A reminder of steady-state Kalman filtering

- In Lesson 2.3.5, you learned that under common conditions, the Kalman-filter covariances converge to steady-state solutions as does the Kalman gain.
  - If we are willing to accept slightly suboptimal state estimates, we can implement a steady-state Kalman filter that uses constant L_{ss} instead of an optimal Kalman filter that uses time-varying L_k.
- The benefit of doing so is a huge reduction in computation.
  - L_{ss} is computed once only, and the state-estimate-update equations are streamlined.
- To compute L_{ss}, we needed to solve a discrete algebraic Riccati equation (DARE), which is somewhat complicated, but is facilitated using `dlqe` in Octave.

A steady-state Kalman filter for target tracking

- For a steady-state solution to L_{ss} to exist, we must have constant A, B, C, D, and noise-covariance matrices.
- When these conditions are met, there is still no general closed-form solution to computing L_{ss}.
  - However, we can find a closed-form solution for the NCV model, producing an optimized α–β filter.
  - We can also find a closed-form solution for the NCA model (to be introduced), producing an optimized α–β–γ filter.
  - Since these models are commonly used for target-tracking applications, we consider steady-state forms of Kalman filters based on them in this lesson.
- In this lesson, you will learn how to derive the α–β filter and will also learn the result for the α–β–γ filter.

The generic α–β filter

- An general α–β filter assumes an NCV model of the form:

$$\begin{aligned} p_{k+1} &= p_k + (\Delta t) v_k + \text{noise} \\ v_{k+1} &= v_k + \text{noise}. \end{aligned}$$

- The update equations for the general α–β filter are of the form:

$$\begin{aligned} \hat{p}_{k+1} &= \hat{p}_k + (\Delta t)\hat{v}_k + (\alpha)\left(z_k - \hat{p}_k\right) \\ \hat{v}_{k+1} &= \hat{v}_k + (\beta/\Delta t)\left(z_k - \hat{p}_k\right). \end{aligned}$$

- There exists a volume of literature that proposes different ways to select α and β.
- In this lesson, you will learn how to optimize the values of α and β for this method using a closed-form steady-state Kalman filter.
- While the resulting KF is suboptimal because it is not using time-varying gains, the values of α and β are optimal since the α–β filter assumes constant gains.

Review of the NCV model in state-space form

- To show how to derive the optimized α and β values, recall that we can write the NCV model as:

$$x_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} (\Delta t)^2/2 \\ \Delta t \end{bmatrix} w'_k$$
$$z_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k,$$

where $w'_k \sim \mathcal{N}(0, \sigma^2_{\widetilde{w}})$ and $v_k \sim \mathcal{N}(0, \Sigma_{\tilde{v}})$, and where w'_k is a scalar.

- We can rewrite the state equation as:

$$x_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} x_k + w_k,$$

where $w_k \sim \mathcal{N}(0, \Sigma_{\widetilde{w}})$ and:

$$\Sigma_{\widetilde{w}} = \begin{bmatrix} (\Delta t)^2/2 \\ \Delta t \end{bmatrix} \mathbb{E}\left[w'_k (w'_k)^T\right] \begin{bmatrix} (\Delta t)^2/2 & \Delta t \end{bmatrix} = \begin{bmatrix} (\Delta t)^4/4 & (\Delta t)^3/2 \\ (\Delta t)^3/2 & (\Delta t)^2 \end{bmatrix} \sigma^2_{\widetilde{w}}.$$

Solving for L_{ss}

- We are now ready to solve for L_{ss}: we denote the components of the steady-state solution as:

$$L_{ss} = \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta/\Delta t \end{bmatrix}, \quad \Sigma^-_{\tilde{x},ss} = \begin{bmatrix} \Sigma^-_{11} & \Sigma^-_{12} \\ \Sigma^-_{12} & \Sigma^-_{22} \end{bmatrix}, \quad \Sigma^+_{\tilde{x},ss} = \begin{bmatrix} \Sigma^+_{11} & \Sigma^+_{12} \\ \Sigma^+_{12} & \Sigma^+_{22} \end{bmatrix}.$$

- Using this notation, we can write the Kalman gain as:

$$\begin{aligned} L_{ss} &= \Sigma^-_{\tilde{x},ss} C^T \left(C \Sigma^-_{\tilde{x},ss} C^T + \Sigma_{\tilde{v}}\right)^{-1} \\ &= \begin{bmatrix} \Sigma^-_{11} & \Sigma^-_{12} \\ \Sigma^-_{12} & \Sigma^-_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \left(\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \Sigma^-_{11} & \Sigma^-_{12} \\ \Sigma^-_{12} & \Sigma^-_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \Sigma_{\tilde{v}} \right)^{-1} \\ &= \frac{1}{\Sigma^-_{11} + \Sigma_{\tilde{v}}} \begin{bmatrix} \Sigma^-_{11} \\ \Sigma^-_{12} \end{bmatrix}. \end{aligned}$$

- So, if we can solve for $\Sigma^-_{\tilde{x},ss}$, we can find L_{ss} (and therefore α and β).

