Kalman Filter Applications: Tracking and Aided Tracking

J Treurnicht

April 9, 2007

1 Continuous Linear Kalman Filters

The continuous Kalman Filter is an observer applied to a statistical problem, so we call it an *estimator*. In this problem, so-called process and measurement noise contaminates the the measured vector $\mathbf{x}$. The error vector $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ does not diminish to zero for $t \to \infty$. One possibility is to search for a gain $L(t)$ that will minimize the error vector by minimizing the sum of squares

$$\mathcal{E}\left[\int \tilde{\mathbf{x}}^2(t)\,dt\right]$$

1.1 Basic Result

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t) + \mathbf{w}(t)\\ \mathbf{y}(t) &= C(t)\mathbf{x}(t) + D(t)\mathbf{u}(t) + \mathbf{v}(t)\end{aligned}$$

with

$$\begin{array}{llll}\mathcal{E}\left[\mathbf{w}(t)\right] & = \mathcal{E}\left[\mathbf{v}(t)\right] & = 0 & \mathbf{w}(t) \text{ and } \mathbf{v}(t) \text{ zero mean}\\ \mathcal{E}\left[\mathbf{w}(t)\mathbf{w}^T(t)\right] & = Q(t)\,\delta & \geq 0 & \mathbf{w}(t) \text{ white}\\ \mathcal{E}\left[\mathbf{v}(t)\mathbf{v}^T(t)\right] & = R(t)\,\delta & > 0 & \mathbf{v}(t) \text{ white}\\ \mathcal{E}\left[\mathbf{w}(t)\mathbf{v}^T(t)\right] & = S(t)\,\delta & & \mathbf{w}(t) \text{ and } \mathbf{v}(t) \text{ correlated}\\ \mathcal{E}\left[\mathbf{w}(t)\mathbf{x}^T(t_0)\right] & = 0 & & \mathbf{w}(t) \text{ not correlated with } \mathbf{x}(t_0)\end{array}$$

The solution is

$$\begin{aligned}\dot{\hat{\mathbf{x}}}(t) &= A(t)\hat{\mathbf{x}}(t) + B(t)\mathbf{u}(t) + L(t)\left[\mathbf{u}(t) - C(t)\hat{\mathbf{x}}(t) - D(t)\mathbf{u}(t)\right]\\ \hat{\mathbf{y}}(t) &= \hat{\mathbf{x}}(t)\end{aligned}$$

The observer state error is

$$\tilde{\mathbf{x}}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t)$$

If we let the covariance of the state error equals $P(t)$ then

$$\text{trace}\left[P(t)\right] = \mathcal{E}\left[\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}^T(t)\right]$$

The gain $L(t)$ which minimizes the trace of $P(t)$ is obtained from the Ricatti equation set

$$\begin{aligned} L(t) &= \left[P(t)C^T(t) + S(t)\right] R^{-1}(t) \\ \dot{P}(t) &= A(t)P(t) + P(t)A^T(t) - \left[P(t)C^T(t) + S(t)\right] R^{-1}(t) \left[S^T(t) + C(t)P(t)\right] + Q(t) \end{aligned}$$

Note: If $R(t)$ is not positive definite, then $R^{-1}(t)$ will not exist — This is an important requirement for the formulation of this problem.

If the system is linear time invariant (LTI), then the solution becomes

$$\begin{aligned} \dot{\hat{\mathbf{x}}}(t) &= A\hat{\mathbf{x}}(t) + B\mathbf{u}(t) + L(t)\left[\mathbf{u}(t) - C\hat{\mathbf{x}}(t) - D\mathbf{u}(t)\right] \\ L(t) &= \left[P(t)C^T + S\right] R^{-1} \\ \dot{P}(t) &= AP(t) + P(t)A^T - \left[P(t)C^T + S\right] R^{-1} \left[S^T + CP(t)\right] + Q \end{aligned}$$

P is still a function of time and depends on $P(t_0)$ which depends on $\tilde{\mathbf{x}}(t_0)$.

When $t \to \infty$, then $L(t = \infty)$ will approach a constant value L_∞ which we can solve by letting $\dot{P} = 0$. This leads to the algebraic Ricatti equation

$$\begin{aligned} L_\infty &= \left[P_\infty C^T + S\right] R^{-1} \\ 0 &= AP_\infty + P_\infty A^T - \left[P_\infty C^T + S\right] R^{-1} \left[S^T + CP_\infty\right] + Q \end{aligned}$$

Notes:

1. If $\mathbf{w}(t)$ and $\mathbf{v}(t)$ is uncorrelated, the $S(t) = 0$ and

$$\begin{aligned} L(t) &= P(t)C^T R^{-1} \\ \dot{P}(t) &= AP(t) + P(t)A^T - P(t)C^T R^{-1} CP(t) + Q \end{aligned}$$

