needs to change its course relative to the target or to accelerate along a straight-line course [Arulampalam*, et al.*, 2007].

The BOT algorithm tested here utilises a particle filter (PF) with a Bernoulli process that increases its robustness to false detections and intermittent target/source loss. With any PF, the required posterior density of the state vector (the location of the source in the present case) is represented by a set of random samples ('particles') with associated weights [Arulampalam*, et*

al., 2007]. The weighted average of the samples provides an estimate of the state vector. With every new measurement, the weights are updated according to the likelihood of the corresponding particle state. With a Bernoulli PF, the updating of each particle involves a likelihood-ratio calculation for all possible source-originated signals [Ristic *et al.*, 2013]; thus, the level of complexity (and processing time) is higher than that of a traditional PF-based algorithm.

Techniques relying on PFs have shown promise in the solution of challenging nonlinear signal-processing problems, and their recursive structure provides opportunities for real-time utilisation [Arulampalam and Ristic, 2000; Arulampalam *et al.*, 2002; Gordon *et al.*, 1993; Ristic *et al.*, 2004]. PF-based techniques exhibit superior performance in comparison with Kalman filters, though at the cost of higher computational time [Aidala, 1979; Aidala and Nardone, 1982; Aidala and Hammel, 1983; Arulampalam and Ristic, 2000; Gordon, *et al.*, 1993; Nardone and Graham, 1997; Peach, 1995].

2 Background and Overview

The Defence Science and Technology (DST) Group is developing search and tracking algorithms that can enable autonomous localisation of CBR sources and be used in myriad other applications. The research described here is part of a broader program to develop autonomous-control capabilities for robotic systems that can enhance the safety and efficacy of emergency-response missions, amongst others, in urban terrain.

This study seeks to extend previous DST Group research on PF-based algorithms for source localisation performed through simulation [Ristic and Arulampalam, 2012]. In the literature, applications of PF-based search and tracking algorithms to real-world data are scarce, particularly for the more recent algorithms such as those based on the Bernoulli PF. This study addresses the need for real-world data for use in assessing the performance of a PF-based BOT algorithm.

The experiment described here utilises a small radio-controlled unmanned ground vehicle (UGV) carrying an array of photodiodes. A floodlight serves as the target/source. The photodiode measurements, along with motion-tracking data for the UGV, are processed to estimate the bearing of the light source relative to the UGV. Reflective objects in the laboratory and other sources of stray light create noise in the bearing measurements and false detections. The Bernoulli PF-based localisation algorithm [Mahler, 2004] is applied to the accumulated dataset. The UGV's trajectory and the parameters of the BOT algorithm are varied to examine their effects on the accuracy of the resulting source-position estimate.

3 Bearings-Only Tracking

This section provides a brief summary of the mathematical models used in the Bernoulli PF and a description of the algorithm employed in the current experiments. The interested reader is referred to Ristic and Arulampalam [2012] for more details.

3.1 Mathematical Models

Let $\mathbf{x}_k = [x_k, y_k]^T$ denote the target state at time index k, corresponding to its position in Cartesian coordinates. The target is assumed to be stationary so that $\mathbf{x}_k$ is invariant with time. To model target appearance/disappearance, a binary random variable, ε_k, is introduced:

$$\varepsilon_k \in \{0,1\}.$$

The dynamics of ε_k is modelled by a two-state Markov chain with a known transition-probability matrix (TPM). The TPM is completely defined by:

$$p_b \equiv P\{\varepsilon_{k+1} = 1 | \varepsilon_k = 0\},$$

the probability of target 'birth'; and

$$p_s \equiv P\{\varepsilon_{k+1} = 1 | \varepsilon_k = 1\},$$

the probability of target 'survival'.

At each time index k, a set of bearing measurements $Z_k = \{z_{k,1}, ...z_{k,m_k}\}$ is captured. Due to the imperfections of the sensor, at most one measurement is from the target and the rest are false detections. A target-originated measurement can be modelled as

$$z = h(\mathbf{x}_k) + w_k, \quad (1)$$

where w_k is a zero-mean independent Gaussian noise with variance σ_θ^2 and

$$h(\mathbf{x}_k) = \tan^{-1}\left(\frac{y_k}{x_k}\right) \quad (2)$$

is the four-quadrant inverse tangent function that gives the true target bearing at time index k evaluated from values over the interval, $z \in (-\pi, \pi]$. A target-originated measurement is detected with probability p_D, a parameter set *a priori* by use of knowledge of the sensor.

The spatial distribution of false detections (clutter) over z is time invariant and denoted by $c(z)$. The number of clutter points per bearing-measurement scan is assumed to be Poisson distributed, with a mean of λ.

The problem is to detect the existence of a target, and, if one is present, to estimate its kinematic state (position), $\mathbf{x}_k$. In the Bayesian setting, this leads to the estimation of two posteriors: (1) the probability of target existence,

$$q_{k|k} = p\{\varepsilon_k = 1 | Z_{1:k}\},$$

where $Z_{1:k} = \langle Z_1, ..., Z_k \rangle$ is the sequence of measurement sets up to the current time index k; and (2) the spatial probability-density function (PDF) of the target,

$$s_{k|k}(\mathbf{x}) = p(\mathbf{x}_k | Z_{1:k}).$$

3.2 Bernoulli Particle Filter

The Bernoulli filter models the target state at time index k as a random finite set and is the optimal sequential Bayesian estimator for the problem described above [Mahler, 2004]. The prediction equations of the Bernoulli filter [Ristic and Arulampalam, 2012] are:

$$q_{k+1|k} = p_b(1 - q_{k|k}) + p_s q_{k|k} \text{ and} \quad (3)$$

$$s_{k+1|k}(\mathbf{x}) = \frac{p_b(1 - q_{k|k}) b_{k+1|k}(\mathbf{x})}{q_{k+1|k}} + \frac{p_s q_{k|k} \int p_{k+1|k}(\mathbf{x}|\mathbf{x}') s_{k|k}(\mathbf{x}') d\mathbf{x}'}{q_{k+1|k}}. \quad (4)$$

The term $p_{k+1|k}(\mathbf{x}|\mathbf{x}')$ is the target transitional density, which for a stationary target is the delta function; and $b_{k+1|k}(\mathbf{x})$ is the spatial distribution of predicted 'target birth'. This predicted target-birth density is computed using the birth density $b_k(\mathbf{x})$ at the previous time – typically generated using the measurement set at the previous time index [Ristic and Arulampalam, 2012].

