Real-world issue: Initialization

- A Kalman filter must be initialized by specifying values for:

$$\hat{x}_0^+ = \mathbb{E}[x_0], \quad \text{and} \quad \Sigma_{\tilde{x},0}^+ = \mathbb{E}[(x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^T].$$

- But, we rarely know what these expectations should be and we can't measure them.
- So, we are left with two options:
  1. Guess. This often works better than you might think since the linear Kalman filter converges toward the truth even when the initialization is done poorly.
     - Still, we want to use as much domain knowledge as possible when "guessing."
  2. Make a some measurements of the input and output "before time zero" and somehow fuse them together to make an estimate of $\hat{x}_0^+$ and $\Sigma_{\tilde{x},0}^+$.[1]
     - Good idea, but the math is very complicated and the benefits are often small.

[1] For example, see: X. Rong Li and Chen He, "Optimal Initialization of Linear Recursive Filters," in *Proc. 37th IEEE Conference on Decision and Control*, Tampa, FL, 1998, pp. 2335–2340.

Real-world issue: Tuning $\Sigma_{\tilde{x},0}$

- Specifying $\hat{x}_0^+$ and $\Sigma_{\tilde{x},0}^+$ is part of the process of "tuning" a KF.
- Guessing $\hat{x}_0^+$ is more likely to be intuitive than guessing $\Sigma_{\tilde{x},0}^+$.
- Ideally, $\Sigma_{\tilde{x},0} = \mathbb{E}[(x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^T]$.
- Since we don't know x_0 exactly, not possible to initialize $\hat{x}_0^+$ to exactly correct value: Uncertainty in $\hat{x}_0^+$ is captured by $\Sigma_{\tilde{x},0}$.
- If we assume that model states uncorrelated, $\Sigma_{\tilde{x},0}$ is diagonal.
  - We often make this assumption simply because we have no better information regarding state correlation.
  - Diagonal elements in $\Sigma_{\tilde{x},0}$ are variances of initial estimation errors of the corresponding state elements in x.
- Set the diagonal elements of $\Sigma_{\tilde{x},0}$ such that you expect the true state to lie between $\hat{x}_0^+ \pm 3\sqrt{\text{diag}(\Sigma_{\tilde{x},0})}$.

Real-world issue: Tuning $\Sigma_{\widetilde{w}}$

- Assuming that the model of the system is perfect, $\Sigma_{\widetilde{w}}$ is the covariance matrix of process noise w_k driving the state:

$$x_{k+1} = Ax_k + Bu_k + w_k.$$

- In the KF development, process noise is any unmeasured (zero-mean, white) input that affects the state vector.
- In some cases, such as when w_k models sensing error on u_k, $\Sigma_{\widetilde{w}}$ can be determined statistically via experimentation.
- But, as no model is perfect, $\Sigma_{\widetilde{w}}$ also attempts to capture—in some way—state-equation inaccuracies, so we should specify larger uncertainty via $\Sigma_{\widetilde{w}}$ than simply representing input-sensor noise alone.
- We can hand-tune $\Sigma_{\widetilde{w}}$ to account for uncertainties in A and B, to an extent.
  - Run the KF on a dataset using different $\Sigma_{\widetilde{w}}$ until you achieve desired performance.

Real-world issue: Tuning $\Sigma_{\tilde{v}}$

- Assuming that the model of the system is perfect, $\Sigma_{\tilde{v}}$ is the covariance matrix of measurement noise v_k:

$$z_k = Cx_k + Du_k + v_k.$$

- Measurement noise is any unmeasured (zero-mean, white) input that doesn't affect the state vector, but which does corrupt measurements.
- Since $\Sigma_{\tilde{v}}$ is a property of the measurement system (which we design, or which we can exercise), it can be determined statistically via experimentation (or data sheet).
- But, as no model is perfect, $\Sigma_{\tilde{v}}$ also attempts to capture—in some way—output-equation inaccuracies, so we should specify larger uncertainty via $\Sigma_{\tilde{v}}$ than simply representing sensor noise alone.
- We can hand-tune $\Sigma_{\tilde{v}}$ to account for uncertainties in C and D, to an extent.
  - Run the KF on a dataset using different $\Sigma_{\tilde{v}}$ until you achieve desired performance.