2. If we compare the result with that obtained from the optimal control problem

$$\begin{aligned} K(t) &= R^{-1}B^T P(t) \\ -\dot{P}(t) &= P(t)A + A^T P(t) - P(t)BR^{-1}B^T P(t) + Q \end{aligned}$$

   so we can relate the solutions according to the following table

Control		Estimator
A	$\longleftrightarrow$	A^T
B	$\longleftrightarrow$	C^T
K	$\longleftrightarrow$	L^T
$-t$	$\longleftrightarrow$	t

3. If we choose $L(t)$ by any other means (not optimal), then we can still get the covariance from the differential equation

$$\dot{P}(t) = [A(t) - L(t)C(t)]P(t) + P(t)[A(t) - L(t)C(t)]^T - L^T(t)R^{-1}(t)L^T(t) - L(t)S(t) - S(t)L^T(t) + Q$$

4. If the stochastic variables $\mathbf{w}(t)$ and $\mathbf{v}(t)$ have Gaussian distributions, then $\hat{\mathbf{x}}(t)$ is the best estimate of $\mathbf{x}(t)$. The reverse is also true — which can lead to bad results.

1.2 Special Cases

1.2.1 Measurement Noise R only Positive Semi-definite

Re-arrange the output equation with the top block that includes the measurement noise.

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} D_1 \\ D_2 \end{bmatrix} \mathbf{u}(t) + \begin{bmatrix} v \\ 0 \end{bmatrix}, \quad R = \begin{bmatrix} R_1 & 0 \\ 0 & 0 \end{bmatrix}$$

Find a submatrix T_2 so that

$$\begin{bmatrix} C_2 \\ T_2 \end{bmatrix} = \text{ non-singular } = \begin{bmatrix} M_2 & N_2 \end{bmatrix}^{-1}$$

and define $\eta = T_2\mathbf{x}$

If we choose our T_2 of the form

$$T_2 = \begin{bmatrix} I_k & 0 \end{bmatrix} \quad \rightarrow \quad \eta = \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ 0 \end{bmatrix}$$

then we can find $\mathbf{x}(t)$ from the noise-free output y_2 through

$$\mathbf{x}(t) = M_2 \left(y_2(t) - D_2\mathbf{u}(t)\right) + N_2\eta$$

We can now proceed to build a Kalman Filter to estimate $\hat{\eta}$ from the system

$$\dot{\eta} = T_2AN_2\eta + T_2AM_2y_2 + (T_2B - T_2AM_2D_2)\mathbf{u} + T_2w$$

and noisy output equation

$$y_1 = C_1N_2\eta + C_1M_2y_2 + (D_1 - C_1M_2D_2)\mathbf{u} + v_1$$

Through analogy

$$\begin{array}{lll} A^* = T - 2AN_2 & C^* = C_1N_2 & Q^*\delta = \mathcal{E}[T_2ww^TT_2^T] \\ S^*\delta = \mathcal{E}[T_2wv_1^T] & R^*\delta = \mathcal{E}[T_2v_1v_1^T] & R^* = \text{ positive definite!} \end{array}$$

with solution

$$\begin{aligned} L(t) &= \left[P(t)C^{*T} + S^*\right] R^{*-1} \\ \dot{P}(t) &= A^*P(t) + P(t)A^{*T} - \left[P(t)C^{*T} + S^*\right] R^{-1} \left[S^{*T} + C^*P(t)\right] + Q \end{aligned}$$

1.2.2 Measurement Noise $R = 0$ (no measurement noise)

The output equation may be differentiated to obtain white noise

$$y = C\mathbf{x} \quad \rightarrow \quad \dot{y} = C\dot{\mathbf{x}} = CA\mathbf{x} + CB\mathbf{u} + Cw$$

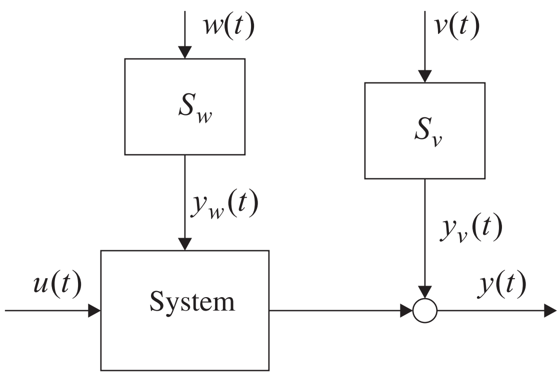
and we can partition the output equation then as $y_1 = y,\ y_2 = \dot{y}$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} CA \\ C \end{bmatrix} \mathbf{x} + \begin{bmatrix} CB \\ 0 \end{bmatrix} \mathbf{u} + \begin{bmatrix} Cw \\ 0 \end{bmatrix}$$

and proceeding as with the semi-definite case.

1.2.3 Coloured (bandlimited) Process and Measurement Noise

Coloured process noise is the result of white noise that is passed through a (typical low-pass) filter.