Given a new measurement set, Z_{k+1}, the Bernoulli filter is updated as follows. The probability of existence is updated by use of the expression:

$$q_{k+1|k+1} = \frac{1-\delta_{k+1}}{1-\delta_{k+1}\, q_{k+1|k}}\, q_{k+1|k}, \qquad (5)$$

where

$$\delta_{k+1} = p_D\left(1 - \sum_{z\in Z_{k+1}} \frac{\int g_{k+1}(z|\mathbf{x})\, s_{k+1|k}(\mathbf{x})\, d\mathbf{x}}{\lambda\, c(z)}\right). \qquad (6)$$

Here, λ and $c(z)$ are the previously defined clutter parameters; and $g_{k+1}(z|\mathbf{x})$ is the measurement likelihood function for a target-originated measurement, which is computed by use of Equation (1). Finally, the target spatial PDF is updated as

$$s_{k+1|k+1}(\mathbf{x}) = \frac{1 - p_D + p_D \sum_{z\in Z_{k+1}} \frac{g_{k+1}(z|\mathbf{x})}{\lambda c(z)}}{1-\delta_{k+1}}\, s_{k+1|k}(\mathbf{x}). \qquad (7)$$

The Bernoulli PF is essentially a sequential Monte Carlo implementation of the Bernoulli filter. It approximates the spatial PDF, $s_{k|k}(\mathbf{x})$, by a set of N weighted random samples (particles):

$$\left\{w^i_{k|k}, \mathbf{x}^i_{k|k}\right\}_{i=1}^{N},$$

where $\mathbf{x}^i_{k|k}$ is the state of particle i and $w^i_{k|k}$ is its corresponding weight. This weighted set of particles is then used in each recursion to compute the integrals appearing in the above equations using Monte Carlo integration. Finally, the target-state estimate at time k can be computed as a weighted average:

$$\hat{\mathbf{x}}_{k|k} = \sum_{i=1}^{N} w^i_{k|k}\, \mathbf{x}^i_{k|k}. \qquad (8)$$

4 Experimental Arrangement

4.1 Unmanned Ground Vehicle

A Dagu robot with a 'Wild Thumper' chassis and a 75:1 gearbox, pictured in Figure 1, is used as the basis for the UGV. It is low-cost, as well as being compact (43 cm long, 30 cm wide, and 14 cm tall) and lightweight, which reduces risks to personnel during the experiment. The system comprises mostly off-the-shelf, hobby-grade components, coupled with a Raspberry Pi 2 mini-computer and an Arduino-based Dagu 'T'Rex' motor-controller board. A 2.4-GHz Spektrum radio-control system is used to manually pilot the UGV. The motor-controller board mixes the two-channel radio-control signals (throttle and steering) and governs the high-current H-bridge circuits used to drive the left and right motors of the UGV.

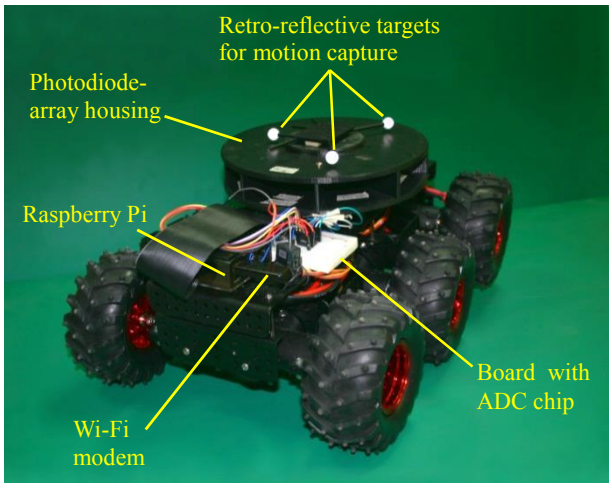

Figure 1: The UGV used in the experiments

The experiments are conducted within a cleared rectangular region measuring 2.5 m × 4 m. The UGV is isolated from the experimenters by a knee-high barricade, which provides an additional fail-safe to prevent injuries from potential collisions with the UGV.

A number of UGV drive patterns are used to provide a variety of datasets for input to the BOT algorithm. The trajectories include lateral passes, differently oriented triangles, rectangles, circular patterns, and snaking patterns. In total, thirteen different trajectories are chosen; and the UGV is piloted manually via radio control to approximate as closely as possible each pattern. The trajectory patterns are displayed in Figure 2, which also indicates the location of the light source with respect to the region in which the UGV travelled. Note that the trajectories are recorded with a motion-capture system, described in the next subsection, and those trajectories are input directly to the BOT algorithm for processing. No other model of the kinematics of the platform is utilised in this proof-of-concept experiment.

4.2 Motion Tracking

An OptiTrack motion-capture system is used to record the position and orientation of the UGV throughout the experiment. Six OptiTrack Prime 17W cameras are mounted on the ceiling of the laboratory, along the perimeter of the experimental space; and OptiTrack's Motive:Tracker software is used to track infrared (IR)-reflective markers fixed to the UGV. Three markers are placed asymmetrically on the UGV, as shown in Figure 1, so that its orientation can be monitored. The error in the recorded position of the UGV is estimated to be less than ±1 mm and the error in its orientation is estimated to be less than ±1°.

A Netgear ProSafe network switch (supplied by OptiTrack) is used to connect the cameras, an eSync time-synchronisation unit, and the PC running the Motive:Tracker software. The software is configured to broadcast frames of tracking data to another PC running a MATLAB™ data-acquisition script, which records the readings at a rate of 5 Hz.

