Turnrate Estimation in the Adrian Multi Radar Tracker in PHOENIX

André Adrian

Deutsche Flugsicherung GmbH
Am DFS Campus, 63225 Langen, Germany
email: Andre.Adrian@dfs.de

Abstract: *A maneuver in the traditional Kalman Filter is estimated with a straight line model and a lot of process noise. Normally a maneuver is flown as "coordinated turn", that is an arc with constant angular velocity (turnrate).*
It is possible to use turnrate as a state-variable in an Extended Kalman Filter. If the system has only incomplete measurements, like Line-of-Sights, an EKF is a must. In Air Traffic Control where complete measurements are available the EKF is not welcome.
The Interactive Multi Model Kalman Filter uses a bank of Kalman Filters for different Mode-of-Flight models. Typically there is a straight line, a 3°/s left-turn and a 3°/s right-turn KF in the IMMKF. Unfortunately only a small minority of aircrafts perform "standard turns". Most aircraft fly 2°/s, 4°/s or whatever turns. Furthermore, the IMMKF is expensive in computation.
The turnrate estimation in the AMRT is a very straightforward algorithm. A measurement-heading is calculated out of two position informations (plots) as delivered from the sensor, a surveillance radar. With two headings and the time difference one measurement-turnrate can be calculated. Up to now everything is simple mathematics. The topic goes "statistical" due to the measurement noise. To get rid of the measurement noise two strategies are used:
First, the heading calculation is not done with adjacent plots, but with plots that are n measurement intervals apart. With larger n the "truth" overcomes more and more the "noise", but maneuver detection time gets larger, too. The paper will show the methods that were used to find the optimum n for a given scenario.
Second, a "smoothed turnrate" can be used as input-variable to the Kalman Filter. The paper shows the tested algorithms and the advantages/disadvantages to the position accuracy at the output of the Kalman Filter.
The presented turnrate-estimation algorithm is much cheaper in computation then the EKF or IMMKF. Nevertheless it gives the "good old linear" KF a better coordinated turn maneuver capability.

Radar Simulator

Tracker evaluation can not be done with live radar data, because live data is noisy. A radar simulator creates two outputs: one with noise as input for the tracker under test and one without noise for the error calculation. The radar simulator shall create realistic errors. To do this, noise is not added to the cartesian target position but to the polar target position.
The flight-path describes the movement of the target. To test the working of the turnrate-estimation, we need different Mode-of-Flight in the flight-path. Therefore, the target performs a straight line segment, a 180° left turn. another straight line segment and a 150° right turn. This pattern gives a flower-like flight-path with 12 "leaves". It is a departure/approach scenario flown with 12 different runway directions.

Radargenerator $s_0 := 15 \cdot 1852$ Uniform Movement length

Straight Turn Straight Turn

$n_1 := \mathrm{rund}\left(\frac{s_0}{v_0 \cdot t}\right)$ $n_2 := n_1 + \frac{\pi}{\omega \cdot t}$ $n_3 := n_2 + n_1$ $n := \mathrm{rund}\left(n_3 + \frac{5}{6} \cdot \frac{\pi}{\omega \cdot t}\right)$ $n = 130$

Define the repetion $m := 4 \cdot 12$

Transition matrix

$$F(\omega) := \mathrm{wenn}\left[\omega = 0, \begin{pmatrix} 1 & t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & \frac{\sin(\omega\cdot t)}{\omega} & 0 & -\frac{1-\cos(\omega\cdot t)}{\omega} \\ 0 & \cos(\omega\cdot t) & 0 & -\sin(\omega\cdot t) \\ 0 & \frac{1-\cos(\omega\cdot t)}{\omega} & 1 & \frac{\sin(\omega\cdot t)}{\omega} \\ 0 & \sin(\omega\cdot t) & 0 & \cos(\omega\cdot t) \end{pmatrix}\right]$$

