OPTIMAL RECURSIVE DATA PROCESSING ALGORITHM USING BAYESIAN INFERENCE FOR UNDERWATER VEHICLE LOCALIZATION AND NAVIGATION SYSTEMS

JAWAHAR A*

Department of Electrical and Electronics Engineering, Sanketika Institute of Technology and Management, Visakhapatnam, India.
Email: 0891jawahar@gmail.com

ABSTRACT

Objective: In the ocean environment, two-dimensional range and bearings target motion analysis is generally used. In the underwater scenario, the active sonar, positioned on an observer, is capable of sensing the sound waves reflected from the target in water.

Methods: The sonar sensors in the water pick up the target reflected signal in the active mode. The observer is assumed to be moving in a straight line, and the target is assumed to be moving mostly in straight line with maneuver occasionally. The observer processes the measurements and estimates the target motion parameters, namely, range, bearing, course, and speed of the target. It also generates the validity of each of these parameters. Here, we try to apply Kalman filter for the sea scenario using the input estimation technique to detect target maneuver, estimate target acceleration, and correct the target state vector accordingly.

Results: There are mainly two versions of Kalman filter – A linearized Kalman filter, in which polar measurements are converted into Cartesian coordinates, and the well-known extended Kalman filter (EKF).

Conclusion: Recently, Pork and Lee presented a detailed theoretical comparative study of the above two methods and stated that both the methods perform well. Here, EKF is used throughout.

Keywords: Estimator, Fire control system, Helicopter, Initial turn angle, Weapon control algorithm, Kalman filter, Splash point algorithm.

INTRODUCTION

The helicopter fire control system (HFCS) system comprises FCS and weapon control system (WCS) [1-10]. The FCS receives the input measurements, i.e., bearing and range measurements (which are corrupted with noise) from the active sonar system and helicopter course and speed from helicopter sensors. The FCS estimates range (R), course (C), bearing (B), and speed (S) of the target. The system also consists of simulator, which simulates active sonar. This simulator generates range, bearing (corrupted with Gaussian noise) and platform course and speed along with the time at which sample is generated.

FCS consists of contact motion analysis (CMA). As sonar is not able to give depth of the target, it is assumed that target and active sonar are in the same plane [11-20]. This system uses adaptive Kalman filter to track the target. The output of Kalman filter is range, course, bearing, and speed (RCBS).

WCS consists of intercept guidance algorithm which generates initial turn angle (ITA) and initial search run and these can be fed directly to presetter. When maximum Hit probability reaches to the required level, then torpedo can be released. Finally, ITA is to be corrected as per wind velocity entry angle, etc., using splash point (SP) algorithm. The functional block diagram of HFCS is shown in Fig. 1.

HFCS SIMULATOR

The HFCS system comprises FCS and WCS. The FCS receives the input measurements, bearing and range from the sonar subsystem and helicopter course and speed from helicopter sensors. The FCS estimates range (R), course (C), bearing (B), and speed (S) of the target. The simulator of HFCS system performs the following tasks:
1. Accepts the geometry information as input
2. Simulates the helicopter motion and target motion
3. Generates range and bearing at discrete intervals
4. Induces Gaussian noise in range and bearing measurements.

Block diagram of the simulator is shown in Fig. 2.

Assumptions
Following are the assumptions made in the simulator:
1. Initially, ownship is considered to be at the origin
2. Y-axis is taken as a reference axis for measuring all angles.

Target motion parameters (RCBS) and ownship parameters (C, S) are read and used as input by the simulator. Error in bearing measurement (sigma_b) and in range measurement (sigma_r).

Ownship motion
The ownship motion is introduced as follows. The ownship is moving with a velocity V_0. X_0 is the distance of the ownship from X-coordinate. Y_0 is the distance of the ownship from Y-coordinate, and optical character recognition (OCR) is the angle making with north (Fig. 3).

$$Sin(OCR) = X_0/Y_0$$

$$Cos(OCR) = Y_0/V_0$$

For every second, change in X and Y components of ownship position is found and added to the previous X and Y components of ownship position.

For t_s = 1 second, $dX_0 = V_0*sin(OCR)*t_s$, $dY_0 = V_0*cos(OCR)*t_s$

where

dX_0 is change in X-component of ownship position in 1 second,

dY_0 is change in Y-component of ownship position in 1 second,

Then, $X_0 = X_0+dX_0$ and $Y_0 = Y_0+dY_0$.