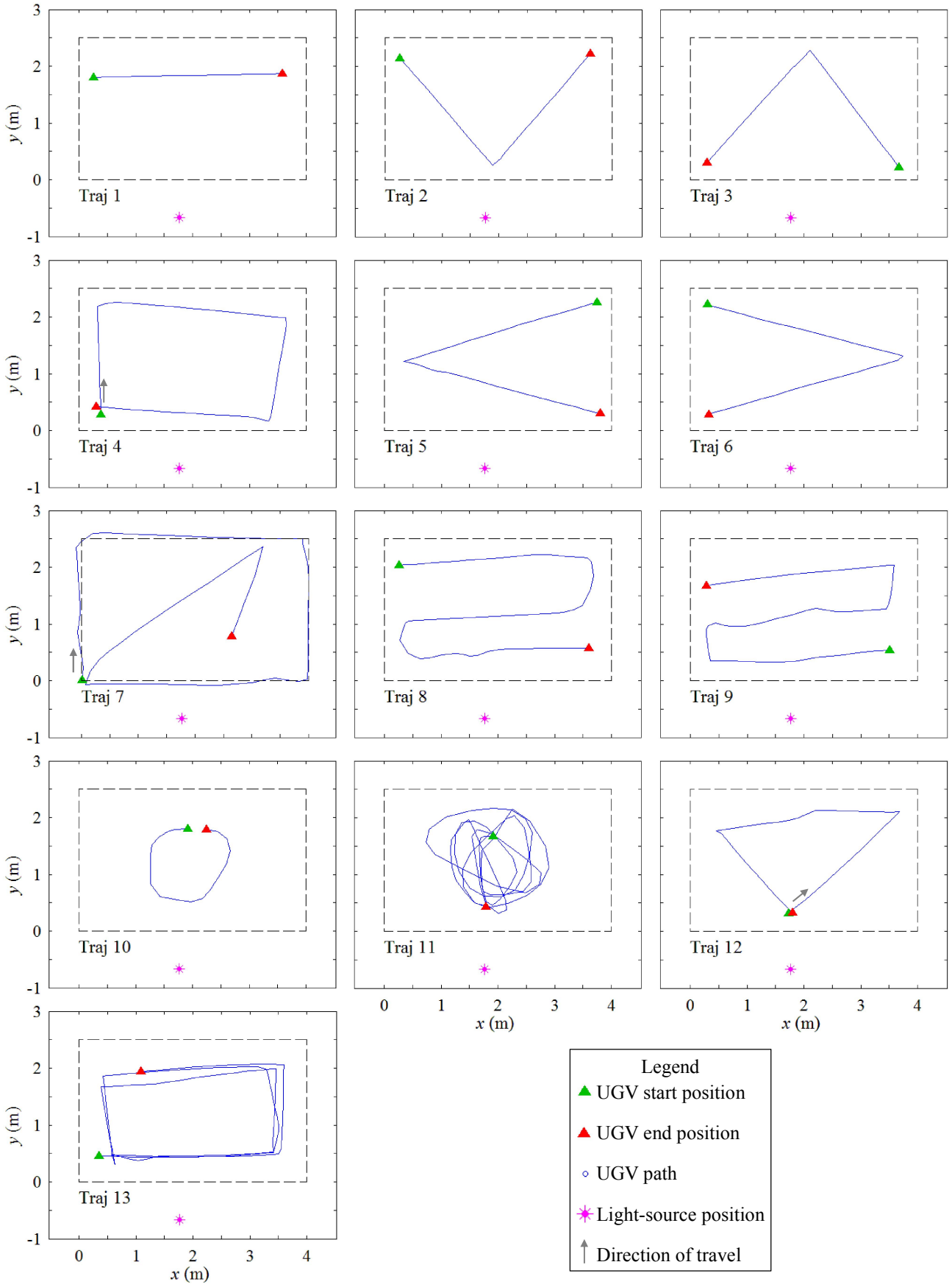

Figure 2: UGV trajectories used in the experiments

4.3 Light Source and Sensor

An Arlec 500-W portable halogen lamp is used as the light source. It has a rectangular aperture that is 16 cm wide and 12 cm tall; and the centre of the lamp (the horizontal bulb) is positioned 17.4 cm from ground level. As shown in Figure 2, it is located outside the 2.5-m × 4.0-m region in which the UGV travels, at a perpendicular distance of 66.5 cm from one of the long edges of the rectangle and 23.1 cm laterally from the centre of that side. Its face is approximately parallel to the long side of the rectangle, and its location relative to the near-right corner of the region is recorded with the motion-capture system.

Silicon photodiodes from Osram (Opto BPW 21) are used to create a sensor to detect the bearing of the light with respect to the UGV. The photodiodes are sensitive at visible wavelengths (350–820 nm). At the 800-nm wavelength of the IR pulses emitted by the OptiTrack system, their sensitivity falls to ~ 10% of its peak; thus, the selected photodiodes minimise false detections caused by light from the motion-tracking system.

Eight photodiodes are arranged in a circular array and mounted atop the UGV in a housing printed in ABS plastic. Figure 3 shows its design, along with the placement of the photodiodes and other system components. The housing prevents direct exposure of the photodiodes to light from above (*e.g.*, from the motion-tracking system) and limits the angle of horizontal acceptance of each photodiode to 45°. When mounted on the UGV, the photodiodes are 17.6 cm from the ground and oriented so that one faces directly ahead of the UGV (normal to its main – longitudinal – axis).

A 10-bit analogue-to-digital converter (a Microchip Technology, Inc., MCP3008 ADC chip) mounted on a small board atop the UGV samples the voltage output of the photodiode circuit, providing the relative intensity of the light incident on each photodiode. The ADC chip is digitally interfaced to the Raspberry Pi via an SPI serial interface. The data are transmitted from the Raspberry Pi to the computer running the MATLAB data-acquisition script via a Wi-Fi modem and recorded at a rate of 5 Hz.