1. Coloured process noise is augmented as follows

$$\begin{aligned} S_w: \quad \dot{\mathbf{x}}_w &= A_w\mathbf{x}_w + B_w w \\ y_w &= C_w\mathbf{x}_w + D_w w \end{aligned}$$

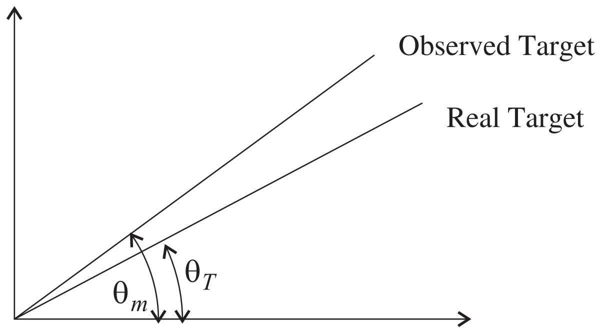
2. Coloured measurement noise is augmented as follows

$$\begin{aligned} S_v: \quad \dot{\mathbf{x}}_v &= A_v\mathbf{x}_v + B_v v \\ y_v &= C_v\mathbf{x}_v + D_v v \end{aligned}$$

The augmented system becomes

$$\begin{aligned} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{x}}_w \\ \dot{\mathbf{x}}_v \end{bmatrix} &= \begin{bmatrix} A & C_w & 0 \\ 0 & A_w & 0 \\ 0 & 0 & A_v \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_w \\ \mathbf{x}_v \end{bmatrix} + \begin{bmatrix} D_w \\ B_w \\ 0 \end{bmatrix} w + \begin{bmatrix} 0 \\ 0 \\ B_v \end{bmatrix} v + \begin{bmatrix} B \\ 0 \\ 0 \end{bmatrix} \mathbf{u} \\ y &= \begin{bmatrix} C & 0 & C_v \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{x}}_w \\ \dot{\mathbf{x}}_v \end{bmatrix} + D\mathbf{u} + D_v v \end{aligned}$$

Example Radar tracker (angle only)



The target dynamics are assumed to be able to provide an acceleration perpendicular to the line-of-sight with statistical intensity q_2.

$$\ddot{\theta} = w_2$$

Choosing states $x_1 = \theta_T,\ x_2 = \dot{\theta}_T$, then our system is described by

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, w = \begin{bmatrix} 0 \\ w_2 \end{bmatrix}$$

For measurement, let $y = \theta_m$ and let $\theta_m - \theta_T = v$, then the output description is

$$y = \mathbf{x} + v, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D = 0$$

Let the Kalman filter feedback vector be $L = [l_1\ l_2]^T$. After feedback, our filter equation becomes

$$\begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \end{bmatrix} = \begin{bmatrix} -l_1 & 1 \\ -l_2 & 0 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} y$$

Using the uncorrelated steady-state Kalman filter equations

$$\begin{aligned} L_\infty &= P_\infty C^T R^{-1} \\ 0 &= AP_\infty + P_\infty A^T - P_\infty C^T R^{-1} C P_\infty + Q \end{aligned}$$

with

$$Q = \begin{bmatrix} 0 & 0 \\ 0 & q_2 \end{bmatrix}, \quad R = r$$

we can fairly easily find the analytical solution

$$P_\infty = \begin{bmatrix} \sqrt{2r\gamma} & \gamma \\ \gamma & \sqrt{2q_2\gamma} \end{bmatrix} \quad \gamma = \sqrt{q_2 r}$$

$$L_\infty = \begin{bmatrix} \sqrt{2}\ \left(\frac{q_2}{r}\right)^{1/4} \\ \left(\frac{q_2}{r}\right)^{1/2} \end{bmatrix}$$

The characteristic polynomial is given by

$$s^2 + l_1 s + l_2 = 0$$

with natural frequency $\omega_n = \sqrt{l_2} = \left(\frac{q_2}{r}\right)^{1/4}$ and damping $\zeta = 1/\sqrt{2}$ (always). This fact is supported by the symmetrical root locus.

Note:

- As r increases (more measurement noise), then ω_n will decrease — lower bandwidth
- As q_2 increases (more process noise), then ω_n will increase — higher bandwidth

For the case when $q_2 = 4 \times 10^{-8}\,\text{rad}^2\text{sec}^{-4}$ and $r = 9 \times 10^{-8}\,\text{rad}^2$, the solution is

$$P = \begin{bmatrix} 1.04 & 0.6 \\ 0.6 & 0.69 \end{bmatrix} \times 10^{-7}, \quad L = \begin{bmatrix} 1.16 \\ 0.67 \end{bmatrix}$$

with errors

$$\sigma^2(\theta_T) = 1.04 \times 10^{-7}\,\text{rad}^2$$
$$\sigma^2(\dot{\theta}_T) = 0.69 \times 10^{-7}\,(\text{rad/sec})^2$$

The transfer function from input $(y = \theta_T + v!)$ to filtered output is

$$\begin{aligned} \frac{\hat{\theta}_T}{\theta_T} = & \frac{l_1 s + l_2}{s^2 + l_1 s + l_2} \\ & \frac{... + l_2}{... + l_2} \quad \rightarrow \text{zero mean postion error} \\ & \frac{... + l_1 s + ...}{... + l_1 s + ...} \quad \rightarrow \text{zero mean speed error} \end{aligned}$$

See the papers by Singer in [1] and [2] for the solution to the general aircraft tracking problem. The paper by Watanabe in [3] provides closed form solutions to the common continuous tracking filters and show how the noise, bandwidth and gain is related.

