Turk J Elec Eng & Comp Sci, Vol.20, No.Sup.1, 2012, © TÜBİTAK
doi:10.3906/elk-1103-4

Observer path design by imitation of competing constraints for bearing only tracking

Rıdvan GÜRCAN[1,*], Mesut KARTAL[2]
Informatics Institute, Department of Satellite Communication and Remote Sensing,
İstanbul Technical University, İstanbul-TURKEY
e-mail: ridvan.gurcan@bte.tubitak.gov.tr

Received: 04.03.2011

Abstract

In the bearing only tracking (BOT) problem the observer tries to estimate the position of the target. The observer maneuver is a requirement to guarantee the uniqueness of the estimation. Determining an observer maneuver for BOT is a difficult problem due to its nonlinear nature. This paper proposes a new method that combines the reference BOT scenarios as a priori knowledge with the incoming bearing measurements to obtain an observer maneuver in a generic optimization framework. The use of a priori knowledge makes the observer maneuver design more realistic. The maneuver recommendation process uses the imitation of competing constraints. In this paper, an approach called programming by demonstration is used to recommend a maneuver to the observer. Simulations show that a programming by demonstration framework manages the constraints successfully and generates realistic observer maneuvers.

Key Words: *Bearing only tracking, observer maneuver recommendation, Gaussian mixture regression, programming by demonstration, target tracking*

1. Introduction

Bearing (angle) only tracking (BOT) techniques are used in a variety of theoretical and practical applications [1-5]. In the ocean environment, the BOT problem refers to the methods that are applied to determine the trajectory of the target based only on the time series of noisy bearing data. This process is referred to as passive tracking. In many applications, it is assumed that the target moves along a straight line with a nearly constant velocity [3]. In the BOT problem, even if the measurements are noise free, the observer must maneuver to guarantee the uniqueness of the solution; otherwise the problem remains unobservable [6-12].

Several papers have tried to determine the observer course sequence (maneuver) with nonlinear optimization techniques that minimize the accuracy criteria on the basis of the Fisher information matrix (FIM). Fawcett [6] supposes that the observer has a fixed course on the first leg and determines the observer course on

*Corresponding author: Informatics Institute, Department of Satellite Communication and Remote Sensing, İstanbul Technical University, İstanbul-TURKEY

the second leg according to the target range accuracy criterion. The method in [6] uses a variance of the error in the Cramer-Rao lower bound (CRLB) to perform a better maneuver from the several maneuvers available, and in the paper it is assumed that the BOT geometry is known. However, in the BOT geometry, until a solution of the BOT problem is obtained, the target range and target course cannot be known. On the other hand, some researchers have claimed that observer path optimization can be done without prior knowledge; however, this hypothesis is debatable [8]. Quach and Farooq [7] used the CRLB to determine the best observer course sequence. Le Cadre and Jauffret [4] carried out a FIM determinant analysis to optimize the observer path without any prior knowledge about the source path. In [8], Koteswara Rao proposed a method to understand the BOT geometry and to determine the approximate maneuver.

In reality, the observer maneuver is to be selected using the available bearing only information [8]. To perform the best observer maneuver adaptively, this paper proposes a new method by combining the incoming bearing information with previously defined reference BOT geometries in a general optimization framework. Reference BOT geometries are a set of BOT scenarios and serve as a priori knowledge for future calculations. In this study, a programming by demonstration (PbD) approach is used to recommend the best observer maneuver [13-15]. The key concept of this method is to determine a metric for imitation performance. In ideal BOT geometry, the target and observer move toward each other and the observer performs an 'S' shaped maneuver perfectly. Here, our aim is to perform the 'S' shaped observer maneuver to solve the BOT problem, even if the BOT geometry is not ideal. To achieve this, characteristic features of the observer maneuver in the ideal BOT scenario are defined and saved, and these features are then imitated to make the observer maneuver as in the ideal BOT scenario. Once a metric is defined, it is then possible to find an optimal controller for imitation to minimize this metric [13]. The metric works as a cost function for reproduction of the observer maneuver. The proposed method defines an adaptive process because the observer changes its course according to the incoming bearing measurement.

The rest of the paper is organized as follows. The formulation of the BOT problem is given in Section 2. Section 3 is devoted to the basic imitated features of the reference BOT scenarios. The general architecture of the proposed PbD framework is then presented in Section 4 in detail. The tracking algorithms used in BOT are explained in the same section Finally, the experimental results and concluding remarks are given in Sections 5 and 6, respectively.

2. Bearing only target tracking from a single moving platform

In this section, the definition and formulation of the BOT problem are given. The objective of BOT is to estimate the position and velocity of a moving target via noise corrupted bearing measurements. Figure 1 shows the geometry of the BOT problem, where V_T is the target speed and V_O is the observer speed. R shows the range between the target and observer The observer and target are supposed to lie on the same horizontal plane. For the single sensor case, measurement is taken from a single moving observer, which is the observer in our case.