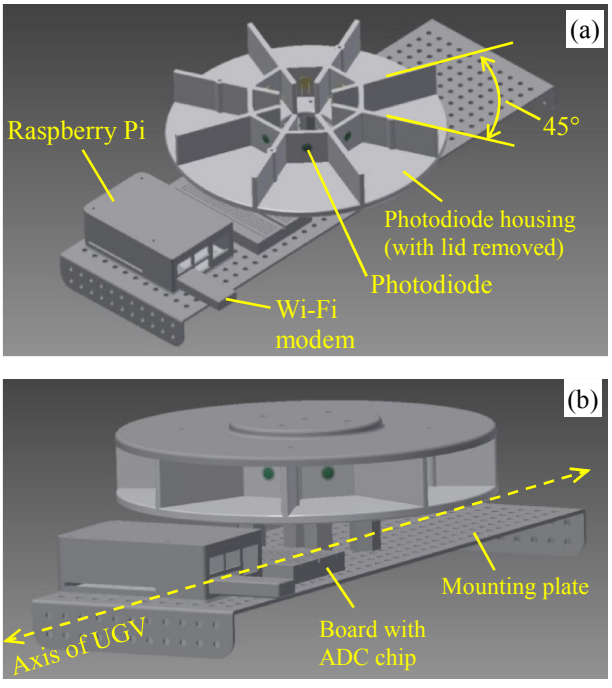

Figure 3: (a) Internal and (b) side views of the custom-made photodiode housing and other system components sitting atop the Dagu robot's mounting plate, which is 38 cm long and 12 cm wide

4.4 BOT-Algorithm Implementation

A computer running the Windows 7 operating system with an Intel i7 3.40-GHz processor and 8 GB of RAM is used to execute the BOT algorithm. It is implemented in a MATLAB script that utilises the simultaneously recorded time-histories of the UGV position and orientation and sensor readings as inputs. During each run, the script can display an animation showing the actual source location, the track of the UGV, its current position, the particles forming the posterior PDF of the source location, and the estimated source position. This permits the user, if desired, to observe the behaviour of the algorithm during processing. A frame from the animation created during the analysis of the dataset acquired with trajectory 6 is given in Figure 4. For each run, the processing time and the x- and y-coordinates of the final source-position estimate are also recorded.

The BOT algorithm utilises several assumptions and requires *a priori* knowledge of parameters related to the characteristics of the tracked source and the operating environment. Firstly, the source is assumed to be omnidirectional. This is obviously untrue for the current experiment, but (at least partially) overcome by the use of a wide-angle (flood)light and its position outside, but facing, the region in which the UGV travels – thus being relatively uniform from where it is observed.

A value for the source-detection probability (p_D), a measure of the sensor's response to source signals, is required. Preliminary analysis of the sensor data indicates that the source is highly visible amongst the clutter created by reflections and other stray light in the laboratory from close-up to distances greater than 5 m. A process of trial-and-error indicates that a value of p_D equal to 0.92 is suitable. The measurement range of the sensor is also required, to restrict the particle-position

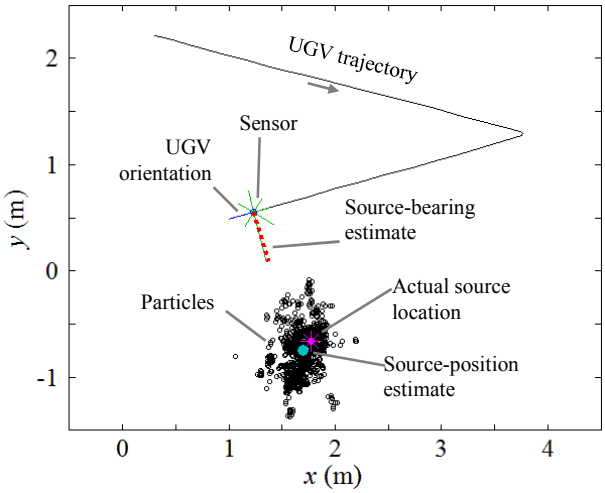

Figure 4: A frame from an animation created during processing of data acquired with trajectory 6, with 3,000 particles used in the BOT algorithm

estimates to realistic values. Here, the sensing range of the photodiodes is taken to be 5 cm – 5 m, based on the size of the laboratory in which the experiments are conducted and the basic geometry shown in Figure 2.

The orientations of the photodiodes are defined in a MATLAB routine that computes the source bearing relative to the longitudinal axis of the UGV (and hence to the normal of the forward-looking photodiode). For each sample of eight light-intensity measurements, the photodiode with the largest signal is identified; and a quadratic function describing the light intensity as a function of bearing angle is fitted to the signals recorded by that photodiode and the ones on either side of it. The bearing angle at which the quadratic function has its maximum value is taken to be the source bearing.

As described in Section 3.1, the algorithm also includes a parameter for the standard deviation of the random source-bearing error (σ_θ). Through trial-and-error, a value of 20° is chosen. It is also representative of the resolution of the individual photodiode measurements (±22.5°).

Source intermittency is modelled with two additional parameters: the probability of particle birth between updates (p_b, as defined in Section 3.1), used so that if the source is lost it can be found again; and the probability of particle survival between PF updates (p_s), used to remove a previously identified source when measurements cease to point to it. Here, values of $p_b = 0.01$ and $p_s = 0.99$ are chosen based on simulations of similar tracking problems [Ristic and Arulampalam, 2012].

Finally, the algorithm employs a parameter for the average number of clutter points per measurement scan (λ, defined in Section 3.1) and models the spatial distribution of the clutter signal ($c(z)$). Here, the clutter is assumed to be uniformly distributed; and λ is taken to be 0.2.

5 Results and Discussion

5.1 Effect of UGV Trajectory

The source-position estimate provided by the BOT algorithm is examined as a function of UGV trajectory by executing the algorithm with a dataset collected with each of the trajectories shown in Figure 2. To reduce the effect of run-to-run differences caused by the random functions used in calculating posteriors in the PF (*e.g.*, in the Monte Carlo integration process), each dataset is processed 1,000 times. The results, computed with 3,000 particles, are averaged; and their standard deviations are computed. Table 1 provides the mean source-position estimate for each trajectory, in terms of the errors in the *x*- and *y*-directions relative to the centre of the light source, Δx and Δy, respectively, and the total source-position error ($\sqrt{\Delta x^2 + \Delta y^2}$). Results from an analysis of the spatial characteristics of each trajectory are given, along with the number of samples and the mean processing time for the dataset.

The data for the mean processing time for the various trajectories show that it increases linearly with the number of samples acquired. On average, just over 1 s is required to compute 25 state updates of the 3,000-particle BOT algorithm on an Intel i7 computer.

The mean source-position estimate for each trajectory is plotted in Figure 5. The standard deviations associated with the source-position estimates (shown by the error bars) are measures of random variation in the 1,000 estimates included in each mean. Those errors are distinct from, though related to, the standard deviations of the *x*- and *y*-oriented spreads of the 3,000 particles used to generate the individual estimates included in each mean. (See Figure 4 for an example of the relative spreads of the particles used to estimate the source position each time the algorithm is executed.)