Note: The dynamic error coefficients for any stable linear control system are given by

$$e = \frac{1}{k_p} r + \frac{1}{k_v} \dot{r} + \frac{1}{k_a} \ddot{r} + \ldots$$

with the coefficients generated through long-division of the error transfer function

$$\frac{E}{R}(s) = \frac{1}{1 + DG(s)}$$

What are the relationship with the static error constants K_p, K_v, ... The dynamic error constants are true for all input signals and the static ones only true in the steady-state.

Example

$$DG(s) = 10\,\frac{1}{s+1}$$

then

$$\frac{1}{1+DG(s)} = \frac{1}{1+\frac{10}{s+1}} = \frac{s+1}{s+11}$$

Perform the inverse long division (we want to get $1/k_x$!)

$$\begin{array}{rl|llll}
 & & \frac{1}{11} & -\frac{10}{121}\,s & +\frac{10}{1331}\,s^2 & \\ \cline{3-6}
11+s & | & 1+ & s & & \\
 & & 1+ & \frac{1}{11}\,s & & \\ \cline{3-6}
 & & & -\frac{10}{11}\,s & & \\
 & & & -\frac{10}{11}\,s & -\frac{10}{121}\,s^2 & \\ \cline{4-6}
 & & & & \frac{10}{121}\,s^2 & \\
 & & & & \frac{10}{121}\,s^2 & +\frac{10}{1331}\,s^3 \\ \cline{5-6}
\end{array}$$

giving

$$e = \frac{1}{11}\,r - \frac{10}{121}\,\dot{r} + \frac{10}{1331}\,\ddot{r} + \dots$$

For an input $r = 0.1\sin(0.2\,t)$, we have $\dot{r} = 0.02\cos(0.2\,t)$ and $\ddot{r} = -0.004\sin(0.2\,t)$ so the errors are:

$$e = 0.009\sin(0.2\,t) - 0.00165\cos(0.2\,t) - 3\times 10^{-5}\sin(0.2\,t) + \dots$$

2 Continuous Non-Linear Kalman Filters

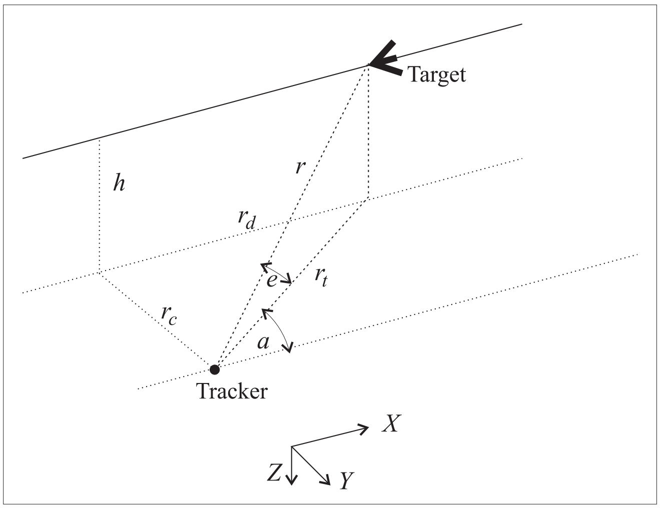
Kalman filters may be extended to some special non-linear system formulations. This involves some form of linearization. The validity of the resultant linearization give rise to new problems such as instability (the linear case was always stable) and stagnation (the state and gains gets stuck).

The following linearization possibilities exist:

- Input linearization. In this case the signals driving the Kalman filter is non-linear, but the dynamics of the filter is still linear. The Kalman filter behaves as in the linear case, but the inputs and noise variances change with current setpoint. A typical example is in the radar tracker formulation, where the signals are measured in Euler axis (traverse and elevation angles and range — the measurements are then transformed to new carthesian coordinates before filtering starts.
- A similar case happens when the output of the Kalman filter is non-linear.
- In the case where the dynamics of the problem is non-linear, the linearization must be done inside the Kalman filter. This form of filter is called the Extended Kalman Filter (EKF).