Target motion
From input, bearing initial position of the target is known.
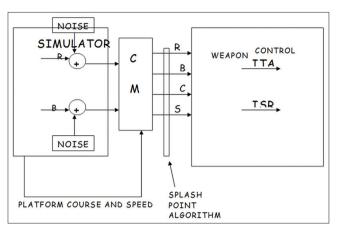
x_t = range*sin (bearing),
y_t = range*cos (bearing).

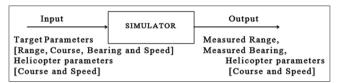**Fig. 1: Functional block diagram of helicopter fire control system**

Fig. 2: Block diagram

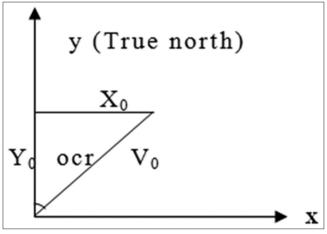**Fig. 3: Target observer geometry**

where (x_t, y_t) is target position with respect to ownship as the origin.

For every 1-second change, x_t and y_t are calculated and added to the previous position.

t_s = 1 second

$dx_t = V_t*Sin(tcr)*t_s$

$dy_t = V_t*Cos(tcr)*t_s$

Then,

$x_t = x_t+dx_t$

$y_t = y_t+dy_t$

Where,

dx_t is the change in X-component of target position in a second,

dy_t is the change in Y-component of target position in a second,

V_t is target velocity,

tcr is target course with respect to true North.

MATHEMATICAL MODELING OF NOISE AND MEASUREMENTS

The noise in the bearing and range is assumed to be additive in nature and follows normal distribution. Using PI generator, pseudo-random number is generated [21-30]. Using central limit theorem, pseudo-random numbers are used to derive normally distributed noise sequence.

PI random number generator
The PI random number generator will return a sequence of uniformly distributed numbers between zero and unity. The function utilizes the equation:

$X = (seed+3.14159265358979)^5$

To produce a sequence of random numbers, the seed is added to the value of PI = 3.1415926358979 and the fifth power is taken. The fractional part is termed as a random number. It is also used as a seed for the generation of next random number. The seed is to be initialized at the start of the program.

Evaluation of PI random number generator
Random number generators are sometimes called pseudo-random number generators to emphasize the fact that they are not truly random.

As the random number sequence follow uniform distribution, the mean of the random number sequence is around 0.5, and standard deviation (SD) of the random number sequence is around 0.2887.

It is observed that the mean of the random number sequence generated using PI generator ranges from 0.44 to 0.55 and S.D values from 0.25 to 0.32.

Gaussian random noise generation
The measurement contains noise generated by several noise sources. According to the central limit theorem, the sum of noises of any density function leads to Gaussian density function. Hence, it is assumed that the noise in the measurements is of Gaussian.

Gaussian random noise can be obtained by summing 12 uniformly distributed random numbers as follows:

Gaussian random noise = (sum-6.0)*Sigma + Mean   (2.1)

Where,

Sum = Sum of 12 uniformly distributed random numbers,

Sigma = Desired SD,

Mean = Desired mean, here, it is zero.

Noises in the bearing and range measurements are generated with sigma_b and sigma_r SDs, and these are added to the true bearing and true range to get measurements with noise.

Measured bearing = True bearing + noise_in_b.

Where, true bearing = $\tan^{-1}([x_t-X_0]/[y_t-Y_0])$.

Measured range = True range + noise_in_r.

Where, true range = $\sqrt{(x_t - X_0)^2 + (y_t - Y_0)^2}$.

Output
At discrete time intervals, following outputs are sampled.
1. True range
2. Measured range
3. True bearing
4. Measured bearing
5. Ownship position
6. Target position.

Simulation results

The positions of observer and target are updated at one-second interval time. However, these values are tapped and fed to target motion analysis (TMA) algorithm only at those time instants when the active sonar measurements are available. The functioning of active sonar has been simulated. Here, time required for the transmitted pulse to come back has been computed using the sound velocity (in underwater 1500 m/s) and current range of the target. The observer is assumed to be at the origin initially.