The target is located at coordinates $(x^t y^t)$ and moves with a nearly constant velocity $(\dot{x}^t \dot{y}^t)$, and thus the state vector is:

$$\mathbf{x}^t = [x^t y^t \dot{x}^t \dot{y}^t]^T \tag{1}$$

The observer state is similarly defined as in [9]:

$\mathbf{x}^o=[x^o y^o \dot{x}^o \dot{y}^o]^T$, where the speed vector is almost constant. The relative state vector is:

$$\mathbf{x}=\mathbf{x}^t-\mathbf{x}^o=[x \quad y \quad \dot{x} \quad \dot{y}]^T \tag{2}$$

The discrete time state equation of the BOT problem is given as:

$$\mathbf{x}_{k+1}=\mathbf{F}_k x_k+\mathbf{\Gamma}_k \nu_k-\mathbf{U}_{k,k+1} \tag{3}$$

$$\mathbf{F}_k=\begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{\Gamma}_k=\begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix} \tag{4}$$

where T is the sampling period, ν_k is the 2 × 1 independent and identically distributed process noise vector with $\nu_k \sim \mathcal{N}(0,Q_k)$ and the process noise covariance matrix Q is $\sigma_a I_2$, where σ_a is a scalar and I_2 is a 2 × 2 identity matrix. Note that $\mathcal{N}(a,b)$ is a Gaussian density with mean (a) and covariance (b) The k value is the time index.

$$\mathbf{U}_{k,k+1}=\begin{bmatrix} u_{k_1} \\ u_{k_2} \\ u_{k_3} \\ u_{k_4} \end{bmatrix}=\begin{bmatrix} x^0_{k+1}-x^0_k-T\dot{x}^o_k \\ y^0_{k+1}-y^0_k-T\dot{y}^0_k \\ \dot{x}^0_{k+1}-\dot{x}^0_k \\ \dot{y}^0_{k+1}-\dot{y}^0_k \end{bmatrix} \tag{5}$$

is a vector of deterministic inputs that explains the effects of the observer accelerations [9]. The $U_{k,k+1}$ vector is known at every instant of time since the observer state vector x^o_k is given by an on-board inertial navigation system aided by a global positioning system [9] The measurement at time k is the observation angle (θ_k), as shown in Figure 1, between true north and line of sight. Therefore, the observation equation is:

$$\theta_k=\tan^{-1} {y_k}/{x_k}+w_k \tag{6}$$

where w_k is a zero mean independent white Gaussian noise with variance σ_θ.

Given a sequence of observation angles θ_k, for $k=1,2,...,n$ defined by the observation equation in Eq. (6) and given the target motion model defined in the state transition equation in Eq. (3), the BOT problem is to estimate the state vector x_k. The observation equation in Eq. (6) makes the bearing only problem nonlinearly related to the state vector [9].

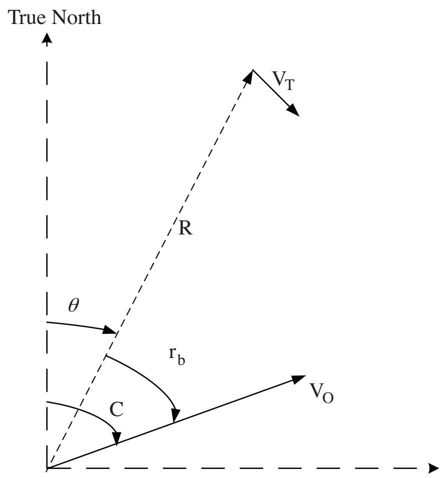

Figure 1. Geometry of BOT.

2.1. Determination of imitated features of ideal BOT scenario

The observer maneuver is a fundamental requirement to solve the BOT problem. In addition, the more the observer performs maneuver successfully, the less time a tracking algorithm needs to solve the BOT problem. Therefore, to achieve the best observer maneuver, the observer tries to imitate the salient features of the observer maneuver of the ideal BOT scenario.

Choice of the imitated features of the observer maneuver of ideal scenarios constitutes the basis for the PbD approach. The PbD framework tries to imitate the features of the observer maneuver and offers the best observer maneuver to the observer, even if the BOT geometry is far from the ideal case. In fact, the observer maneuver is the successive paths that the observer follows on a specific course for every sampling period T.

Mathematical derivation of the observer maneuver is presented in the study of Koteswara Rao [8]. His paper proposes the 'S' maneuver as an observer maneuver in the result of the mathematical analysis. On the other hand, 2 simple principles can summarize the maneuver recommendations in BOT [7,11,18]. These are:

1) The observer has to proceed to the target.

2) The observer maneuver has to maximize the bearing rate throughout the scenario and/or to maximize the difference of the bearing rate between legs. Rule 1 and Rule 2 define the 'S' maneuver indirectly. Therefore, we choose the 'S' maneuver and its variants as observer maneuvers.

After the determination of the observer maneuver shape, imitated features of the observer maneuver of ideal scenarios will now be described. The observer course is an imitated feature because the observer must move as in the ideal BOT scenarios. Moreover, the observer must maneuver to guarantee the uniqueness of the solution [5,7,11,18]. Relative bearing r_b of the observer is the second feature to be imitated. The definition of the relative bearing is given as:

$$r_b = C - \theta \tag{7}$$

where r_b is the relative bearing, C is the observer course, and θ is the bearing measurement, as shown in Figure 1. All of these variables are angles. The importance of relative bearing r_b is that it depends on the bearing measurement adding adaptivity to the maneuvering process.

2.2. Proposed observer path design system architecture

The proposed system architecture of the adaptive observer path design by imitation of competing constraints is explained in this section. Figure 2 summarizes the general operation of the proposed system. In Section 3, by means of the mathematical foundations of the observer maneuver, the constraints of the maneuver recommendation process are determined. These constraints are the observer course and observer relative bearing. In the demonstration section the statistical characteristics, which are the mean and the covariance of the constraints are calculated using the reference BOT scenarios. Until the reproduction section, everything is offline and performed once. After this step, as bearing measurements are coming, the reproduction process generates course offerings for the observer using the statistical characteristics of the constraints, as shown in the reproduction box in Figure 2. The reproduction section is an adaptive process, replying to the incoming bearing measurements by recommending the course to the observer and consequently, the observer performs maneuvers. In the last step, the tracker uses the measurements and observer position and speed to estimate the target range simultaneously. The observer position and speed are used to calculate the **U** vector in Eq. (5)