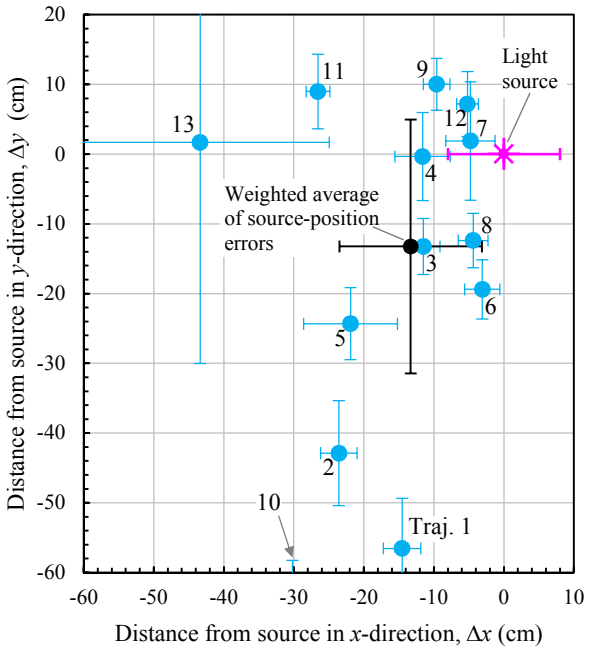

Figure 5: Mean source-position estimate relative to the actual position of the source, computed by the BOT algorithm with 3,000 particles for each trajectory; and the location and horizontal extent of the light source

The data in Figure 5 and Table 1 indicate that the mean source-position estimate varies significantly with trajectory, with differences amongst the estimates being larger than the characteristic length of the source (its horizontal aperture size, 16 cm, represented by the error bars about the symbol representing the centre of the light source). Excluding one outlier with very large standard deviations (the result for trajectory 13), the source-position estimates are clustered about a point 13.3 cm to the left of the light source and 13.2 cm behind it. This position is a weighted average of the means for the trajectories, evaluated with the inverse standard deviations used as the weightings, and is also shown in Figure 5. The error bars displayed with this point represent the standard deviations of its *x*- and *y*-coordinates.

The distance from the centre of the light source to the weighted-average estimate for all trajectories (except 13) is only slightly larger than the horizontal aperture of the source; and five of the trajectories (7, 12, 4, 8, and 9) yield estimates within one characteristic length of the centre of the actual source. However, a systematic error in the mean source-position estimates is suggested by the data presented in Figure 5: all of the errors in the *x*-direction are negative, whereas the errors in the *y*-direction are both positive and negative.

The systematic error may be explained by a small, constant error in the measured orientation of the UGV.

Traj. no.	No. of measure-ments	Std. dev. of traj. in *x*-direction (m)	Std. dev. of traj. in *y*-direction (m)	Mean distance from source to traj. (m)	Std. dev. of distance from source to traj. (m)	Fraction of time spent in near-right corner	Mean source-position error in *x*-direction (cm)	Std. dev. of estimate error in *x*-direction (cm)	Mean source-position error in *y*-direction (cm)	Std. dev. of estimate error in *y*-direction (cm)	Mean absolute source-position error (cm)	Std. dev. of source-position error (cm)	Mean run-time (s)
7	99	1.564	1.063	2.437	0.877	0.11	–4.8	3.5	1.9	8.5	5.1	4.5	4.1
12	71	1.172	0.599	2.510	0.741	0.00	–5.2	1.5	7.2	4.7	8.9	3.9	2.9
4	82	1.368	0.865	2.306	0.729	0.20	–11.6	4.0	–0.3	6.3	11.6	4.0	3.3
8	73	1.202	0.680	2.281	0.704	0.03	–4.4	2.1	–12.4	3.9	13.1	3.7	3.0
9	77	1.258	0.611	2.153	0.613	0.08	–9.6	1.9	10.0	3.7	13.9	3.0	3.1
3	38	0.981	0.747	2.370	0.432	0.13	–11.5	2.4	–13.2	4.0	17.5	3.4	1.5
6	45	1.327	0.555	2.449	0.583	0.00	–3.1	2.5	–19.4	4.2	19.6	4.2	1.8
11	100	0.473	0.529	1.996	0.526	0.00	–26.5	1.7	9.0	5.3	28.0	2.3	4.1
5	44	1.236	0.562	2.291	0.536	0.09	–21.9	6.7	–24.3	5.2	32.7	5.9	1.8
13	200	1.249	0.697	2.283	0.654	0.13	–43.4	18.5	1.7	31.7	43.4	18.5	8.2
2	32	0.999	0.717	1.988	0.883	0.00	–23.6	2.6	–42.9	7.5	48.9	6.7	1.3
1	24	1.234	0.024	2.771	0.193	0.00	–14.5	2.7	–56.5	7.2	58.4	7.0	1.0
10	25	0.450	0.490	2.086	0.477	0.00	–30.2	3.0	–68.8	10.5	75.1	9.7	1.0

Table 1: Trajectory characteristics (number of UGV-position and -orientation and source-bearing measurements acquired, standard deviation of trajectory path, mean distance from source, and fraction of time spent in the near-right corner) and BOT results for each trajectory (mean source-position errors, standard deviations of errors, and run time), averaged over 1,000 runs of the BOT algorithm performed with 3,000 particles

This source of error is identifiable when the data for the trajectories are processed. The error originates during preparation for the experiment, when a UGV-orientation reference is created. The UGV is positioned in the experimental space with its longitudinal axis at a defined orientation and imaged by the motion-capture system. This serves as a reference from which clockwise changes are recorded as the UGV is tracked. The process relies on alignment of the UGV in a particular direction by eye and is thus prone to error. A comparison of the UGV orientation inferred from straight portions of several of the trajectories shown in Figure 2 with the mean UGV orientation recorded by the motion-capture system reveals a constant error of –2° in the recorded orientation. The motion-capture data is corrected for this error when it is processed to obtain the results shown in Table 1 and Figure 5.