CMA TECHNIQUES

Kalman filter

In the ocean environment, two-dimensional range and bearings TMA is generally used. In the underwater scenario, the active sonar, positioned on an observer, is capable of sensing the sound waves reflected from the target in water. The sonar sensors in the water pick up the target reflected signal in the active mode [15-25]. The observer is assumed to be moving in a straight line, and the target is assumed to be moving mostly in straight line with maneuver occasionally. The observer processes the measurements and estimates the target motion parameters, namely, range, bearing, course, and speed of the target. It also generates the validity of each of these parameters. Here, we try to apply Kalman filter for the sea scenario using the input estimation technique to detect target maneuver, estimate target acceleration, and correct the target state vector accordingly.

There are mainly two versions of Kalman filter – A linearized Kalman filter, in which polar measurements are converted into Cartesian coordinates, and the well-known extended Kalman filter (EKF). Recently, Pork and Lee presented a detailed theoretical comparative study of the above two methods and stated that both the methods perform well. Here, EKF is used throughout.

The detection of target maneuver is carried out as follows. In this process, it is assumed that the estimator EKF is of high quality in the sense that solution is possible for all scenarios including all quadrants (several geometries are tested using EKF, and the solution is invariably obtained). It is also assumed that the solution diverges only when target maneuvers. When the target is not maneuvering, it is observed from many geometries that the bearing residuals of EKF are almost zero and their small scatter around the zero-bearing line is the random noise. It is also noted that the bearing residuals are not close to zero when the target is maneuvering. It is very difficult to confirm whether the target has maneuvered or not just by visual inspection of the bearing residual plot, due to the corruption of the bearing measurement with random noise [31-40]. Hence, zero mean Chi-square distributed random sequence residuals of the non-maneuvering model, in sliding window format, are used for the detection of target maneuver. Target maneuver is declared when the normalized squared innovations exceed the threshold. At the same time, using these innovations of the Kalman filter, the acceleration input is estimated and used to correct the state estimate. During the window period, the acceleration input is assumed to be constant. This procedure is called input estimation. Here, we try to extend the input estimation technique being used for in-air applications to on-sea water applications.

Target motion parameters

Let the target state vector be $X_s(k)$,

$$\text{Where, } X_s(k) = \left[\dot{x}(k) \quad \dot{y}(k) \quad R_x(k) \quad R_y(k)\right]^T \tag{3.1}$$

Where, $\dot{x}(k)$ and $\dot{y}(k)$ are target velocity components, and $R_x(k)$ and $R_y(k)$ are range components, respectively. For the purpose of introducing concepts, to start with the target is assumed to be non-maneuvering. The target state dynamic equation is given by:

$$X_s(k+1) = \Phi(k+1/k) X_s(k)+b(k+1)+\omega(k)$$

Where, $\omega(k)$ is zero mean Gaussian plant noise. $\Phi(k + 1/k)$ and $b(k + 1)$ are transient matrix and the deterministic vector, respectively. These are given by:

$$\Phi(k+1/k)=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ t & 0 & 1 & 0 \\ 0 & t & 0 & 1 \end{bmatrix} \tag{3.3}$$

Where, t is sample time between measurements, and

$$b(k+1)=\begin{bmatrix} 0 & 0 & -[X_o(k+1)+X_0(k)] & -[Y_o(k+1)+Y_o(k)] \end{bmatrix}^T \tag{3.4}$$

Where, $X_0(k)$ and $Y_0(k)$ are ownship position components, respectively. True North convention is followed for all angles to reduce mathematical complexity and easy implementation. The measurement vector Z (k) is given by:

$$Z(k)=\begin{bmatrix} B_m(k) \\ R_m(k) \end{bmatrix}$$

Where, $R_m(k)$ and $B_m(k)$ are range and bearing measurements and are given by:

$$B_m(k)=B(k)+\gamma(k)$$

$$R_m(k)=R(k)+\eta(k)$$

Where, B(k) and R(k) are actual bearing and range, respectively. These are given by:

$$B(k) = \tan^{-1}\left(\frac{R_x(k)}{R_y(k)}\right)$$

$$R(k)=\sqrt{R_x^2(k)+R_y^2(k)}$$

Where, $R_x(k)$ and $R_y(k)$ are x and y components of Range. $\eta(k)$ and $\gamma(k)$ are zero mean uncorrelated Gaussian noises in range and bearing measurements, respectively.

$$Z(k) = H(k)X_s(k)+\xi(k)$$

$$\text{Where, } H(k) = \begin{bmatrix} 0 & 0 & \dfrac{\cos B(k)}{R(k)} & \dfrac{-\sin B(k)}{R(k)} \\ 0 & 0 & \sin B(k) & \cos B(k) \end{bmatrix} \tag{3.9}$$

It is assumed that the plant and measurement noises are uncorrelated to each other.