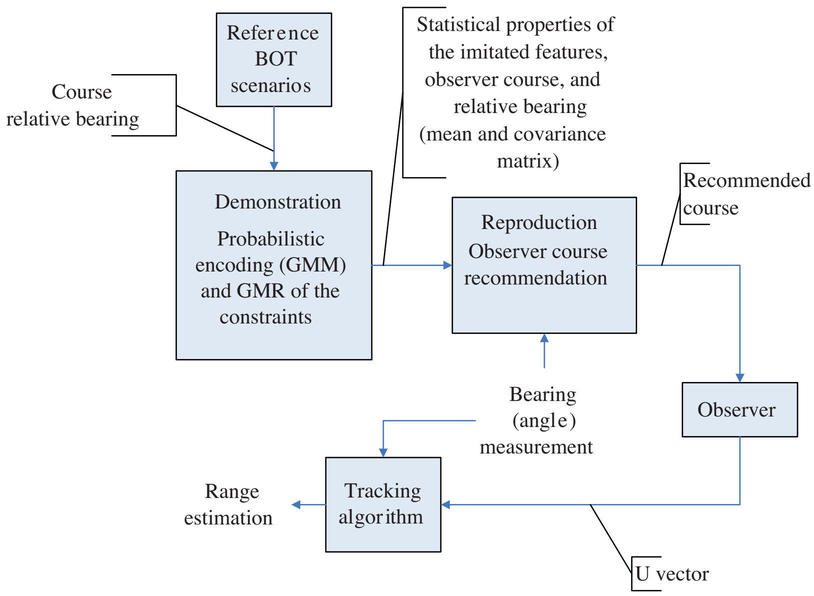

Figure 2. Proposed PbD system architecture.

2.3. Demonstration

The aim of this step is to generate the prior knowledge for the observer maneuver recommendation process. In [8], Koteswara Rao claimed that the observer path can be optimized if the target trajectory is known. Koteswara Rao's approach uses target trajectory as prior knowledge to recommend the maneuver to the observer. However, this approach is not realistic. In this study, prior knowledge is acquired from the reference BOT geometries and is stated as the mean and the covariance of the constraints for every time index. Inputs of the demonstration phase, which are the observer course and relative bearing of the reference BOT geometries, are encoded in a Gaussian mixture model (GMM) before a Gaussian mixture regression (GMR) process. Reference BOT geometries, whose observer course and observer relative bearing features are encoded in the GMM, contain the 'S' observer maneuver variants. Thus, prior knowledge has features of ideal BOT geometries and the reproduction phase tries to imitate the ideal BOT geometry features. Note that in addition to the observer maneuver shape the position and course of the target is also important for ideal BOT geometry. In ideal BOT geometry, the target and observer approach each other and the observer performs 'S' maneuvers.

Generally, a GMR approach considers multiple constraints based on the principle of the Gaussian conditioning theorem and the linear combination properties of Gaussian distributions [13]. A set of relative bearing and course data indexed with time is collected in light of the optimal maneuvering techniques, which are defined in Rule 1 and Rule 2 in Section 3 [6,8,11,12]. These data are inputs ξ^I of the GMR process. Simply, the GMR process produces the estimates of the conditional expectations of outputs ξ^O in response to given inputs ξ^I [13,14]. In this study, the ξ^O output of the GMR is the generalized relative bearing $\hat{r}_b$ the generalized observer course $\hat{\mathrm{C}}$, and their generalized covariance matrixes in response to given time index t. The generalized course and relative bearing are retrieved by calculating the expected value (mean) of $E\left[P(\xi^O|\xi^I)\right]$ and the bounds of the generalized course and relative bearing are represented by $cov(P\left(\xi^O\,|\xi^I\right))$ [13].

Before the GMR process, the dataset is encoded in a GMM learned through an expectation-maximization algorithm. The number of Gaussians in the GMM, which is stated as K in the Appendix, determines the compromise for the GMR between having an accurate estimation of the response and having a smooth response, known as the bias-variance trade-off [1314]. In our case, a smooth response is favorable to perform maneuvers realistically.

At the end of the demonstration step parameters of a Gaussian distribution, the mean and covariance of the observer course and relative bearing are obtained for every time index, and thus the prior knowledge is calculated. In the next section, reproduction approximates these generalized values to get an ideal maneuver.

2.4. Reproduction

The aim now is to find an optimal controller for the observer course, by taking C constraints and r_b constraints into account.

Let $\dot{r}_b$ and $\dot{C}$ be the candidate angular velocities of the relative bearing and observer course. Next, let $\hat{r_{b_t}}$ and $\hat{C}_t$ denote, respectively, the generalized relative bearing and generalized observer course at time step t, and let $\hat{\Sigma}_{r_{b_t}}$ and $\hat{\Sigma}_{C_t}$ be the covariance matrixes of the generalized relative bearing and generalized observer course. The changes in relative bearing $\dot{r}_{b_t}$ and in course $\dot{C}_t$ are calculated by the outputs of the demonstration phase, as shown in Eqs. (8) and (9).

$$\dot{r}_{b_t} = {}^{1}/_{\Delta t} \left[\hat{r_{b_t}} - r_{b_{t-\Delta t}} \right] \tag{8}$$

$$\dot{C}_t = {}^{1}/_{\Delta t} \left[\hat{C}_t - C_{t-\Delta t} \right] \tag{9}$$

Using Gaussian product properties, the minimum change value in the relative bearing is then calculated as:

$$\Delta r_{b_t} = \left(\Sigma_{\hat{r}_b}^{-1} + \Sigma_{\hat{C}}^{-1} \right)^{-1} \left(\Sigma_{\hat{r}_b}^{-1} \dot{r}_{b_t} + \Sigma_{\hat{C}}^{-1} \dot{C}_t \right) \tag{10}$$

The Gaussian product property is equivalent to the Lagrange multiplier method [14]. Eq. (10) represents the cost function that is used in the reproduction procedure. Therefore, the cost function (metric of imitation) is described to minimize the relative bearing change and course change, and then, by use of the cost function, the reproduction phase imitates the outputs of the demonstration phase with minimum difference.