The movement with measurement noise $i := 1 .. m\cdot n$ $\sigma_v := \frac{1}{2}\cdot\sigma_{va}\cdot t^2$ $\sigma_v = 12.263$

acceleration noise $nvx := \mathrm{rnorm}\left(m\cdot n + 1, 0, \sqrt{\sigma_v}\right)$ $nvy := \mathrm{rnorm}\left(m\cdot n + 1, 0, \sqrt{\sigma_v}\right)$

$$v^{\langle i\rangle} := \begin{pmatrix} nvx_i \\ 0 \\ nvy_i \\ 0 \end{pmatrix} \qquad x^{\langle 0\rangle} := \begin{pmatrix} 0 \\ v_0\cdot\sin(h_0) \\ 0 \\ v_0\cdot\cos(h_0) \end{pmatrix} + v^{\langle 0\rangle}$$

x position
x speed

y position
y speed

$$x^{\langle i\rangle} := \mathrm{wenn}\left(\mathrm{mod}(i,n) < n_1, F(0)\cdot x^{\langle i-1\rangle}, \mathrm{wenn}\left(\mathrm{mod}(i,n) < n_2, F(\omega)\cdot x^{\langle i-1\rangle}, \mathrm{wenn}\left(\mathrm{mod}(i,n) < n_3, F(0)\cdot x^{\langle i-1\rangle}, F(-\omega)\cdot x^{\langle i-1\rangle}\right)\right)\right) + v^{\langle i\rangle}$$

$s_x := \mathrm{mittelwert}\left[\left(x^T\right)^{\langle 0\rangle}\right]$ $s_y := \mathrm{mittelwert}\left[\left(x^T\right)^{\langle 2\rangle}\right]$ Sensor in the center of the "flower"

Measurement in polar coordinates with white noise

$n\rho := \mathrm{rnorm}\left(m\cdot n + 1, 0, \sigma_\rho\right)$ $n\theta := \mathrm{rnorm}\left(m\cdot n + 1, 0, \sigma_\theta\right)$ $i := 0 .. m\cdot n$

$a_{0,i} := \sqrt{\left(x_{0,i} - s_x\right)^2 + \left(x_{2,i} - s_y\right)^2} + n\rho_i$ $a_{1,i} := \mathrm{atan2}\left(x_{2,i} - s_y, x_{0,i} - s_x\right) + n\theta_i$

Measurement in cartesian coordinates with white noise

$$z^{\langle i\rangle} := \begin{pmatrix} s_x + a_{0,i}\cdot\sin(a_{1,i}) \\ s_y + a_{0,i}\cdot\cos(a_{1,i}) \end{pmatrix}$$

Heading Definition: North = 0°, clockwise

$j := 0 .. n\cdot 11$

$\sigma_\rho \equiv 0.04\cdot 1852$ $\sigma_\rho = 74.08$ $t \equiv 5$ Scanrate in s

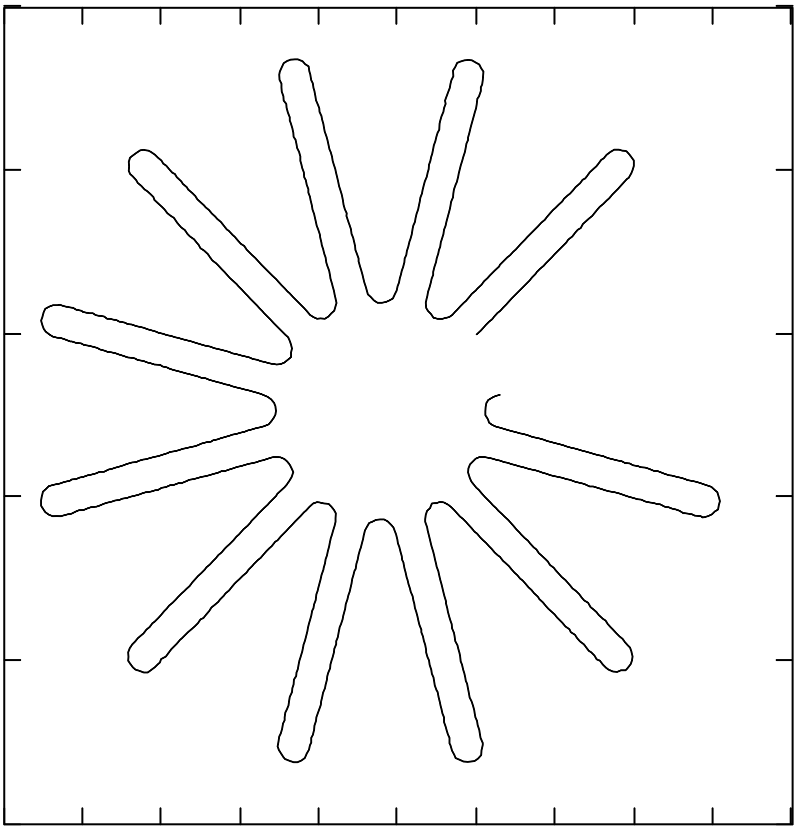
$\sigma_\theta \equiv 0.07\cdot \mathrm{Grad}$ Measurement Noise