Solution #1 for Σ^+_{ss}

- To find the steady-state covariance matrices, we first write (using the estimation-error covariance-matrix measurement-update equation):

$$\begin{aligned} \Sigma^+_{\tilde{x},ss} &= (I - L_{ss} C) \Sigma^-_{\tilde{x},ss} \\ &= \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \right) \begin{bmatrix} \Sigma^-_{11} & \Sigma^-_{12} \\ \Sigma^-_{12} & \Sigma^-_{22} \end{bmatrix} \\ &= \begin{bmatrix} (1 - L_1)\Sigma^-_{11} & (1 - L_1)\Sigma^-_{12} \\ (1 - L_1)\Sigma^-_{12} & \Sigma^-_{22} - L_2 \Sigma^-_{12} \end{bmatrix}. \end{aligned}$$

- We call this “solution 1” for $\Sigma^+_{\tilde{x},ss}$, which we will use in just a minute.

Solution #2 for Σ^+_{ss}

- We continue by finding another relationship for $\Sigma^+_{\tilde{x},ss}$. Recall the prediction-error covariance-matrix time-update equation:

$$\Sigma^-_{\tilde{x},ss} = A\Sigma^+_{\tilde{x},ss}A^T + \Sigma_{\widetilde{w}}.$$

- We rewrite this equation in terms of $\Sigma^+_{\tilde{x},ss}$:

$$\begin{aligned}\Sigma^+_{\tilde{x},ss} &= A^{-1}(\Sigma^-_{\tilde{x},ss} - \Sigma_{\widetilde{w}})A^{-T}\\ &= \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix}\left(\begin{bmatrix} \Sigma^-_{11} & \Sigma^-_{12} \\ \Sigma^-_{12} & \Sigma^-_{22} \end{bmatrix} - \begin{bmatrix} (\Delta t)^4/4 & (\Delta t)^3/2 \\ (\Delta t)^3/2 & (\Delta t)^2 \end{bmatrix}\sigma^2_{\widetilde{w}}\right)\begin{bmatrix} 1 & 0 \\ -\Delta t & 1 \end{bmatrix}.\end{aligned}$$

- We can write this out and find all the terms for $\Sigma^+_{\tilde{x},ss}$.
- We call this "solution 2" for $\Sigma^+_{\tilde{x},ss}$.

Final α–β filter solution

- When we equate the terms of "solution 1" and "solution 2" for $\Sigma^+_{\tilde{x},ss}$ and do some algebra, we arrive at the solution:

$$L_{ss} = \begin{bmatrix} -\frac{1}{8}\left(\lambda^2 + 8\lambda - (\lambda+4)\sqrt{\lambda^2+8\lambda}\right) \\ \frac{1}{4\Delta t}\left(\lambda^2 + 4\lambda - \lambda\sqrt{\lambda^2+8\lambda}\right) \end{bmatrix}$$

$$\Sigma^+_{\tilde{x},ss} = \begin{bmatrix} L_1\Sigma_{\widetilde{v}} & L_2\Sigma_{\widetilde{v}} \\ L_2\Sigma_{\widetilde{v}} & \frac{L_2\Sigma_{\widetilde{v}}}{1-L_1}\left(\frac{L_1}{\Delta t} - \frac{L_2}{2}\right) \end{bmatrix},$$

where:

$$\lambda = \frac{\sigma_{\widetilde{w}}(\Delta t)^2}{\sqrt{\Sigma_{\widetilde{v}}}}.$$

- The term λ is the "target-maneuvering index" or "target-tracking index" and is proportional to the ratio of the motion uncertainty to the measurement uncertainty.