After that, adding the change in relative bearing Δr_{b_t} to the current relative bearing, the new relative bearing is obtained in:

$$r_b = r_{b_{t-\Delta t}} + \Delta t \Delta r_{b_t} \tag{11}$$

Using Eq. (7), a new observer course is calculated by adding the new relative bearing to the incoming bearing measurement. At the end of the reproduction phase, a new path recommendation is offered for maneuvering by calculating observer course C The observer performs the maneuver by moving in the direction of the offered course C. Indirectly, tracking algorithms use the observer position and speed, which are calculated by use of the course value C, as a control variable defined as U in Eq. (5) In conclusion, tracking algorithms use the output of the maneuver recommendation procedure and solve the BOT problem effectively. Figure 2 shows the flow of the proposed procedure with the input and outputs of each substructure.

The controller should use the definition of relative bearing r_b in Eq. (7) to associate the measurement data with the course and relative bearing. This relation explains the competing nature of the constraints. When the bearing measurement arrives at the observer, by putting the generalized course (mean of course) of the demonstration phase and the bearing value into Eq. (7), the relative bearing value can be calculated.

However, the relative bearing is not imitated in this case. In the same manner, if we put the generalized relative bearing (mean of relative bearing) of the demonstration phase and measurement into Eq. (7), this time the course value is obtained and is not imitated by the reproduction phase.

The reproduction phase operates as an optimization procedure, determining the new relative bearing and course values by considering the competing nature of the constraints. Table 1 shows the reproduction procedure in detail.

Table 1. Reproduction procedure.

Start the iteration with relative bearing $r_{b_0}=\hat{r}_{b_0}$ and observer course$C_0=r_{b_0}+\theta_0$, where θ_0 is the first bearing measurement. for t = $\Delta t \rightarrow T$
1. Compute the expected angular velocity of relative bearing, which is approximated by Euler numerical differentiation for the constraint $\dot{r}_b$, as in Eq. (8). $\dot{r}_{b_t}={}^1\!/_{\Delta t}\left[\hat{r_{b_t}}-r_{b_{t-\Delta t}}\right]$ $\hat{r_{b_t}}$ and $\hat{\Sigma}_{r_{b_t}}$, which comes from the demonstration section.
2. Compute the expected angular course velocity change, as in Eq. (9). $\dot{C}_t={}^1\!/_{\Delta t}\left[\hat{C}_t-C_{t-\Delta t}\right]$, $\hat{C}_t$ and $\hat{\Sigma}_{C_t}$, which comes from the demonstration section.
3. Compute the new r_b by considering the Gaussian product property, which denotes the joint probability of constraints r_b and C, as in Eqs. (10) and (11). $\Delta r_{b_t}=\left(\Sigma_{\hat{r}_b}^{-1}+\Sigma_{\hat{C}^{-1}}\right)^{-1}(\Sigma_{\hat{r}_b}^{-1}\dot{r}_{b_t}+\Sigma_{\hat{C}}^{-1}\dot{C}_t)$ $r_b=r_{b_{t-\Delta t}}+\Delta t\Delta r_{b_t}$
4. The new course of the observer is defined, as in Eq. (7). $C=r_b+\theta$ End of loop t

2.5. Algorithms

While the PbD framework offers the best maneuver, on the other hand, a tracking algorithm tries to get the range of the target. In this study, we used 2 different algorithms to observe the performance of the PbD based maneuver recommendation process: the regularized particle filter (RPF) was used as a sequential Monte Carlo method and the modified polar extended Kalman filter (MPEKF) was used as an analytic approximation method [9,16,17] Both of the algorithms constitute the posterior density function to estimate state vector x [9].

3. Experimental results

Two specific applications of our general framework are presented:

Case 1. A BOT scenario is run to see the effect of the constraints over the recommended maneuver

Case 2. Possible BOT scenarios are run and the results of the tracking algorithms are presented to evaluate the success of the maneuver recommendation framework.

3.1. Demonstration

The demonstration phase is the first step of the proposed PbD based maneuver recommendation system, as shown in Figure 2 The 'S' maneuver and its variants are preferred as reference maneuvers due to the reasons mentioned in Section 3. The observer moves forward at a fixed speed of 8 m/s, with an initial course of 0°. The observer then performs different 'S' maneuvers, as shown in Figure 3. In all of the scenarios, the observer heads straight toward the target for a while. The target speed is 2 m/s and it is 10,000 m away from the observer moving south. Three scenarios are run for 10 min; measurements are taken every 20 s.