The covariance prediction is,

$$P(k+1|k)=\phi(k+1|k)P(k|k)\phi^T(k+1|k)+Q(k+1) \tag{3.10}$$

Where, Q in the covariance prediction. The Kalman gain is,

$$G(k+1) = P(k+1|k)H^T(k+1)[r(k+1)+H(k+1)P(k+1|k)H^T(k+1)]^{-1} \tag{3.11}$$

Where, r(k+1) is input measurement error covariance matrix. The state and its covariance corrections are given by:

$$\text{State: } X(k+1/k+1) = X(k+1/k) + G(k+1)\left[Z(k+1) - \hat{Z}(k+1)\right] \tag{3.12}$$

Covariance: $P(k+1/k+1)=[I-G(k+1)\ H(k+1)]P(k+1/k)$

Whenever target maneuvers, there is a considerable change in the innovation process. This change is monitored and correspondingly state vector is corrected, during target maneuver.

TORPEDO WEAPON CONTROL

The torpedo WCS receives range, bearing, course, and speed from CMA, own ship course from gyro and torpedo speed (preset) as shown in

Fig. 4. It generates total firing angle (TFA) and run distance or distance to hit the target (eT).

Intercept guidance

This guidance steers weapon for an intercept with the target at a future point in time. Here, all the angles are considered with respect to Y-axis. The fire control triangle is solved to find out time to hit, run to hit, and weapon course to steer. The TFA of the weapon is obtained by subtracting platform course from weapon course to steer as shown in Fig. 5. Time to hit and run distance to hit (eT) are also outputs of the algorithm. It is assumed that the target maintains a fixed course and speed during the run of the weapon T.

Intercept guidance law steers weapon for an intercept with the target at a future point in time. It is assumed that the estimated range, bearing, course, and speed of the target are available for the computation of guidance law. The fire control triangle is solved to find out time to hit, run to hit, and weapon course to steer.

The derivation is simplified by considering all angles with respect to the Y-axis. The limits of the angles are 0 to 360°. The solution is obtained in terms of "course to steer," i.e. the angle of the torpedo with respect to Y-axis. The TFA of the weapon is obtained by subtracting platform course from weapon course to steer. In this process, no iterations are required as shown in the Figs. 6 and 7.

Let the

initial position of the own ship be F (X_0, Y_0)

course and speed of the own ship be Ψ and v_0

initial position of the target be A (x_t, y_t)

course and speed of the target be ϕ and V_t

initial position of the weapon be F (X_0, Y_0)

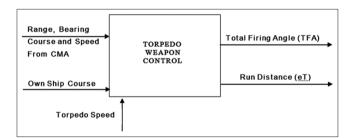course and speed of the weapon be θ and V_{t0}
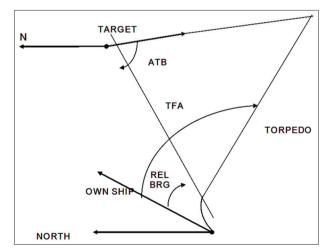

Fig. 4: Torpedo weapon control system

Fig. 5: Intercept guidance for torpedo

initial range be R and angle made by the line of sight of the target w.r.t. true north be β

From the Fig. 4, the following equation can be obtained.

In Δ^{le} ABC, BC = AB Sin (ϕ)= Vt.t.Sin (ϕ)

AC = AB Cos (ϕ)= V_t.t.Cos (ϕ)

CD = $(x_t - X_0)$= R.Sin (β)

EF = $(y_t - Y_0)$= R.Cos (β)

The run distance of the weapon is given by r = V_{t0}.T.