Compare Octave `dlqe` with α–β solution

```
dT = 0.3;       % A generic sample period
sigmaW = 0.1;   % Standard deviation of (scalar) process noise
SigmaV = 0.2;   % Covariance of measurement noise

% Compute the DLQE solution
Ad = [1 dT; 0 1]; Cd = [1 0]; % Specify NCV matrices and compute process-noise
SigmaW = [(dT)^4/4 (dT)^3/2; (dT)^3/2 (dT)^2]*sigmaW^2; % vector covariance
[Lss1,SigmaMss1,SigmaPss1] = dlqe(Ad,eye(2),Cd,SigmaW,SigmaV);

% Compute the alpha-beta solution
lambda = sigmaW * (dT)^2 / sqrt(SigmaV);
Lss2 = [-(lambda^2+8*lambda-(lambda+4)*sqrt(lambda^2+8*lambda))/8; ...
       (lambda^2+4*lambda-lambda*sqrt(lambda^2+8*lambda))/(4*dT)];
SigmaMss2 = [Lss2(1)*SigmaV/(1-Lss2(1)), Lss2(2)*SigmaV/(1-Lss2(1)); Lss2(2)* ...
  SigmaV/(1-Lss2(1)), (Lss2(1)/dT + Lss2(2)/2)*Lss2(2)*SigmaV/(1-Lss2(1))];
SigmaPss2 = [Lss2(1)*SigmaV, Lss2(2)*SigmaV; Lss2(2)*SigmaV, ...
  Lss2(2)*SigmaV*(Lss2(1)/dT -Lss2(2)/2)/(1-Lss2(1))];

Lss1, Lss2              % They agree!
SigmaMss1, SigmaMss2    % These also agree!
SigmaPss1, SigmaPss2    % These agree as well!
```

The NCA model used by the α–β–γ filter

- The NCA model is also popular for tracking. For scalar w'_k:

$$x_{k+1} = \begin{bmatrix} 1 & \Delta t & (\Delta t)^2/2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} (\Delta t)^2/2 \\ \Delta t \\ 1 \end{bmatrix} w'_k$$

$$z_k = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x_k + v_k.$$

- We can rewrite the state equation for vector $w_k \sim \mathcal{N}(0, \Sigma_{\tilde{w}})$ as:

$$x_{k+1} = \begin{bmatrix} 1 & \Delta t & (\Delta t)^2/2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} x_k + w_k, \quad \text{where}$$

$$\Sigma_{\tilde{w}} = \begin{bmatrix} (\Delta t)^4/4 & (\Delta t)^3/2 & (\Delta t)^2/2 \\ (\Delta t)^3/2 & (\Delta t)^2 & \Delta t \\ (\Delta t)^2/2 & \Delta t & 1 \end{bmatrix} \sigma^2_{\tilde{w}}.$$

The α–β–γ solution framework

- We denote the components of the steady-state solution as:

$$L_{ss} = \begin{bmatrix} \alpha \\ \beta/\Delta t \\ \gamma/(2(\Delta t)^2) \end{bmatrix}, \quad \Sigma^+_{\tilde{x},ss} = \begin{bmatrix} \Sigma^+_{11} & \Sigma^+_{12} & \Sigma^+_{13} \\ \Sigma^+_{12} & \Sigma^+_{22} & \Sigma^+_{23} \\ \Sigma^+_{13} & \Sigma^+_{23} & \Sigma^+_{33} \end{bmatrix}.$$

- The solution is computed via some auxiliary variables (λ is the same as earlier):

$$b = \frac{\lambda}{2} - 3 \qquad c = \frac{\lambda}{2} + 3$$

$$p = c - \frac{b^2}{3} \qquad q = \frac{2b^3}{27} - \frac{bc}{3} - 1$$

$$z = \left[\frac{-q + \sqrt{q^2 + 4p^3/27}}{2}\right]^{1/3} \qquad s = z - \frac{p}{3z} - \frac{b}{3}.$$

The α–β–γ filter solution

- The details for using these variables to find a solution are straightforward but tedious; I omit them here.
- For reference, the final answer is:

$$\alpha = 1 - s^2, \qquad \Sigma^+_{11} = \alpha \Sigma_{\tilde{v}}, \qquad \Sigma^+_{22} = \frac{8\alpha\beta + \gamma(\beta - 2\alpha - 4)}{8(\Delta t)^2(1-\alpha)}\Sigma_{\tilde{v}}$$

$$\beta = 2(1-s)^2, \qquad \Sigma^+_{12} = \beta\Sigma_{\tilde{v}}/\Delta t, \qquad \Sigma^+_{23} = \frac{\beta(2\beta - \gamma)}{4(\Delta t)^3(1-\alpha)}\Sigma_{\tilde{v}}$$

$$\gamma = 2\lambda s, \qquad \Sigma^+_{13} = \gamma\Sigma_{\tilde{v}}/(2(\Delta t)^2), \qquad \Sigma^+_{33} = \frac{\gamma(2\beta - \gamma)}{4(\Delta t)^4(1-\alpha)}\Sigma_{\tilde{v}}.$$

Summary

- Target-tracking applications often use simplified dynamic models such as NCV and NCA.
- Closed-form solutions exist for steady-state Kalman filters for these two models.
- They are optimized versions of α–β and α–β–γ filters, respectively.
- In this lesson, you learned how the α–β filter is derived, and compared its solution to that computed by dlqe for the same model (they matched!)
- You also learned the solution to the α–β–γ filter.
- The big advantage of knowing these solutions is that they can be computed using standard algebra, without requiring an algebraic-Riccati-equation solver.