For every maneuver, the observer course and relative bearing are recorded. After that, the recorded data are encoded by the GMM. The number of Gaussians K, is determined based on the experiments and found as 3

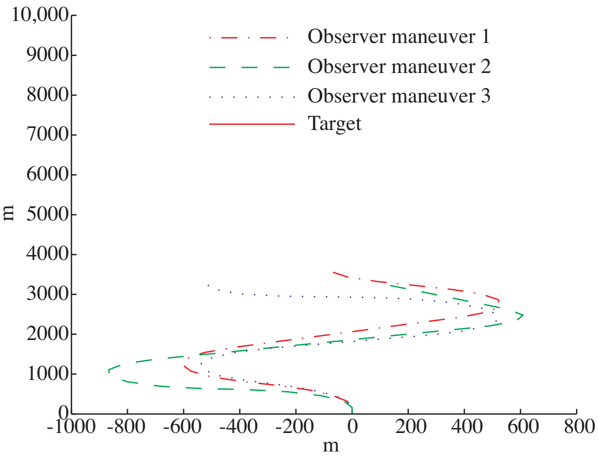

Figure 3. Reference observer trajectories.

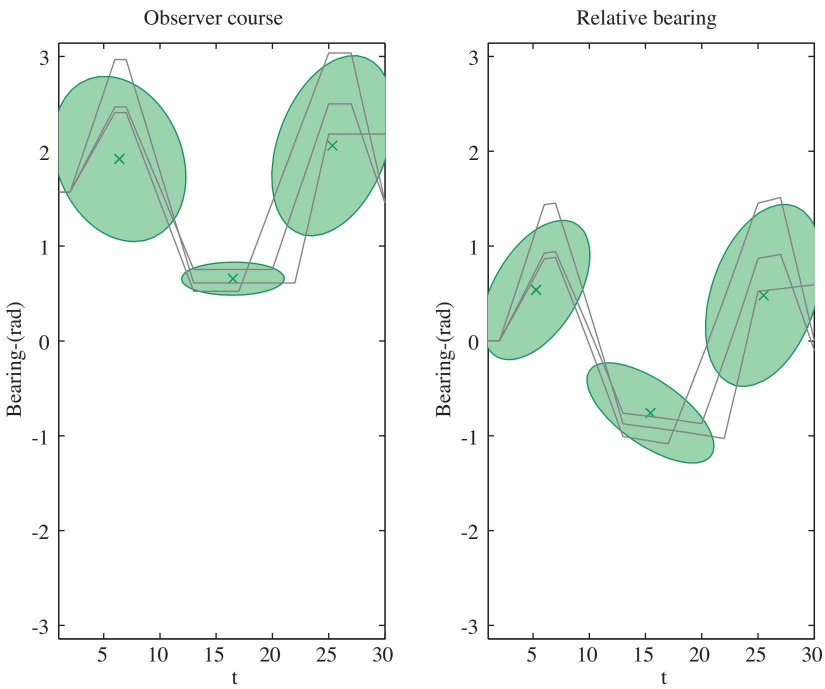

Figure 4. Encoded competing constraints by means of the GMM.

Figure 4 shows the automatic extraction of the constraints, observer course, and relative bearings from the 'S' maneuver variants. At the end of the demonstration phase, a single Gaussian is generated for every time step for the observer course and relative bearing variables by use of the GMR method.

3.2. Case 1

To better understand the effects of the constraints extracted in the demonstration phase on the recommended maneuver, let us examine the following scenario, where the target speed is 5 m/s and the target is 10,000 m away from the origin, and the observer speed is 15 m/s. While the target moves in moves in an eastward direction the observer initiates the maneuvering procedure at the point of (4000, 4000). For the maneuver recommendations, the scenario is run 3 times for different constraint considerations with the same initial conditions and same target trajectory. These constraint considerations are the only course constraint, the only relative bearing constraint, and both of them, respectively. If the PbD framework considers only one constraint, it can result in unrealistic observer maneuvers, as shown in Figure 5. However, when the PbD framework considers the 2 constraints, it produces the realistic observer maneuver in Figure 5

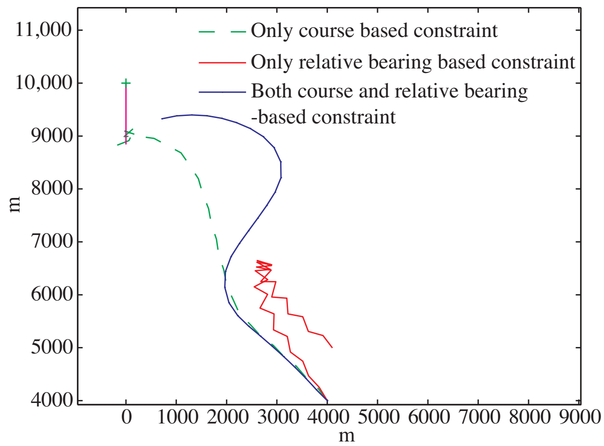

Figure 5. Three different observer maneuvers according to the different constraint considerations.

The result obtained with the 2 constraints in Figure 5 shows that imitation of the competing constraints is handled successfully. Moreover, the adaptiveness of the proposed system architecture is presented in this scenario. The PbD based course recommendation process considers the bearing measurements and controls the observer motion using the bearing measurement to offer realistic observer maneuvers

3.3. Case 2

The success of the maneuver recommendation process can be examined by considering the results of the tracking algorithms. Sixteen scenarios with different initial ranges and line of sights are considered. Possible BOT geometries are used to run the tracking filters and PbD based maneuver recommendation procedure simultaneously; therefore, the performance of the tracking algorithms indicates the feasibility of the observer maneuver recommendation process. Generally, the performance of the tracking algorithms in the BOT problem is evaluated in case the target approaches the observer. However, our scenarios also contain the going away case due to the fact that the observer can catch the target from any line of sight in reality. After the demonstration phase, 16 different scenarios for 2 different tracking filters are performed to estimate the range. Tables 2-5 depict the specifications and convergence times of the scenarios.