For simplification of computation, a factor χ is introduced as:

$\chi = V_t/V_{t0}$

$r\chi = V_t.T$

Now considering the Δ^{le} FDB, Sin(θ) = BD/BF

$\qquad$ = (BC+CD)/r

$\qquad$ = [$(x_t - X_0)$+V_t.t.Sin(ϕ)]/r

$\therefore$ Sin(θ) = [$(x_t - X_0)$+$r\chi$.Sin(ϕ)]/r

$\qquad$ = [R Sin(β)+$r\chi$.Sin(ϕ)]/r

Similarly, Cos(θ) = [$(y_t - Y_0)$+$r\chi$.Cos(ϕ)]/r

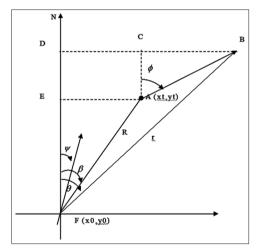$\qquad$ = [R Cos(β)+$r\chi$Cos(ϕ)]/r
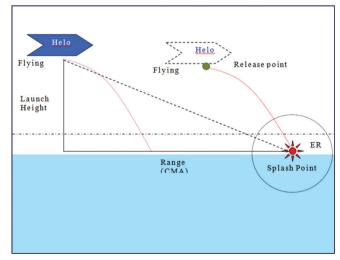

Fig. 6: Weapon control angle

Fig. 7: Scenario I

Consider $\text{Sin}^2(\theta)+\text{Cos}^2(\theta)=1$

Substituting equation (4.5) and equation (4.6) in equation (4.7), we get,

$1 = [[R\text{Sin}(\beta)+r\chi.\text{Sin}(\phi)]/r]^2+[[R\text{Cos}(\beta)+r\chi.\text{Cos}(\phi)]/r]^2$ is obtained. After straight forward simplification,

$r^2 = R^2[\text{Sin}^2(\beta)+\text{Cos}^2(\beta)]+2r\chi.[R\text{Sin}(\beta).\text{Sin}(\phi)+R\text{Cos}(\beta).\text{Cos}(\phi)]+r^2\chi^2. [\text{Sin}^2(\phi)+\text{Cos}^2(\phi)]$

$r^2 (1-\chi^2). -2 r \chi.R[\text{Sin}(\beta).\text{Sin}(\phi)+ \text{Cos}(\beta). \text{Cos}(\phi)]-R^2 = 0$

$r^2 (1-\chi^2). -2 r \chi.R.\text{Cos}(\beta-\phi)- R^2 = 0$

$r^2 (1-\chi^2). -r. 2R\chi \text{Cos}(\beta-\phi)- R^2 = 0$

Equation (8) is a quadratic equation and so r can be written as,

$$r_1= \frac{-b+\sqrt{b^2-4ac}}{2a}$$

$$r_2= \frac{-b-\sqrt{b^2-4ac}}{2a}$$

Where, $a=(1-\chi^2)$, $b=-2R\chi \text{Cos} (\beta-\phi)$ and $c=-R^2$

$r = r_1$ or r_2, whichever is positive and minimum.

$\text{Sin} (\theta) = [(x_t-X_0)+r\chi.\text{Sin}(\phi)]/r$

$\text{Cos} (\theta) = [(y_t-Y_0)+r\chi.\text{Cos}(\phi)]/r$

$\therefore \theta = \text{Tan}^{-1}[\{(x_t-X_0)+r\chi.\text{Sin}(\phi)\}/[(y_t-Y_0)+r\alpha.\text{Cos}(\phi)]$

Where, θ is weapon course to steer w.r.t true north. Here, "r" is nothing but torpedo run distance or distance to hit the target (eT). And "T" is the run time of the torpedo to hit the target.

TFA and eT are calculated by assuming that torpedo can turn instantaneously. However, in practice, torpedo can turn at a limited rate. Hence, taking this limitation, TFA and eT are corrected.

SP

SP is the point at which the torpedo touches the water after release from helicopter in flying mode as shown in Appendix Fig 1. The latitude and longitude of SP are estimated by the user and fed to the HFCS. To drop the torpedo at that SP, we need to estimate the latitude and longitude of the helicopter to release the torpedo. As this release takes place in FLY mode, it is sufficient that the torpedo can be released within the vicinity of the SP. The ideal situation is dropping the torpedo exactly at the given SP.

SCENARIOS

Case I (no-wind condition)

As per the simulation studies (air trajectory simulation with no-wind model), maximum entry range (ER) the light weight torpedo (LWT) may achieve is 300 m. Hence, in this case, the torpedo can be dropped at any point on the circle of radius ER surrounding the SP as center.