$\sigma_{va} \equiv 0.1\cdot 9.81$ Process Noise as Acceleration Noise $\omega \equiv 3\cdot \mathrm{Grad}$ Turnrate in °/s

$h_0 \equiv 45\mathrm{Grad}$ Start Heading $v_0 \equiv 200\cdot\frac{1852}{3600}$ Speed in m/s

The groundspeed is 200 nautical miles per hour (Knots), the turnrate is 3°/s. Radar scantime is 5 seconds, range standard deviation is 0.04 nautical miles (NM), azimuth is 0.07° and detection probability is 100%. The straight line segments are 15 NM long. The radar is located at the center of the "flower".

first 11 "leaves" of Flight-Path



Confirmation by Monte Carlo Simulation

"Who can, do. Who can not, simulate". The author is not able to perform a mathematical analysis for the problem of turnrate-estimation, therefore a Monte Carlo simulation is used as surrogate. Everytime a simulation is used instead of an analysis the risk of hyping a stupid idea is given.
After Monte Carlo simulations are finished successful, a system goes into use. In the real world the Monte Carlo results get confirmed or not. If, after some years of use, the system has a good reputation, one can assume that the simulation results were transferable.
Still a little caution: Humans tend to superstition. Sometimes null changes create a good feeling in everybody: We have done something, the problem went away, we did the RIGHT THING(TM). Instead, the problem has just gone away for a while, and will come back later, ruining the believe in the right thing.
For example: In Germany we have every winter problems with primary clutter that goes away every spring. A good candidate for null, void or invalid changes.

Turnrate Estimation Algorithm 1

The first algorithm estimates the turnrate on "measurement headings". In real use the following formulas would be part of the tracking algorithm. One new plot comes in, the heading and turnrate estimates are calculated and with this input-variable the Kalman Filter is evaluated.
For demonstration it is easier to run the turnrate-estimate as block algorithm. The interval length is 4, that is the heading is calculated with the new plot and the 4. last plot.

$N := n \cdot m$

$i := 2..N \quad hx_i := \text{atan2}(x_{1,i}, x_{3,i})$ Reference Heading

$i := 3..N \quad wx_i := \dfrac{\text{csub}(hx_i, hx_{i-1})}{t}$ Reference Turnrate

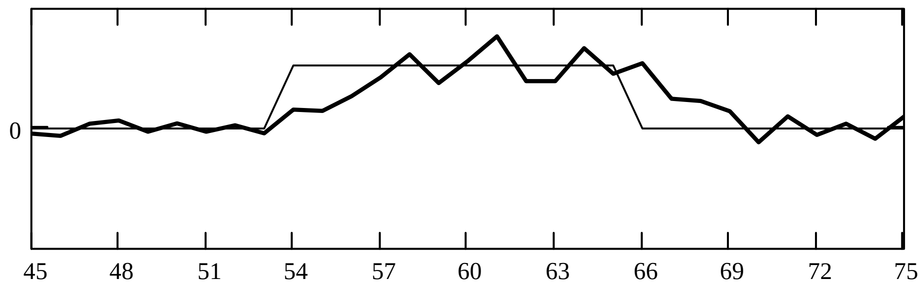
$i := 4..N \quad h4_i := \text{atan2}(z_{0,i} - z_{0,i-4}, z_{1,i} - z_{1,i-4})$ Measurement heading estimate

$\text{csub}(a,b) \equiv [a - b + (a - b \le -\pi) \cdot 2 \cdot \pi - (a - b > \pi) \cdot 2 \cdot \pi]$ Difference of angles in circle