A comparison of mean source-position estimates generated by the BOT algorithm with and without the 2°-correction demonstrates that the correction, on average, halves the *x*-direction error in the mean source-position estimate for each trajectory (with the exception of trajectory 12, for which the error increases by 1 cm when the correction is applied). The fact that the estimates still lie exclusively to the left of the actual source position (Figure 5) suggests that an additional, as-yet unidentified systematic error may exist in the UGV-orientation measurements. Based on the results obtained with and without the 2°-correction, the unidentified error is estimated to be between –2° and –4°.

Although this small systematic error makes comparative assessments of the different trajectories difficult, some observations may be made about the characteristics of the source-position estimates as a group. For example, the standard deviations of the mean source-position estimates indicate that the uncertainties are consistently larger in the *y*-direction than in the *x*-direction. This is in accordance with the spread of the particles shown in Figure 4 (and hence the standard deviations of the *x*- and *y*-coordinates of the resultant source-position estimate). On average, the standard deviation of the position error in the *y*-direction is twice that of the error in the *x*-direction. The disparity in the relative magnitudes of the uncertainties is likely caused by the fact that the region in which the UGV travels is exclusively in front of the light source. It is likely not to be influenced strongly by the suspected systematic error, as the standard deviations of the *x*- and *y*-coordinates of the mean source-position estimate for each trajectory change relatively little with the 2°-correction to the UGV-orientation data.

It is also useful to examine the trajectory that leads to a mean source-position estimate with large uncertainties: trajectory 13. The animation displayed as the BOT algorithm is executed reveals significant clutter in light-intensity data acquired in the right-hand corner of the UGV's region of travel, nearer to the light source (approximately defined by $x > 3.3$ m, $y < 0.7$ m). Signal clutter is created because the source is not visible in that region, as the UGV is outside the horizontal spread of the floodlight. Ambient room lighting and reflections of the source from the laboratory walls are thus the only sources seen by the sensor.

Figure 6 shows a frame from the animation generated with trajectory 13 at the point when the UGV has just reached that corner (for the 3rd time). While the algorithm is designed to operate in the presence of false detections, the sensor signals in this region contain clutter that exceeds the threshold value (related to λ, as defined in Section 3.1). The clutter is spatially non-

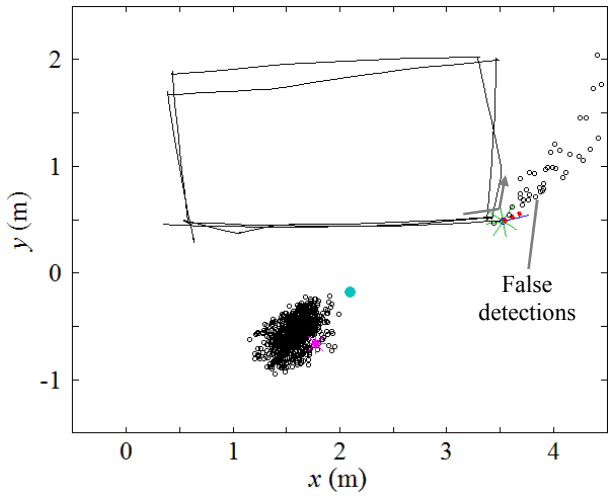

Figure 6: A frame from the MATLAB-generated animation during processing of data acquired with trajectory 13, with 3,000 particles used in the BOT algorithm

homogeneous and hence violates the assumptions of the Bernoulli PF implementation. Therefore, before the UGV leaves the near-right corner, the source-position estimate computed by the BOT algorithm migrates to a spot in the right-hand corner of the laboratory farther from the source. As the UGV proceeds along its path, the light source is again seen by the sensor, and the source-position estimate returns to a location close to the actual source. However, there are insufficient state updates for the 3,000-particle BOT algorithm to regain certainty about the source-position estimate. As shown in Table 1, approximately 13% of the samples taken during trajectory 13 are in the near-right corner, where the source intensity is negligible.

Trajectory 4 is a similar rectangular path with the same starting position as trajectory 13. It places the UGV in the near-right corner for 20% of the sampling time; however, trajectory 4 ends where it starts, completing about two-thirds of the rectangular path shared by the trajectories after it passes through the near-right corner. The longer path (and greater number of state updates) after the corner in which clutter predominates significantly increases the certainty of the final source-position estimate for trajectory 4, compared with trajectory 13. The greater certainty is evidenced by the fact that the standard deviations of the x- and y-coordinates of the final source-position estimate for trajectory 4 are equal to ~20% of the values for trajectory 13.

The final source-position estimate for trajectory 4 is also significantly closer to the actual source location than is the estimate obtained with trajectory 13; however, this may be an artefact of the systematic error believed to be present in the data. The differences in the source-position estimates provided by different trajectories will be reconsidered if the experiment is repeated with more care to quantifying the UGV's orientation.

Random error associated with the orientation of the UGV may also significantly contribute to random error in the source-position estimates. The random error in the UGV-orientation measurement is exacerbated by the small spacing amongst the retro-reflective markers used to track the UGV as it moves through the experimental space. If the experiments are repeated, the markers will be place farther apart.

5.2 Effect of Number of Particles

As noted previously, the BOT algorithm is implemented in a MATLAB script that uses as inputs previously recorded time-histories of trajectory and light-sensor data. This post-processing permits an analysis of the BOT algorithm's performance with different numbers of particles (N, defined in Section 3.2).

Figure 7 shows the mean error in the source-position estimate and the mean processing time for the algorithm, obtained by processing the dataset acquired with trajectory 3 with 10 to 10,000 particles. For each different number of particles employed in the PF, the results are averaged over 100 runs to reduce run-to-run variations created by the random functions in the algorithm.

When 300 particles are used, the results (a mean absolute source-position error of 17.9 cm with a standard deviation of 3.5 cm) are nearly identical to those obtained with 3,000 particles, shown in Table 1. Computations with more than 300 particles do not significantly reduce the error in the mean source-position estimate: with 10,000 particles, it settles at a value just less than 17.5 cm with a standard deviation of 3.4 cm.

In contrast, the mean run-time for the algorithm increases rapidly as the number of particles increases. Because of the many transformations of two-dimensional matrices used in the BOT algorithm, the processing time is expected to be proportional to the square of the number of particles used. This is verified by fitting the mean run-time as a function of the number of particles with a 2nd-order polynomial. The coefficient of determination (R^2) for the fit, displayed in Figure 7, is greater than 0.999.