Note: for flight envelope, 500 m altitude and 60 m/s launch velocity the ER will be 300 m.

Algorithm:
Step 1: Get latitude and longitude of SP from HFCS.
Step 2: Convert latitude and longitudes to XY coordinates.
Step 3: Add ER to X coordinate (range).
Step 4: Convert XY coordinates to latitude and longitude.
Step 5: Submit release point's (RPs) latitude and longitude (step 4) to HFCS.

Case II (with-wind condition)

As per the simulation studies (air trajectory Simulation with wind model), maximum ER the LWT may achieve is <300 m. The direction of wind is always measured with respect to true north and is given by the ISP of helicopter.

As depicted in Fig. 8, the RP can be computed as:

$X = x+r*\text{Cos}(\theta)$ and $Y=y+r*\text{Sin}(\theta)$

Note: The correction in wind direction "θ" is as follows:

If θ falls in first quadrant, i.e. (0 90)θ is taken as it is.

If θ falls in second quadrant, i.e. (90 180)θ is $\prod - \theta$.

If θ falls in third quadrant, i.e. (180 270)θ is $\prod + \theta$.

If θ falls in fourth quadrant, i.e. (270 360)θ is $2\prod - \theta$.

Algorithm:
Step 1: Get latitude and longitude of SP from HFCS.
Step 2: Convert latitude and longitudes to XY coordinates.
Step 3: Get wind angle from sensor data.
Step 4: Compute RP as $X = x+ER*\text{Cos}(\theta)$ and $Y = y+ER*\text{Sin}(\theta)$.
Step 5: Convert XY coordinates to latitude and longitude.
Step 6: Submit RP's latitude and longitude (Step 5) to HFCS.

Assumptions:
1. No-wind scenario means wind direction and wind velocity are zero
2. The aircraft altitude at the time of launch is same as that of the altitude at the time of measurement
3. Firing angle is known to the HFCS operator
4. The estimated RP bears minimum 5-10% error
5. Maximum ER means, ER estimated, considering maximum launch velocity, launch height, and wind speed.

RESULTS AND OBSERVATIONS

The SP latitude-longitudes are converted to Cartesian XYs, and they are corrected with maximum ER the LWT can achieve and wind direction. The corrected XYs are again converted back to latitude-longitudes, which are considered to be the RP latitude-longitudes. It was observed that the splash and release latitude-longitudes may vary in longitude minutes, seconds, and the direction, but the longitude degrees and latitude remain the same.

For example: (no wind case: ER = 300 m; wind direction = 0°).

SP latitude-longitudes: 35° 17' 17.933" N, 120° 39' 10.443" W;
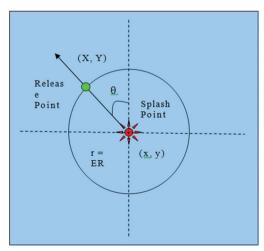
Cartesian SP: X = −2649.694, Y = −4470.911;

Fig. 8: Air trajectory simulation with wind

Comp. RP: X = X+ER = −2349.694, Y=−4470.911;

RP latitude-longitudes: 35° 17' 17.933" N, 120° 38' 56.599" W;