The scenarios are divided into 2 groups according to the observer speed. Target speed is 5 m/s for both of the scenario groups. In the first scenario group (Table 2) the observer speed is 10 m/s, while in the second scenario group (Table 4), the observer speed is 15 m/s. Observer speed V_o clearly affects the bearing rate difference [8]; therefore, the high observer speed produces a high bearing rate difference and makes the range more observable.

Table 2. The initial observer positions and target and observer speeds of the first group of scenarios.

	V_T m/s	V_O m/s	Initial observer position, m	Starting point number
Scenario 1	5	10	(4000, 4000)	1
Scenario 2	5	10	(8000, 8000)	2
Scenario 3	5	10	(–4000, 4000)	3
Scenario 4	5	10	(–8000, 8000)	4
Scenario 5	5	10	(–4000, –4000)	5
Scenario 6	5	10	(–8000, –8000)	6
Scenario 7	5	10	(4000, –4000)	7
Scenario 8	5	10	(8000, –8000)	8

For all of the scenarios, both the course and relative bearing constraints are considered to approximate an optimal maneuver of the observer. In Figure 6, the starting points of the observer, indicated as circles, are spread on 4 quadrants and the target, indicated as '+', is located at the origin and moves eastward. For the first group of scenarios the PbD framework generates the maneuvers given in Figure 6.

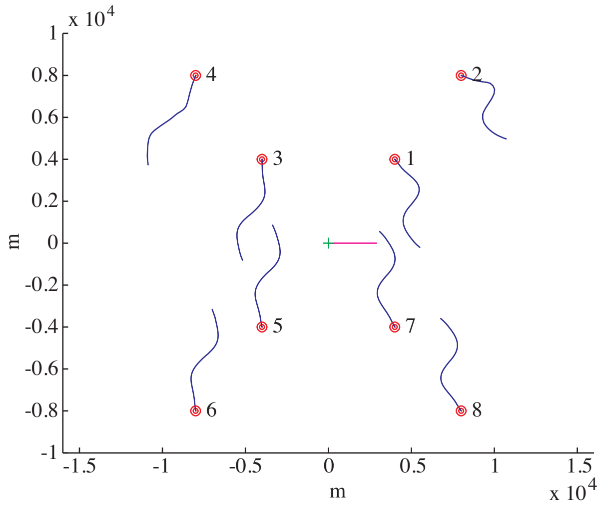

Figure 6. Recommended maneuvers for the first group of scenarios.

Simultaneously, target range estimation is performed by the RPF and MPEKF for the first group of scenarios. Table 3 indicates the observer convergence time of both tracking filters. When the range estimation error of the tracking filters goes down to 600 m, it is accepted that the filters converge to the right trajectory. Notice that measurements come every 20 s; for example, the time value 19 indicates 380 s from the beginning of the scenario. All of the scenarios take 10 minutes.

Table 3. Number of the convergence time steps of the first group of scenarios.

Scenario name	1	2	3	4	5	6	7	8
RPF	20	19	-	-	21	22	13	19
MPEKF	19	22	-	26	20	22	10	11

After carrying out of the first group of scenarios for 100 Monte Carlo runs, it is seen that the MPEKF convergence to the true range is better than that of the RPF, as shown in Table 3. Although the performance of the RPF is acceptable in most of the first group of scenarios, it generates divergent tracks for scenarios 3 and 4. For scenario 4, the RPF error is especially high, while the MPEKF is generating errors lower than the convergence border. The superiority of the MPEKF over the RPF is clearly shown. By considering the result of the tracking filters, the PbD based maneuver recommendation can cope with competing constraints and offers realistic observer maneuvers.

Table 4. The initial observer positions and target and observer speeds of the second group of scenarios.

	V_T m/s	V_O m/s	Initial observer position, m	Starting point number
Scenario 9	5	15	(4000, 4000)	9
Scenario 10	5	15	(8000, 8000)	10
Scenario 11	5	15	(–4000, 4000)	11
Scenario 12	5	15	(–8000, 8000)	12
Scenario 13	5	15	(–4000, –4000)	13
Scenario 14	5	15	(–8000, –8000)	14
Scenario 15	5	15	(4000, –4000)	15
Scenario 16	5	15	(8000, –8000)	16

For the second group of scenarios, the only difference is the observer speed, and this is switched to 15 m/s. The second group of scenarios in Table 4 is executed and the recommended maneuvers are presented in Figure 7. The performance of the range estimation process is shown in Table 5.

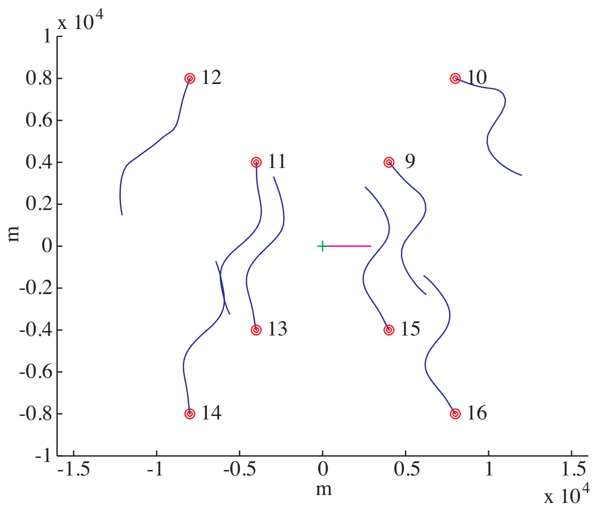

Figure 7. Recommended maneuvers for the second group of scenarios.

The effect of the increment in observer speed can clearly be seen in Table 5

Table 5. Number of the convergence time steps of the second group of scenarios.