2.1 Kalman filter with Input Linearization

Consider the following tracking geometry

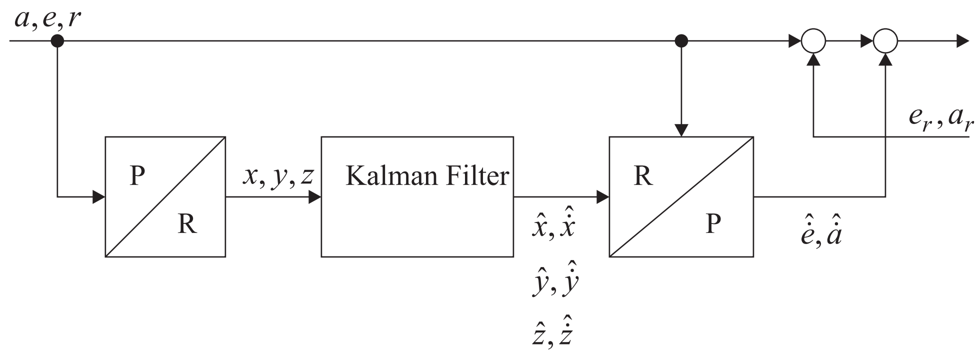


With our defined axis system, we have

$$x = r_d = r\cos a \cos e; \quad y = -r_c = -r\sin a \cos e; \quad z = -h = -r\sin e; \quad \dot{x} = -U; \quad \dot{y} = \dot{z} = 0$$

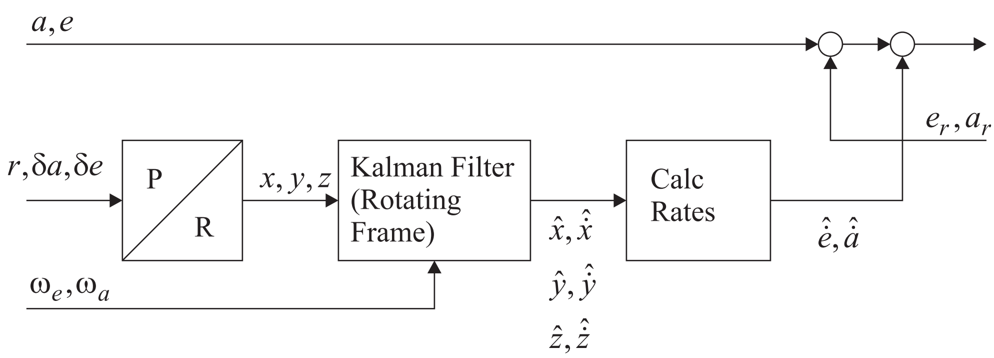
The measurement noise on the range r is now modified and added into the three axes according to the above formulas for different positions of x_d, which changes the angle values of e and a.

To provide a rate-aided tracker, one implementation of the feedforward rate would be



In this case, the Kalman filter implementation is simple and follows the cartesian formulation as per [1]. The rates are calculated by transforming the velocities in the fixed XYZ plane to the rotating tracker frame to calculate the feedforward velocities.

The paper by Fitts in [4] provides a complete reference to high-accuracy tracking using aided Kalman filters. A modern implementation uses a strapped-down approach by pulling the polar to reactangular and reactangular to polar conversions into the Kalman filter as described in [4]. The new block diagram would be



The Kalman filter is now a two-step process: (1) Filter and (2) Incremental rotation. The last operation may be performed multiple times for each filter step — this helps to keep the filter linear.

2.2 Extended Kalman Filter

Given the system description

$$\begin{aligned} \dot{x} &= f(x, u) \\ y &= g(x) \end{aligned}$$

we can linearize the system around the workpoint

$$\dot{x}(w) = Ax(w) + Bu(w), \quad y = Cx(w)$$

with the Jacobian (partial derivative matrix)

$$
\begin{aligned}
\mathbf{A} &= \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{x(w)} \\
\mathbf{B} &= \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \frac{\partial f_n}{\partial u_2} & \cdots & \frac{\partial f_n}{\partial u_m} \end{bmatrix}_{u(w)} \\
\mathbf{C} &= \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial g_k}{\partial u_1} & \frac{\partial g_k}{\partial x_2} & \cdots & \frac{\partial g_k}{\partial x_n} \end{bmatrix}_{x(w)}
\end{aligned}
$$

Example In a satellite attitude control application, we would typically include in our state vector the attitude quaternion (at least components $q_1 \dots q_3$). The new attitude is given by the output equation

$$\hat{v} = A(\hat{q})v_r$$

Using the standard (3-2-1) Euler set with first rotation in yaw by ψ, the second in pitch by θ and the third in roll by ϕ, the direction cosine matrix A from the reference axis system to the new displaced axis system becomes (first with Euler angles and then with quaternions)

$$
\begin{aligned}
A &= \begin{bmatrix} C\theta C\psi & C\theta S\psi & -S\theta \\ S\phi S\theta C\psi - C\phi S\psi & S\phi S\theta S\psi + C\phi C\psi & S\phi C\theta \\ C\phi S\theta C\psi + S\phi S\psi & C\phi S\theta S\psi - S\phi C\psi & C\phi C\theta \end{bmatrix} \\
&= \begin{bmatrix} q_4^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_4 q_3 + q_1 q_2) & 2(q_1 q_3 - q_4 q_2) \\ 2(q_1 q_2 - q_4 q_3) & q_4^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_4 q_1 + q_2 q_3) \\ 2(q_4 q_2 + q_1 q_3) & 2(q_2 q_3 - q_4 q_1) & q_4^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}
\end{aligned}
$$