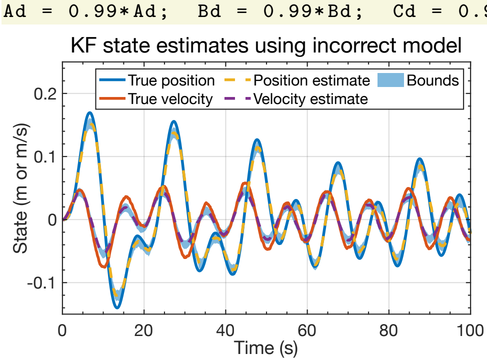
Rate of convergence

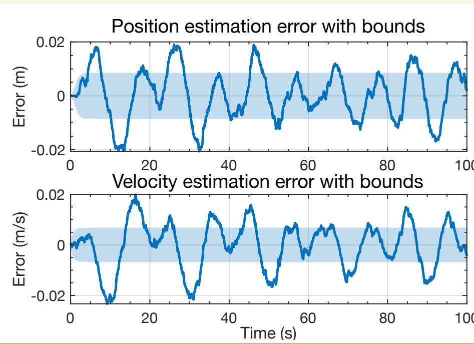
- Filter convergence rates are determined by $\Sigma_{\tilde{w}}$ and $\Sigma_{\tilde{v}}$:
  - Large $\Sigma_{\tilde{w}}$ says "trust the sensor more than the model," and makes state error bounds large, but convergence faster.
  - Large $\Sigma_{\tilde{v}}$ says "trust the model more than the sensor," which makes error bounds narrower but convergence slower (pseudo open loop).
- Since model inaccuracies are difficult to quantify, some trial-and-error "tuning" of $\Sigma_{\tilde{w}}$ and $\Sigma_{\tilde{v}}$ is common and these "rules of thumb" can sometimes fail.
- In some cases (noisy sensors or bad model), desired state-estimate accuracy and convergence rates are simply impossible.
- In general, it is not possible to have arbitrarily fast convergence to arbitrarily narrow error bounds.

Review of KF performance for incorrect model

- In Lesson 1.4.5, you learned that KF estimation performance degrades considerably when the system model is wrong.
- We intentionally mis-modeled A, B, and C, resulting in poor KF estimates.

```
% Force the KF to use different Ad, Bd, Cd from the model simulation:
Ad = 0.99*Ad;  Bd = 0.99*Bd;  Cd = 0.99*Cd;
```

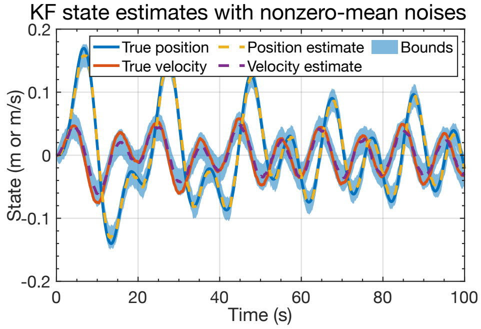


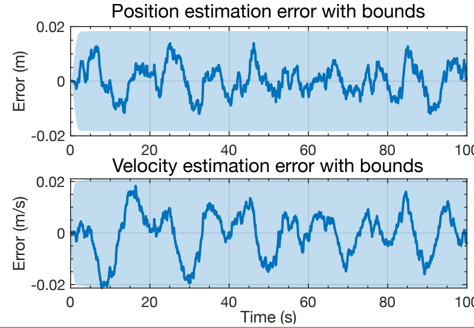


Compensating for bad model using $\Sigma_{\tilde{w}}$ and $\Sigma_{\tilde{v}}$

- Now, we seek to "tune" $\Sigma_{\tilde{w}}$ and $\Sigma_{\tilde{v}}$ to see if they can (at least partially) compensate for an incorrect model.
- After some trial and error, we arrive at a decent compromise:

```
SigmaW = SigmaW*diag([1 12]);
SigmaV = SigmaV*3;
```





Summary

- You have learned how to set $\Sigma_{\tilde{x},0}$ during initialization.
- You have seen tips on tuning $\Sigma_{\tilde{w}}$, which represents process-noise uncertainty (and sometimes inaccuracies in the state equation).
- You have seen some tips on tuning $\Sigma_{\tilde{v}}$, which represents sensor-noise uncertainty (and sometimes inaccuracies in the measurement equation).
- Since model inaccuracies are difficult to quantify, some trial-and-error "tuning" of $\Sigma_{\tilde{w}}$ and $\Sigma_{\tilde{v}}$ is common.
  - In course 3, "Nonlinear Kalman Filters," you will learn about some kinds of "adaptive" Kalman filters that seek to adapt $\Sigma_{\tilde{w}}$ and $\Sigma_{\tilde{v}}$ along with the state estimate. These can avoid some of the challenges of trial-and-error tuning.
- Since convergence rates depend on $\Sigma_{\tilde{w}}$ and $\Sigma_{\tilde{v}}$, it is not possible in general to have arbitrarily fast convergence to arbitrarily narrow error bounds.
- Still, KF is optimal MMSE estimator for the assumptions made during derivation.

Credits

Credits for photos in this lesson

- Tuning fork on slide 2: By Helihark (own work) [CC BY-SA 3.0 (`http://creativecommons.org/licenses/by-sa/3.0`)], via Wikimedia Commons, `https://commons.wikimedia.org/wiki/File:Tuning-fork.jpg`