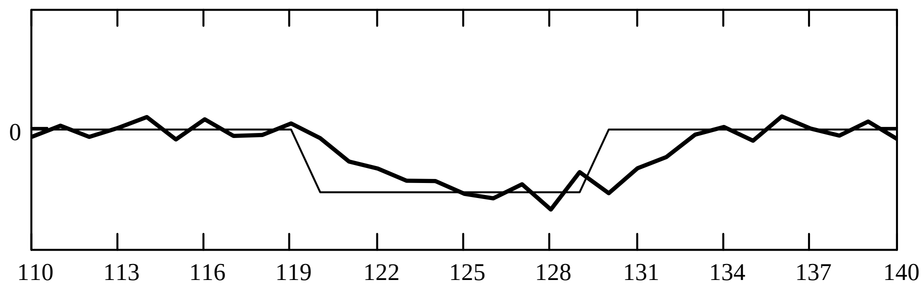
$i := 5..N \quad wr4_i := \dfrac{\text{csub}(h4_i, h4_{i-1})}{t}$ Measurement turnrate estimate

The diagrams show the estimated turnrate for a left turn and for a right turn. The thin line is the turnrate without noise, the thick line is the estimate. One can see that the interval creates a delay in Mode-of-Flight change detection. This is the typical trade-off between an agile system responding with a lot of noise and an inert system responding with less noise. The x-axis has plot number as unit.

Turnrate estimate with interval n=4



Turnrate estimate with interval n=4



Error calculation is done with the absolute difference. For successful tracker evaluation the maximum error is the most critical value.

$ewr4_i := |wr4_i - wx_i| \qquad \dfrac{\text{stdev}(ewr4)}{\text{Grad}} = 0.579 \qquad \dfrac{\max(ewr4)}{\text{Grad}} = 4.599$

Turnrate Estimation Algorithm 2

The second algorithm estimates the turnrate on "smoothed headings", that is on the state-variables of the Kalman-Filter. This approach has the disadvantage that the turnrate-estimate is calculated after the KF algorithm. We will see that the advantage of using smoothed heading overcomes the disadvantage of having a delayed turnrate-estimate.
But first the presentation of our KF. We use a position/velocity KF with state vector y elements x-position, x-velocity, y-position and y-velocity. The sensor-errors are transformed from polar to cartesian with matrix R. The measurement matrix H shows that we have position-only measurements. The covariance matrix P is initialized with some "large numbers", that the KF algorithm will quickly reduce to correct numbers. Q, the process noise matrix uses a "piecewise constant white acceleration" model. We use a "coordinated turn" transition matrix F. Process noise σ_a is 0.1 gravo. Smoothed turnrate is w.

$$R(\rho,\theta) := \begin{bmatrix} (\sigma_\rho \cdot \sin(\theta))^2 + (\rho\cdot\sigma_\theta\cdot\cos(\theta))^2 & (\sigma_\rho{}^2 - \rho^2\cdot\sigma_\theta{}^2)\cdot\sin(\theta)\cdot\cos(\theta) \\ (\sigma_\rho{}^2 - \rho^2\cdot\sigma_\theta{}^2)\cdot\sin(\theta)\cdot\cos(\theta) & (\sigma_\rho\cdot\cos(\theta))^2 + (\rho\cdot\sigma_\theta\cdot\sin(\theta))^2 \end{bmatrix}$$

$$R1 := R(a_{0,1}, a_{1,1})$$

$$r := \max(R1_{0,0}, R1_{1,1})$$

$$H := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad P := \mathrm{diag}\left(\begin{pmatrix} r \\ \frac{2\cdot r}{t^2} \\ r \\ \frac{2\cdot r}{t^2} \end{pmatrix}\right) \qquad Q := \begin{pmatrix} \frac{t^4}{4} & \frac{t^3}{2} & 0 & 0 \\ \frac{t^3}{2} & t^2 & 0 & 0 \\ 0 & 0 & \frac{t^4}{4} & \frac{t^3}{2} \\ 0 & 0 & \frac{t^3}{2} & t^2 \end{pmatrix}$$