REFERENCES

1. Aidala VJ, Hammel SE. Utilization of modified polar coordinates for for bearings only tracking. IEEE Trans Autom Control 1983;28(3):283-94.
2. Wan EA, Van Der Merwe R. The unscented Kalman filter for non-linear estimation. In: Proceedings of Symposiums 2000 on Adaptive System for Signal Processing, Communication and Control, Canada, October; 2000. Blackman S, Popolli R. Design and Analysis of Modern Tracking Systems. Norwood, MA: Artech House; 1999.
3. Bar-Shalom Y, Fortman TE. Tracking and Data Association. San Diego, USA: Academic Press, Inc.; 1991.
4. Bar-Shalmon Y, Lee XR. Estimation and Tracking Principles, Techniques and Software. Boston, London: Artech House; 1993.
5. Ristic B, Arulampalam S, Gordon N. Beyond the Kalman Filter: Particle Filters for Tracking Applications. Boston: Artech House; 2004.
6. Simon D. Optimal State Estimation: Kalman, H and Nonlinear Approaches. Chichester: John Wiley & Sons, Inc.; 2006.
7. Liu Y, Li XR. Measure of nonlinearity for estimation. IEEE Trans Signal Proc 2015;63(9):2377-88.
8. Kumari B, Leela S, Rao K, Raju KP. Simplified target location estimation for underwater vehicles 2013 Ocean Electronics; 2013.
9. Rao SK. Algorithm for detection of maneuvering targets in bearings only passive target tracking. IEEE Proc Sonar Navigation 1999;146:141-6.
10. Rao SK. Modified gain extended Kalman filter with application to bearings only passive maneuvering target tracking. IEE Proc Radar Sonar Navig 2005;152(4):239-44.
11. Jawahar A, Rao SK, Murthy AS, Srikanth KS, Das RP. Multi-sensor submarine surveillance system using MGBEKF. Indian J Sci Technol 2015;8(35). DOI: 10.17485/ijst/2015/v8i35/82088.
12. Morgado M, Oliveira P, Silvestre C, Vasconcelos JF. Embedded vehicle dynamics aiding for USBL/INS underwater navigation system. IEEE Trans Control Syst Technol 2014;22(1):322-30.
13. Jayasiri A, Gosine RG, Mann GK, McGuire P. AUV based plume tracking: A simulation study. J Control Sci Eng 2016;2016:Article ID: 1764527, 15.
14. Packard GE, Kukulya A, Austin T, *et al*. Continuous autonomous tracking and imaging of white sharks and basking sharks using a remus-100 AUV. In: Proceedings of the IEEE Oceans. San Diego, California, USA: September; 2013. p. 1-5.
15. Stilinovic N, Nad D, Miskovic N. AUV for diver assistance and safety—design and implementation. In: Proceedings of the Oceans. Geneva, Switzerland: May; 2015. p. 1-4.
16. Zhang T, Zeng W, Wan L, Ma S. Underwater target tracking based on Gaussian particle filter in looking forward sonar images. J Comp Inform Syst 2010;6(14):4801-10.
17. Krout DW, Kooiman W, Okopal G, Hanusa E. Object tracking with imaging sonar. In: Proceedings of the 15th International Conference on Information Fusion (FUSION '12). IEEE, Singapore, September; 2012. p. 2400-2405.
18. Han KM, Choi HT. Shape context based object recognition and tracking in structured underwater environment. In: Proceedings of the IEEE International Conference on Geoscience and Remote Sensing Symposium; 2011. p. 617-620.
19. Chen SY. Kalman filter for robot vision: A survey. IEEE Trans Ind Electron 2012;59(11):4409-20.
20. Eickstedt D, Benjamin M, Schmidt H, Leonard JJ. Adaptive tracking of underwater targets with autonomous sensor networks. J Underwater Acoustics 2006;56:465-95.
21. Shi H, Wang W, Kwok NM, Chen SY. Game theory for wireless sensor networks: A survey. Sensors 2012;12(7):9055-97.
22. Chen S, Huang W, Cattani C, Altieri G. Traffic dynamics on complex networks: A survey. Mathematical Problems Eng 2012;2012:Article ID: 732698, 23.
23. Xue J, Li M, Zhao W, Chen S. Bound maxima as a traffic feature under DDOS flood attacks. Mathematical Problems Eng 2012;2012:Article ID: 419319, 20.
24. Xu LZ, Li XF, Yang SX. Wireless network and communication signal processing. Intellig Autom Soft Comput 2011;17(8):1019.
25. Akyildiz IF, Pompili D, Melodia T. Underwater acoustic sensor networks: research challenges. Ad Hoc Networks 2005;3(3):257-79.
26. Yu CH, Lee KH, Choi JW, Seo YB. Distributed single target tracking in underwater wireless sensor networks. In: Proceedings of the SICE Annual Conference. Japan, August; 2008. p1351-6.
27. Isbitiren G, Akan OB. Three-dimensional underwater target tracking with acoustic sensor networks. IEEE Trans Vehicular Technol 2011;60(8):3897-906.
28. Kim E, Lee S, Kim C, Kim K. Bearings-only tracking systems with distributed floating beacons in underwater sensor networks. In: Proceedings of the IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing, (EUC 10). December; 2010. p311-315.
29. Xu LZ, Ding XF, Wang X, Lv GF, Huang FC. Trust region based sequential quasi-Monte Carlo filter. Acta Electron Sin 2011;39(3A):24-30.
30. Chen SY, Wang ZJ. Acceleration strategies in generalized belief propagation. IEEE Trans Ind Inform 2012;8(1):41-8.
31. Ding XF, Xu LZ, Wang X, Guofang Lv, Xuewen Wu. Robust visual object tracking using covariance features in QuasiMonte Carlo filter. Intellig Autom Soft Comput 2011;17(5):571-82.
32. Wang X, Tang Z. Modified particle filter-based infrared pedestrian tracking. Infrared Phys Technol 2010;53(4):280-7.
33. Chen SY, Tong H, Wang Z, Liu S, Li M, Zhang B. Improved generalized belief propagation for vision processing. Mathematical Problems Eng 2011;2011:Article ID: 416963, 12.
34. Cattani C. Shannon wavelets for the solution of integrodifferential equations. Mathematical Problems Eng 2010;2010Article ID: 408418, 22.
35. Jawahar A, Chakravarthi CV. Comparative analysis of non linear estimation schemes used for undersea sonar applications. Innov J Eng Technol 2016;4(4):14-9.
36. Jawahar A, Chakravarthi CV. Improved nonlinear signal estimation technique for undersea sonar-based naval applications. Innov J Eng Technol 2016;4(4):20-5.
37. Jawahar A. Feasible course trajectories for Undersea sonar target tracking systems. Indian J Autom Artif Intellig 2016;3(1):1-5.
38. Jawahar A, Rao SK, Karishma SB. Target estimation analysis using data association and fusion. Int J Oceans Oceanograp 2015;9(2):203-10.
39. Prasanna KL, Rao SK, Jawahar A, Karishma SB. Modern estimation technique for undersea active target tracking. Int J Eng Technol 2016;8(2):791-803.
40. Prasanna KL, Rao SK, Jawahar A, Karishma SB. Ownship strategies during hostile torpedo attack. Indian J Sci Technol 2016;9(16). DOI: 10.17485/ijst/2016/v9i16/85556.