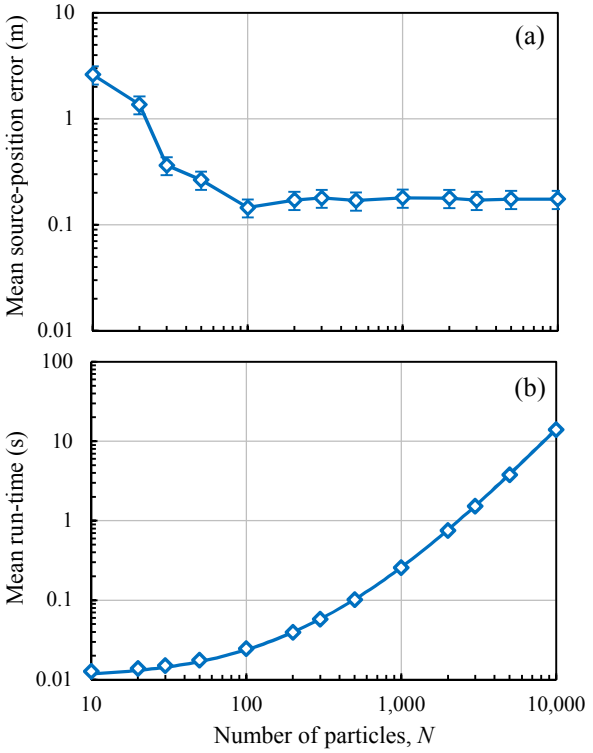

Figure 7: (a) Mean error in source-position estimate and (b) mean processing time of the BOT algorithm as functions of the number of particles, averaged over 100 runs, for data captured with trajectory 3

5.3 Localisation-Error Distribution

Varying numbers of runs of the BOT algorithm are also performed with the dataset obtained in trajectory 3 with a fixed number of particles (3,000), to examine the distributions of the errors in the x- and y-directions and of the total source-position error ($\sqrt{\Delta x^2 + \Delta y^2}$). The results give an indication of the effect of the random functions in the BOT algorithm, as they provide the only source of run-to-run variation.

In the limit of a large number of runs, the errors in the x- and y-coordinates of the source-position estimate are found to have Gaussian distributions, even though the particles in the PF are not (strictly) independent and identically distributed. However, the non-linear dependence of the total source-position error on the x- and y-oriented errors produces a skew-normal distribution for the total error, as shown by the histograms in Figure 8 for 1,000, 10,000, and 30,000 runs of the BOT algorithm. This effect also leads to a small difference between the source-position error obtained from the means of the x- and y-coordinates of the source estimate produced during each run of the algorithm (17.5 cm, as shown for trajectory 3 in Table 1) and the mean of the total source-position error for each run (18.1 cm).

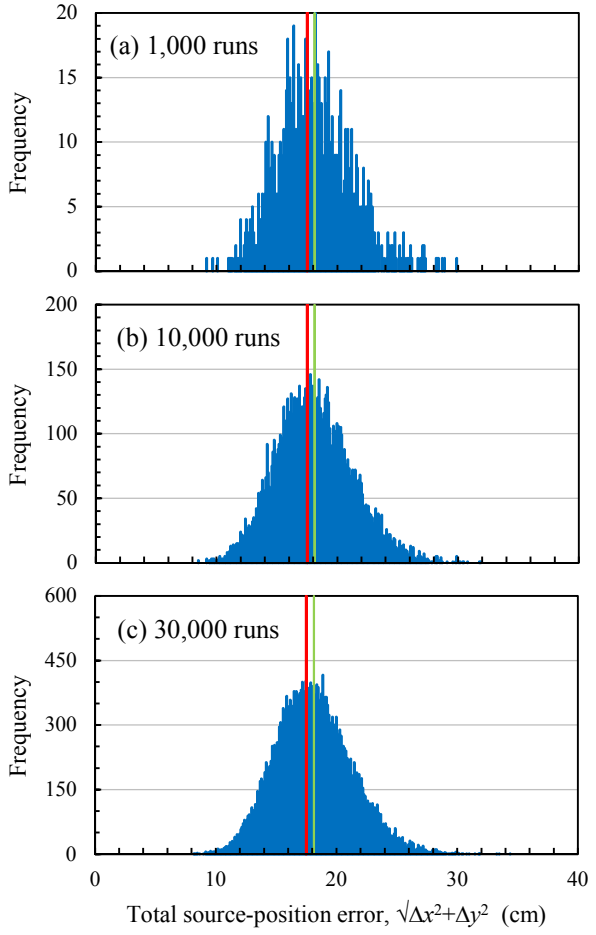

Figure 8: Histograms of source-position error for trajectory 3 obtained with different numbers of BOT-algorithm runs using 3,000 particles; and source-position error evaluated from the means of Δx and Δy for each run (shown in red) and from the mean of $\sqrt{\Delta x^2 + \Delta y^2}$ for each run (in green)

6 Conclusion

In this research, a Bernoulli PF-based BOT algorithm is tested by use of a simple experimental arrangement. The aim is to provide a proof-of-concept of the technique and real-world data against which it and similar localisation algorithms can be tested. A bright floodlight is used as the source (target) to be localised (tracked); and a circular array of photodiodes forms a sensor that measures the relative intensity of the light in a horizontal plane, providing low-resolution bearing measurements over 2π radians for input to the BOT algorithm. A radio-controlled UGV carries the sensor and is driven in a variety of simple trajectories, while a motion-capture (camera) system is used to record its motion.

A Bernoulli PF is chosen because of its robustness to false detections and intermittent source loss; and the results presented in this paper demonstrate that the algorithm performs well, providing source-position estimates that are, on average, only 19 cm from the centre of the actual source. However, several limitations in the experimental arrangement are apparent. For example, the (flood)light source has a finite spatial dimension and provides a source signal over less than π radians, rather than being a point source with uniform intensity over 2π radians (an omnidirectional source), as is assumed by the BOT algorithm. This lack of omnidirectionality requires that the sensor acquire signals exclusively in front of the source, rather than circling it, which might be expected to improve localisation.