$$\sigma_a \equiv 0.1\cdot 9.81$$

$$F(\omega) := \mathrm{wenn}\left[\omega = 0, \begin{pmatrix} 1 & t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & \frac{\sin(\omega\cdot t)}{\omega} & 0 & -\frac{1-\cos(\omega\cdot t)}{\omega} \\ 0 & \cos(\omega\cdot t) & 0 & -\sin(\omega\cdot t) \\ 0 & \frac{1-\cos(\omega\cdot t)}{\omega} & 1 & \frac{\sin(\omega\cdot t)}{\omega} \\ 0 & \sin(\omega\cdot t) & 0 & \cos(\omega\cdot t) \end{pmatrix}\right]$$

$$KF(n) := \left| \begin{array}{l} y^{\langle 1 \rangle} \leftarrow \left(z_{0,1} \quad \frac{z_{0,1} - z_{0,0}}{t} \quad z_{1,1} \quad \frac{z_{1,1} - z_{1,0}}{t} \right)^T \\ \text{for } i \in 0..n \\ \quad w_i \leftarrow 0 \\ \text{for } i \in 2..N \\ \quad \left| \begin{array}{l} y^{\langle i \rangle} \leftarrow F(w_{i-1}) \cdot y^{\langle i-1 \rangle} \\ \nu \leftarrow z^{\langle i \rangle} - H \cdot y^{\langle i \rangle} \\ P \leftarrow F(w_{i-1}) \cdot P \cdot F(w_{i-1})^T + Q \cdot \sigma_a^{\ 2} \\ S \leftarrow H \cdot P \cdot H^T + R(a_{0,i}, a_{1,i}) \\ W \leftarrow P \cdot H^T \cdot S^{-1} \\ y^{\langle i \rangle} \leftarrow y^{\langle i \rangle} + W \cdot \nu \\ P \leftarrow P - W \cdot S \cdot W^T \\ w_i \leftarrow \frac{csub(atan2(y_{1,i}, y_{3,i}), atan2(y_{1,i-n}, y_{3,i-n}))}{n \cdot t} \quad \text{if } i > n \end{array} \right. \\ (y \quad w)^T \end{array} \right.$$

KF with Turnrate estimate

y is state prediction

ν is measurement residual

P is state prediction covariance

S is innovation covariance

W is filter gain

y is updated state estimate

P is updated state covariance

$yw := KF(4) \qquad y := yw_0 \qquad ws4 := yw_1$

Like above we can calculate the turnrate-errors.

$ews4_i := \left| ws4_i - wx_i \right| \qquad \frac{stdev(ews4)}{Grad} = 0.586 \qquad \frac{max(ews4)}{Grad} = 3.356$

With algorithm 2 the maximum error is smaller. Now we compare the position accuracy of the turnrate-estimating KF with the traditional KF.

The measurement position noise $\qquad i := 4..N$

$dzx_i := \sqrt{(z_{0,i} - x_{0,i})^2 + (z_{1,i} - x_{2,i})^2} \qquad \sigma_{zx} := Stdev(dzx) \qquad maxdzx := max(dzx)$

$\frac{\sigma_{zx}}{1852} = 0.023 \qquad \frac{maxdzx}{1852} = 0.159$

The KF output position noise from turnrate-estimate algorithm 2 is

$dyx_i := \sqrt{(y_{0,i} - x_{0,i})^2 + (y_{2,i} - x_{2,i})^2} \qquad \sigma_{yx} := Stdev(dyx) \qquad maxdyx := max(dyx)$

$\frac{\sigma_{yx}}{1852} = 0.019 \qquad \frac{maxdyx}{1852} = 0.121$

The Smoothed position noise reduction of the turnrate-estimate KF. $\qquad \left(1 - \frac{\sigma_{yx}}{\sigma_{zx}}\right)\frac{1}{\%} = 20.499 \qquad \left(1 - \frac{maxdyx}{maxdzx}\right)\frac{1}{\%} = 23.737$