Scenario name	9	10	11	12	13	14	15	16
RPF	18	18	-	-	20	21	11	15
MPEKF	19	14	-	25	18	21	10	11

All of the range estimation errors fall in the convergence border except for scenario 11 for the MPEKF, and the RPF diverges in scenarios 11 and 12. The most important result of the second group of scenarios is that the tracking algorithms converge faster due to the increase in the observer speed

In the recommended maneuvers of scenarios 4 and 11, the tracking algorithms do not generate satisfactory results. In these geometries, the target goes away from the observer and the maneuver recommendation process neither performs the 'S' maneuver nor approaches the target. In fact, Rule 1 and Rule 2 of Section 3 are not applied to the maneuver recommendation process. Generally in the going away target case it is a difficult problem to estimate the range. Moreover, the case encountered in scenarios 4 and 11 is very far from the ideal BOT geometry.

As a result, the simulations of the scenarios indicate that the PbD based maneuver recommendation framework offers reasonable observer maneuvers for all kinds of scenarios, including the going away target trajectory. Despite a long distance between the observer and the target, tracking algorithms usually generate satisfactory range estimation results.

As for the performance of the tracking algorithms, the MPEKF operates in the polar coordinates and describes the observation equation as linear [16]. As shown in convergence Tables 3 and 5 for the 16 tracking scenarios, the MPEKF diverges for a few scenarios. The MPEKF runs faster than the RPF and generates accurate results. Moreover, its computational burden is extremely low with respect to the RPF. In this study, the performance of the RPF achieves a comparable performance with the MPEKF. However, as shown in Tables 3 and 5, the RPF diverges for all of the second quadrant scenarios, while the MPEKF produces convergent results for scenarios 4 and 12. BOT geometry can explain the divergent results of the RPF. For long range scenarios, as seen in the second quadrant scenarios where the observer is moving away from the target, the degree of linearity is high and the posterior density of the state variable becomes more Gaussian [9]. Thus, the posterior density produced by the RPF does not fit into a Gaussian distribution. In linear Gaussian cases, the Kalman filter based tracker outperforms the RPF [9] However, when the observer gets closer to the target, then the RPF performs as well as the MPEKF. As a result, the overall performance of the MPEKF is considered to be better than the RPF. In addition, the computational burden of the RPF is high and increases according to the number of the particles used to represent the posterior density [9,16].

In the demonstration phase, 3 Gaussians are used to encode the course and relative bearing data; thus, the reproduction phase produces smooth observer maneuvers. Bearing measurements can be received from every direction; these measurements, which are very different from the measurements of the demonstration phase, are tolerated by means of the smoothness behavior of the encoded data. The smoothness of the encoded data is described by the covariance matrixes. The most important result of recommending smooth maneuvers is the realistic observer maneuver. However, many maneuvering algorithms generate unrealistic observer maneuvers that are physically impossible [6, 8]. The proposed method solves this problem, as shown in Figures 6 and 7

4. Conclusion and future work

In a BOT problem, even if the bearing measurements are perfect, an observer maneuver is necessary. The performance of BOT significantly depends on the observer maneuver. Some of the existing algorithms offer unrealistic observer maneuvers and many of them do not use a priori knowledge to offer the best observer maneuver. Furthermore, some studies assume that the BOT geometry is known, although this is not a true assumption and not feasible for practical applications.

In this study, a PbD frameworkbased adaptive observer maneuver recommendation algorithm was developed. The reference BOT geometries are used as a priori knowledge. The control of the observer motion is handled by a PbD optimization framework, which combines the prior knowledge with the incoming bearing measurements. Thus, for possible BOT geometries, a PbD based observer maneuver recommendation framework generates realistic observer maneuvers, even if different ranges and different line of sights are considered. The results prove that the proposed approach works well.

In this paper, a set of scenarios was collected in the demonstration phase and maneuver constraints, which are the relative bearing and the observer course, are represented by a Gaussian distribution for every time step In the reproduction phase, an optimum controller is defined to approximate the outputs of the demonstration phase, which are the mean and covariance of the course and relative bearing. At the end of the reproduction phase, the PbD framework recommends a course to the observer. The recommended course then imposes a change on the x and y components of the speed and position of the observer. Thus, the observer maneuvers and tracking algorithms use the speed and position of the observer and bearing measurement to estimate the target range

On the other hand, different constraints can be described in addition to the course and relative bearing. For example, the bearing rate can be another parameter to be considered, but the relative bearing is an indispensable constraint due to the association of the bearing measurement to the maneuver recommendation process. Likewise, available range information and other metrics mentioned in [6,7,18] can be considered as constraints in the optimization process.

Our proposed approach presents a solution for the observer maneuver recommendation in a general framework, systematically and generically. Every box in the proposed system architecture in Figure 2 can be replaced by a more effective one. Different optimization techniques can be used, like Lagrange in the reproduction phase, and different tracking algorithms can be utilized.