Datasets obtained with thirteen different trajectories are processed with the BOT algorithm to assess its performance as a function of trajectory pattern. The source-position errors obtained in 1,000 independent runs for each trajectory are averaged to reduce the effect of randomness in the algorithm. The results indicate that most trajectories produce source-position estimates close to the actual source location, though an apparent systematic error in the UGV-orientation data makes a comparative analysis of the accuracy produced with different trajectories difficult.

Low light-source levels in one corner of the experimental space create false detections that the algorithm is unable to fully reject. For one trajectory (13), the presence of clutter produces large uncertainties in the final source-position estimate. During this trajectory, the source is lost when the UGV transits the area where clutter predominates. An estimate close to the actual source position is regained afterward, but the algorithm has an insufficient number of samples to arrive at a result with low uncertainty before the trajectory's end.

The mean source-position error is also assessed with different numbers of particles in the BOT algorithm for a trajectory (3) that yields a mean source-position estimate typical of the entire set. The results demonstrate that, above a certain number of particles (300), increasing the number provides little benefit, though it comes at the cost of higher processing time, which rises with the square of the number of particles.

Finally, the effect of the random functions employed in the BOT algorithm is examined by processing the dataset acquired with the same trajectory in a large number of independent runs (up to 30,000). The error in the source-position estimate is seen to have a normal distribution about a mean very similar to that obtained with 1,000 runs. The standard deviation of the error distribution is found to be small compared with the size of

the light source, indicating that, for that particular trajectory, a single run would likely yield a source-position estimate with a high degree of certainty.

These findings suggest that real-time processing on-board an airborne or ground-based robot could yield source-position estimates with adequate accuracy and certainty. On-board processing could also enable a robot to autonomously manoeuvre in response to the feedback from the BOT algorithm and to thus improve its source-position estimate. However, PF-based algorithms, including the BOT algorithm tested here, are computationally demanding; and real-time processing can be performed by the Raspberry Pi minicomputer on-board the UGV used here only with small numbers of particles. The single trajectory examined in detail in this paper required just 300 particles to produce a source-position estimate with a relatively high certainty; thus, it would have been possible to compute on-board the UGV at 5 Hz. However, other trajectories may require more particles to reach an acceptable level of certainty and thus may not be suitable for on-board processing.

The simulated search-and-localisation mission presented here also indicates some of the other challenges associated with real-world applications of algorithms for source/target tracking. Given that the BOT algorithm can utilise measurements from a variety of source and sensor types (light, sound, chemical, radiation, thermal, *etc.*), this initial proof of concept suggests paths for future research on localisation of other source types of interest, including chemical, biological, and radiative sources in unknown environments.

Acknowledgements

This research is supported by DST Group's Strategic Research Initiative on Trusted Autonomous Systems. The authors wish to acknowledge Paul Jacquemin for his assistance in the design and fabrication of the photodiode housing.

References

[Aidala, 1979] Vincent J. Aidala. Kalman filter behaviour in bearings-only tracking applications. *IEEE Trans. Aerospace and Electronic Systems*, AES-15(1): 29–39, January 1979.

[Aidala and Nardone, 1982] Vincent J. Aidala and Steven C. Nardone. Biased estimation properties of the pseudo-linear tracking filter. *IEEE Transactions on Aerospace and Electronic Systems*, AES-18(4): 432–441, July 1982.

[Aidala and Hammel, 1983] Vincent J. Aidala and Sherry E. Hammel. Utilization of modified polar coordinates for bearings only tracking. *IEEE Trans. Automatic Control*, AC-28(3): 283–294, March 1983.

[Anon., 2012] Anonymous. Executive Overview: IHS Jane's All the World's Aircraft: Unmanned. Jane's Unmanned Aerial Vehicles and Targets, 2012.

[Arulampalam and Ristic, 2000] M. Sanjeev Arulampalam and Branko Ristic. Comparison of the particle filter with range parametrised and modified polar EKFs for angle-only tracking. *Proc. of the SPIE, Signal and Data Processing of Small Targets*, Vol. 4048, pages 288–299, Orlando, FL, 24 April 2000. SPIE.

[Arulampalam, et al., 2002] M. Sanjeev Arulampalam, Simon Maskell, Neil J. Gordon and Tim Clapp. A tutorial on particle filters for non-linear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Processing*, 50(2): 174–188, February 2002.

[Arulampalam, et al., 2007] Sanjeev Arulampalam, Martin Clark and Richard Vinter. Performance of the shifted Rayleigh filter in single sensor bearings-only tracking. *10th International Conference on Information Fusion*, Quebec, Canada, 9–12 July 2007.

[Gordon, et al., 1993] Neil J. Gordon, D. J. Salmond and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proc. F*, 140(2): 107–113, April 1993.

[Mahler, 2004] Ronald P. S. Mahler. *Statistical multi-source-multitarget information fusion*. Artech House, Norwood, MA, 2004.

[Nardone and Graham, 1997] S. C. Nardone and M. L. Graham. A closed form solution to bearings-only target motion analysis. *IEEE Journal of Oceanic Engineering*, 22(1): 168–178, January 1997.

[Peach, 1995] N. Peach. Bearings-only tracking using a set of range-parametrised extended Kalman filters. *IEEE Proc. Control Theory and Applications*, 142(1): 73–80, January 1995. IEEE.

[Ristic, et al., 2004] Branko Ristic, Sanjeev Arulampalam and Neil J. Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House, Norwood, MA, 2004.

[Ristic and Arulampalam, 2012] Branko Ristic and Sanjeev Arulampalam. Bernoulli particle filter with observer control for bearings-only tracking in clutter. *IEEE Trans. Aerospace and Electronic Systems*, 48(3): 2405–2415, July 2012.

[Ristic, et al., 2013] Branko Ristic, V. Ba-Tuong, V. Ba-Ngu and A. Farina. A tutorial on Bernoulli filters: Theory, implementation and applications. *IEEE Trans. Signal Processing*, 61(13): 3406–3430, July 2013.

[Yan, *et al.*, 2014] Z. Yan, D. Wang and J. Li. Consensus 3-D bearing-only tracking in switching sensor networks. *Signal Processing*, 105: 148–155, December 2014.