The output matrix $\mathbf{C}$ now becomes the sensitivity of the direction cosine matrix (actually the error quaternion from the previous value!)

$$c_i = \frac{\partial A(q)}{\partial q_i}\,\hat{v}$$

with submatrices

$$
c_1 = 2\begin{bmatrix} q_1 & q_2 & q_3 \\ q_2 & -q_1 & q_4 \\ q_3 & -q_4 & -q_1 \end{bmatrix}\hat{v}, \quad c_2 = 2\begin{bmatrix} -q_2 & q_1 & -q_4 \\ q_1 & q_2 & q_3 \\ q_4 & q_3 & -q_2 \end{bmatrix}\hat{v}
$$

$$
c_3 = 2\begin{bmatrix} -q_3 & q_4 & q_1 \\ -q_4 & -q_3 & q_2 \\ q_1 & q_2 & q_3 \end{bmatrix}\hat{v}, \quad c_4 = 2\begin{bmatrix} q_4 & q_3 & -q_2 \\ -q_3 & q_4 & q_1 \\ q_2 & -q_1 & q_4 \end{bmatrix}\hat{v}
$$

and

$$\mathbf{C}\,\hat{\mathbf{v}} = \left[\begin{array}{cccc} c_1 & c_2 & c_3 & c_4 \end{array}\right]$$

A more natural derivation of the C-matrix comes from expanding the innovation equation relative to the error quaternion δq. For the normal summing innvovation (measurement - base vector) we have

$$\mathbf{i}_m = \hat{\mathbf{v}}_m - \mathbf{v}_b = \hat{\mathbf{v}}_m - \mathbf{A}(\hat{q})\tilde{\mathbf{v}}_r = \mathbf{A}(\delta q)\mathbf{A}(\hat{q})\mathbf{v}_r - \mathbf{A}(\hat{q})\tilde{\mathbf{v}}_r$$

Rewriting in terms of body axis

$$\mathbf{i}_m = \mathbf{A}(\delta q)\mathbf{v}_b - \tilde{\mathbf{v}}_b \approx \left[\mathbf{A}(\delta q) - I\right]\mathbf{v}_b$$

The transformation of δq is a simplified version with $\delta q_4 \approx 1$, therefore

$$\mathbf{A}(\delta q) \approx \left[\begin{array}{ccc} 1 & 2\delta q_3 & -2\delta q_2 \\ -2\delta q_3 & 1 & 2\delta q_1 \\ 2\delta q_2 & -2\delta q_1 & 1 \end{array}\right]$$

Now expanding the right-hand side

$$\begin{aligned}
\left[\mathbf{A}(\delta q) - I\right]\hat{\mathbf{v}}_b(k) &= \left[\begin{array}{ccc} 0 & 2\delta q_3 & -2\delta q_2 \\ -2\delta q_3 & 0 & 2\delta q_1 \\ 2\delta q_2 & -2\delta q_1 & 0 \end{array}\right]\left[\begin{array}{c} \hat{v}_{bx} \\ \hat{v}_{by} \\ \hat{v}_{bz} \end{array}\right] \\
&= \left[\begin{array}{c} 2\delta q_3\hat{v}_{by} - 2\delta q_2\hat{v}_{bz} \\ -2\delta q_3\hat{v}_{bx} + 2\delta q_1\hat{v}_{bz} \\ 2\delta q_2\hat{v}_{bx} - 2\delta q_1\hat{v}_{by} \end{array}\right] \\
&= 2\left[\begin{array}{ccc} 0 & -\hat{v}_{bz} & \hat{v}_{by} \\ \hat{v}_{bz} & 0 & -\hat{v}_{bx} \\ -\hat{v}_{by} & \hat{v}_{bx} & 0 \end{array}\right]\left[\begin{array}{c} \delta q_1 \\ \delta q_2 \\ \delta q_3 \end{array}\right] \\
&= \mathbf{C}(\hat{\mathbf{v}}_b)\delta q
\end{aligned}$$

with

$$\mathbf{C}(\hat{\mathbf{v}}_b) = 2\left[\begin{array}{ccc} 0 & -\hat{v}_{bz} & \hat{v}_{by} \\ \hat{v}_{bz} & 0 & -\hat{v}_{bx} \\ -\hat{v}_{by} & \hat{v}_{bx} & 0 \end{array}\right]$$

For the MEKF case where the innovation is given by the crossproduct of the measurement and body vectors, the analysis is similar.