Appendix

After the encoding of N data points of D dimension via K GMM components, the probability that a data point fits the GMM is defined by:

$$P(\xi) = \sum_{k=1}^{K} \pi_k N(\xi; \mu_k, \Sigma_k)$$

$$= \sum_{k=1}^{K} \pi_k \frac{1}{\sqrt{(2\pi)^D \left|\sum_k\right|}} e^{-\frac{1}{2}\left[(\xi-\pi_k)^T \sum_k^{-1} (\xi-\pi_k)\right]},$$

where π_k represents the prior probabilities and $N(\xi; \mu_k \Sigma_k)$ are Gaussian distributions defined by mean μ_k and covariance matrixes Σ_k. Input and outputs are defined separately as:

$$\mu_k = \begin{bmatrix} \mu_k^I \\ \mu_k^O \end{bmatrix}, \quad \Sigma_k = \begin{bmatrix} \Sigma_k^I & \Sigma_k^{IO} \\ \Sigma_k^{IO} & \Sigma_k^O \end{bmatrix}.$$

The expected distribution of output ξ^O, given input variable ξ^I and given a Gaussian distribution k, is represented as:

$P\left(\xi^O\,|\xi^I,k\right)\sim N(\hat{\xi}_k\hat{\Sigma}_k)$, where

$$\hat{\xi}_k=\mu_k^O+\Sigma_k^{IO}\left(\Sigma_k^I\right)^{-1}(\xi^I-\mu_k^I),$$

$$\hat{\Sigma}_k=\Sigma_k^O-\Sigma_k^{IO}\left(\Sigma_k^I\right)^{-1}\Sigma_k^{IO}.$$

Regarding the GMM, the expected distribution of ξ^O, while ξ^I is known, is estimated as:

$$P\left(\xi^O\,|\xi^I\right)\sim=\sum_{k=1}^{K}h_kN(\hat{\xi}_k,\hat{\Sigma}_k)$$

where $h_k = P\left(k|\xi^I\right)$ is the probability that ξ^I is drawn from the Gaussian distribution k.

$$h_k=\frac{\pi_kN(\xi^I;\mu_k^I,\Sigma_k^I)}{\sum\limits_{k=1}^{K}\pi_kN(\xi^I;\mu_k^I,\Sigma_k^I)}$$

Concerning the linear transformation property of Gaussian distribution, the conditional expectation of ξ^O, while ξ^I is given, can be extracted as a single Gaussian distribution $N(\hat{\xi}_k\hat{\Sigma}_k)$ with parameters:

$$\xi=\sum_{k=1}^{K}h_k\hat{\xi}_k\quad\hat{\Sigma}_k=\sum_{k=1}^{K}h_k^2\hat{\Sigma}_k.$$

References

[1] J.M.C. Clark, S.A. Robbiati, R.B. Vinter, “The shifted Rayleigh mixture filter for bearings-only tracking of maneuvering targets”, IEEE Transactions on Signal Processing, Vol. 55, pp. 3218-3226, 2007.

[2] Y.J. Zhang, G.Z. Xu, “Bearings-only target motion analysis via instrumental variable estimation”, IEEE Transactions on Signal Processing, Vol. 58, pp. 5523-5533, 2010.

[3] A.N. Bishop, P.N. Pathirana, “Localization of emitters via the intersection of bearing lines: a ghost elimination approach”, IEEE Transactions on Vehicular Technology, Vol. 56, pp. 3106-3110, 2007.

[4] J.P. Le Cadre, C. Jauffret, “Discrete time observability and estimability analysis for bearings only target motion analysis”, IEEE Transactions on Aerospace and Electronic Systems, Vol. 33, pp. 178-201,1997.

[5] K.C. Ho, Y.T. Chan, “An asymptotically unbiased estimator for bearings-only and Doppler-bearing target motion analysis”, IEEE Transactions on Signal Processing, Vol. 54, pp. 809-822, 2006.

[6] J.A. Fawcett, “Effect of course maneuvers on bearings-only range estimation”, IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 36, pp. 1193-1199, 1988.

[7] T. Quach, M.A. Farooq, “Simple approach to observer path design for bearing only tracking”, Proceedings of the 35th Midwest Symposium on Circuits and Systems, Vol. 2, pp. 1147-1150, 1992.

[8] S. Koteswara Rao, Comments on "Discrete-time observability and estimability analysis for bearings-only target motion analysis", IEEE Transactions on Aerospace and Electronic Systems, Vol. 34, pp. 1361-1367, 1998.

[9] B. Ristic, S. Arulampalam, N. Gordon, Beyond the Kalman Filter: Particle Filters for Tracking Applications, Boston, Artech House Radar Library, 2004.

[10] S.E. Hammel, P.T. Liu, E.J. Hilliard, K.F. Gong, "Optimal observer motion for localization with bearings measurements", Computer and Mathematics with Applications, Vol. 18, pp. 171-186, 1989.

[11] J.M. Passerieux, D. Van Cappel, "Optimal observer maneuver for bearings-only tracking", IEEE Transactions on Aerospace and Electronic Systems, Vol. 34, pp. 777-788, 1996.

[12] J.P. Helferty, D.R. Mudgett, "Optimal observer trajectories for bearings-only tracking by minimizing the trace of the Cramer-Rao bound", Proceedings of the 32nd Conference on Decision and Control, pp. 936-939, 1993.

[13] S. Calinon, F. Guenter, A. Billard, "On learning, representing and generalizing a task in a humanoid robot", IEEE Transactions on Systems, Man and Cybernetics, Part B, Vol. 37, pp. 286-298, 2007.

[14] S. Calinon, A. Billard, "Statistical learning by imitation of competing constraints in joint space and task space", RSJ Advanced Robotics, Vol. 23, pp. 2059-2076, 2009.

[15] S. Calinon, A. Billard, "A probabilistic programming by demonstration framework handling constraints in joint space and task space", Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 367-372, 2008.

[16] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking", IEEE Transactions on Signal Processing, Vol. 50, pp. 174-188, 2002.

[17] V.J. Aidala, S.E. Hammel, "Utilization of modified polar coordinates for bearings-only tracking", IEEE Transactions on Automatic Control, Vol. 28, pp. 283-294, 1983.

[18] T. Kronhamn, "Bearings-only target motion analysis based on multihypothesis Kalman filter and adaptive ownship motion control", IEEE Proceedings - Radar, Sonar and Navigation, Vol. 145, pp. 247-252, 1998.