APPENDIX

Spherical coordinates and the GPS

In this module, we develop simplified formulas for finding the distance between two points on the earth. The coordinates of each of the two points are given in the form:

AAA degrees, BB.BBB minutes

This is the form used by many global positioning systems. We use a simplified model of the earth. In this model, the earth is a sphere whose radius is 6367 km. Because the earth is not a sphere, this model is somewhat inaccurate. For our purposes, it is a good first approximation. See the module. The earth is round - Most maps are flat for more details.

The first step is to convert the measurements of latitude and longitude into a more usable form. Since there are 60 minutes in a degree, the number of degrees is given by

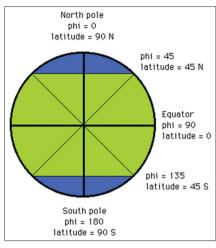$$\text{Degrees} = \text{AAA} + \text{BB.BBB}/60$$

We use spherical coordinates in this module. Appendix Fig 1 compares spherical coordinates (in degrees) with latitude coordinates. In spherical coordinates, we measure the angle phi from the north pole. Thus, the north pole corresponds to phi = 0; the equator to phi = 90°; and the south pole to phi = 180°.

Thus, the following formula converts from latitude expressed in degrees to phi also expressed in degrees.

$$\text{phi} = \begin{cases} /90 \ - \ \text{latitude if latitude is North} \\ \backslash 90 \ + \ \text{latitude if latitude is South} \end{cases}$$

In spherical coordinates, we measure the angle theta starting at the prime meridian (longitude 0) and moving east. Thus,

$$\text{Theta} = \begin{cases} / \ +\text{longitude if longitude is East} \\ \backslash \ -\text{longitude if longitude is West} \end{cases}$$

Appendix Fig. 1: Spherical coordinates with latitude coordinates

Both phi and theta as described above are measured in degrees. It is mathematically much better to measure angles in radians. The conversion formula is

$$\text{Angle in radians} = \frac{\text{Angle in degrees} \ * \ 2 \ * \ \text{Pi}}{360}$$

We want to express the location of a point in Cartesian coordinates with the origin at the center of the earth, the north pole at the point

North pole = (0, 0, 6367 km)

And the positive X-axis going through the prime meridian. The conversion formulas are:

x = 6367 (Cos theta)(Sin phi)

y = 6367 (Sin theta)(Sin phi)

z = 6367 (Cos phi)

Author Query???
AQ1:   Kindly provide author corresponding author email id.