Let the innovation $\mathbf{i}_m(k)$ be given by the cross-product of the measurement $\mathbf{v}_m(k)$ and the reference vector $\hat{\mathbf{v}}_b(k)$ in body coordinates

$$\mathbf{i}_m(k) = \mathbf{v}_m(k) \times \hat{\mathbf{v}}_b(k) = \mathbf{v}_m(k) \times \mathbf{A}(\hat{q})\hat{\mathbf{v}}_r(k)$$

Using the MEKF strategy of $\mathbf{A}(q) = \mathbf{A}(\delta q)\mathbf{A}(\hat{q})$ and expanding $\mathbf{v}_m(k)$ to

$$\mathbf{v}_m(k) = \mathbf{A}(q)\mathbf{v}_r(k) = \mathbf{A}(\delta q)\mathbf{A}(\hat{q})\mathbf{v}_r(k) = \mathbf{A}(\delta q)\hat{\mathbf{v}}_b(k)$$

Now rewrite the innovation to

$$\mathbf{i}_m(k) = \mathbf{A}(\delta q)\hat{\mathbf{v}}_b(k) \times \hat{\mathbf{v}}_b(k) = \mathbf{C}(\hat{\mathbf{v}}_b)\delta q'$$

with the quaternion

The transformation of δq is a simplified version with $\delta q_4 \approx 1$, therefore

$$\mathbf{A}(\delta q) \approx \begin{bmatrix} 1 & 2\delta q_3 & -2\delta q_2 \\ -2\delta q_3 & 1 & 2\delta q_1 \\ 2\delta q_2 & -2\delta q_1 & 1 \end{bmatrix}$$

Now expanding the left side

$$\mathbf{A}(\delta q)\hat{\mathbf{v}}_b(k) = \begin{bmatrix} 1 & 2\delta q_3 & -2\delta q_2 \\ -2\delta q_3 & 1 & 2\delta q_1 \\ 2\delta q_2 & -2\delta q_1 & 1 \end{bmatrix} \begin{bmatrix} \hat{v}_{bx} \\ \hat{v}_{by} \\ \hat{v}_{bz} \end{bmatrix} = \begin{bmatrix} \hat{v}_{bx} + 2\delta q_3 \hat{v}_{by} - 2\delta q_2 \hat{v}_{bz} \\ \hat{v}_{by} - 2\delta q_3 \hat{v}_{bx} + 2\delta q_1 \hat{v}_{bz} \\ \hat{v}_{bz} + 2\delta q_2 \hat{v}_{bx} - 2\delta q_1 \hat{v}_{by} \end{bmatrix} = \begin{bmatrix} \alpha_x \\ \alpha_y \\ \alpha_z \end{bmatrix}$$

and

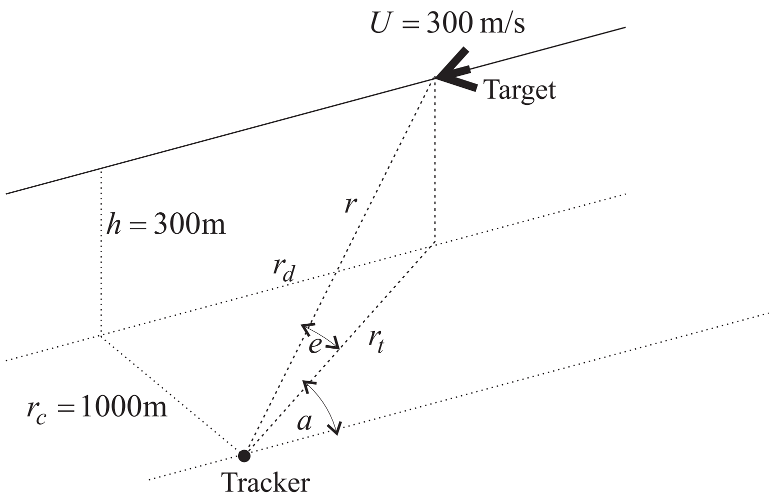
$$\begin{aligned}
\mathbf{A}(\delta q)\hat{\mathbf{v}}_b(k) \times \hat{\mathbf{v}}_b(k) &= \begin{bmatrix} \alpha_y \hat{v}_{bz} - \alpha_z \hat{v}_{by} \\ \alpha_z \hat{v}_{bx} - \alpha_z \hat{v}_{bz} \\ \alpha_x \hat{v}_{by} - \alpha_y \hat{v}_{bx} \end{bmatrix} \\
&= \begin{bmatrix} (\hat{v}_{by} - 2\delta q_3 \hat{v}_{bx} + 2\delta q_1 \hat{v}_{bz})\hat{v}_{bz} - (\hat{v}_{bz} + 2\delta q_2 \hat{v}_{bx} - 2\delta q_1 \hat{v}_{by})\hat{v}_{by} \\ (\hat{v}_{bz} + 2\delta q_2 \hat{v}_{bx} - 2\delta q_1 \hat{v}_{by})\hat{v}_{bx} - (\hat{v}_{bx} + 2\delta q_3 \hat{v}_{by} - 2\delta q_2 \hat{v}_{bz})\hat{v}_{bz} \\ (\hat{v}_{bx} + 2\delta q_3 \hat{v}_{by} - 2\delta q_2 \hat{v}_{bz})\hat{v}_{by} - (\hat{v}_{by} - 2\delta q_3 \hat{v}_{bx} + 2\delta q_1 \hat{v}_{bz})\hat{v}_{bx} \end{bmatrix} \\
&= 2 \begin{bmatrix} -\delta q_3 \hat{v}_{bx}\hat{v}_{bz} + \delta q_1 \hat{v}_{bz}^2 - \delta q_2 \hat{v}_{bx}\hat{v}_{by} + \delta q_1 \hat{v}_{by}^2 \\ \delta q_2 \hat{v}_{bx}^2 - \delta q_1 \hat{v}_{by}\hat{v}_{bx} - \delta q_3 \hat{v}_{by}\hat{v}_{bz} + \delta q_2 \hat{v}_{bz}^2 \\ \delta q_3 \hat{v}_{by}^2 - \delta q_2 \hat{v}_{bz}\hat{v}_{by} + \delta q_3 \hat{v}_{bx}^2 - \delta q_1 \hat{v}_{bz}\hat{v}_{bx} \end{bmatrix} \\
&= 2 \begin{bmatrix} \hat{v}_{bz}^2 + \hat{v}_{by}^2 & -\hat{v}_{bx}\hat{v}_{by} & -\hat{v}_{bx}\hat{v}_{bz} \\ -\hat{v}_{by}\hat{v}_{bx} & \hat{v}_{bx}^2 + \hat{v}_{bz}^2 & -\hat{v}_{by}\hat{v}_{bz} \\ -\hat{v}_{bz}\hat{v}_{bx} & -\hat{v}_{bz}\hat{v}_{by} & \hat{v}_{by}^2 + \hat{v}_{bx}^2 \end{bmatrix} \begin{bmatrix} \delta q_1 \\ \delta q_2 \\ \delta q_3 \end{bmatrix} \\
&= \mathbf{C}(\hat{\mathbf{v}}_b)\delta q
\end{aligned}$$