$$kf(d) := \left| \begin{array}{l} y^{\langle 1 \rangle} \leftarrow \left(z_{0,1} \quad \frac{z_{0,1} - z_{0,0}}{t} \quad z_{1,1} \quad \frac{z_{1,1} - z_{1,0}}{t} \right)^T \\ \text{for } i \in 2..N \\ \quad \left| \begin{array}{l} y^{\langle i \rangle} \leftarrow F(0) \cdot y^{\langle i-1 \rangle} \\ \nu \leftarrow z^{\langle i \rangle} - H \cdot y^{\langle i \rangle} \\ P \leftarrow F(0) \cdot P \cdot F(0)^T + Q \cdot \sigma_a^2 \\ S \leftarrow H \cdot P \cdot H^T + R(a_{0,i}, a_{1,i}) \\ W \leftarrow P \cdot H^T \cdot S^{-1} \\ y^{\langle i \rangle} \leftarrow y^{\langle i \rangle} + W \cdot \nu \\ P \leftarrow P - W \cdot S \cdot W^T \end{array} \right. \\ y \end{array} \right.$$

KF with straight line only model

y is state prediction

n is measurement residual

P is state prediction covariance

S is innovation covariance

W is filter gain

y is updated state estimate

P is updated state covariance

$$y := kf(0)$$

The filter output position noise is

$$dyx_i := \sqrt{\left(y_{0,i} - x_{0,i}\right)^2 + \left(y_{2,i} - x_{2,i}\right)^2} \qquad \sigma_{yx} := \text{Stdev}(dyx) \qquad maxdyx := \max(dyx)$$

$$p^{\langle i \rangle} := F(0) \cdot y^{\langle i-1 \rangle}$$

$$dpx_i := \sqrt{\left(p_{0,i} - x_{0,i}\right)^2 + \left(p_{2,i} - x_{2,i}\right)^2} \qquad \sigma_{px} := \text{Stdev}(dpx) \qquad maxdpx := \max(dpx)$$

Now the linear KF. The Smoothed position noise reductions are

$$\frac{\sigma_{yx}}{1852} = 0.021 \qquad \frac{maxdyx}{1852} = 0.136 \qquad \left(1 - \frac{\sigma_{yx}}{\sigma_{zx}}\right)\frac{1}{\%} = 10.209 \qquad \left(1 - \frac{maxdyx}{maxdzx}\right)\frac{1}{\%} = 14.295$$

The Reduction of standard-deviation error is half of the reduction from the turnrate-estimate KF. The difference in performance for maximum errors is less.

Further Results

Tests with Interval lengths from 1 to 5 plots were performed. The best results are with n=3 or n=4. Every Turnrate-Estimate KF algorithm was better then the traditional KF.
The TEMPO Evaluation tested the multi-radar case with algorithm 1. The results are mixed. In en-route scenario, where all radars are more then 60 NM away from the targets, errors became smaller. In approach scenario with 2 airport surveillance radars algorithm 1 performed worse then the previous software version.
We have to TEMPO test algorithm 2 and have to check if the bad performance of the last software version does not have other reasons. "Bad performance" is relativ. The maximum position errors are good enough to pass the TEMPO evaluation.

Reference

[1] A. Adrian, "Design of the Adrian Multi Radar Tracker in Phoenix for Enhanced Mode-S Surveillance", Proc. International Radar Symposium 2003, Dresden, Germany, 30.9.-2.10.2003, pp161-166

[2] Adrian, Heidger, Klenner, Mallwitz, "the DFS Multiradar-Tracking System PHOENIX", Proc. German Radar Symposium 2002, Bonn, Germany, 3.-5.9.2002, pp123-130

[3] Heidger, Klenner, Mallwitz, "Mode-S Evaluation and Practical Implementation Results with the DFS Multiradar-Tracking System PHOENIX", Proc. International Radar Symposium 2003, Dresden, Germany, 30.9.-2.10.2003, pp145-153

[4] R. Mallwitz, "TEMPO Tracker Evaluation on Multi Sensor Performance. Phoenix, Phase 3", DFS, Langen, 2004

[5] Bar-Shalom, Li, "Estimation and Tracking: Principles, Techniques and Software", Artech House, Boston, London, 1993

[6] Bar-Shalom, Li, "Multitarget-Multisensor Tracking: Principles and Techniques", Yaakov Bar-Shalom, 1995