with

$$\mathbf{C}(\hat{\mathbf{v}}_b) = 2 \begin{bmatrix} \hat{v}_{bz}^2 + \hat{v}_{by}^2 & -\hat{v}_{bx}\hat{v}_{by} & -\hat{v}_{bx}\hat{v}_{bz} \\ -\hat{v}_{by}\hat{v}_{bx} & \hat{v}_{bx}^2 + \hat{v}_{bz}^2 & -\hat{v}_{by}\hat{v}_{bz} \\ -\hat{v}_{bz}\hat{v}_{bx} & -\hat{v}_{bz}\hat{v}_{by} & \hat{v}_{by}^2 + \hat{v}_{bx}^2 \end{bmatrix}$$

3 Task 1

Given the geometry of the following aircraft tracking scenario and with the tracker consisting of a outer yaw axis (angle a) and inner elevation axis (angle e) shown below:



The downranges r_d of interest are $1000 \leq r_d \leq 4000$ m. The yaw tracker plant dynamics are given by

$$\frac{a}{u} = \frac{\omega_a^2}{s(s^2 + \omega_a s + \omega_a^2)}, \quad \omega_a = 4\pi \text{ rad/s } (2\,\text{Hz})$$

Develop:

1. Basic yaw-axis continuous tracking loop controller.
   (a) Design a simple tracking loop controller.
   (b) The dynamic equations of the yaw and elevation axis up to the accelaration terms
   (c) The dynamic error coeficients for the yaw axis using the designed controller
   (d) A Simulink model of the yaw axis
   (e) Verify the simulated errors with the calculated errors
2. A rate-aided tracker for the yaw axis
   (a) Design a non-linear input Kalman filter in the range and yaw axes. Preferably, design your KF to rotate with the line-of-sight. Assume the following target characteristics: (1) lateral acceleration < 5 g; (2) time constant 0.2 sec; (3) range measurement error 5 m; and (4) angle measurement error 1 mrad.
   (b) Calculate a rate feedforward from the Kalman filter output and compare the simulated rate feedforward with the simulated tracker error.
   (c) Integrate the rate feedforward with the yaw axis tracker and compare the results with angle-only tracker.

References

[1] *Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets*, Robert A Singer, IEEE Transactions on Aerospace and Electronic Systems, AES-6, Vol. 4, 1970.

[2] *Real-Time Tracking Filter Evaluation and Selection for Tactical Applications*, Robert A Singer, IEEE Transactions on Aerospace and Electronic Systems, AES-7, Vol. 1, 1971.

[3] *Closed form solutions to ECV and ECA tracking filters based on a transfer function approach*, K Watanabe, IEE Procedings, Vol. 134, no. 6, pages 368–372, 1987.

[4] *Aided Tracking as Applied to High-Accuracy Pointing Systems*, JM Fitts, IEEE Transactions on Aerospace and Electronic Systems, AES-9, Vol. 13, 1973.