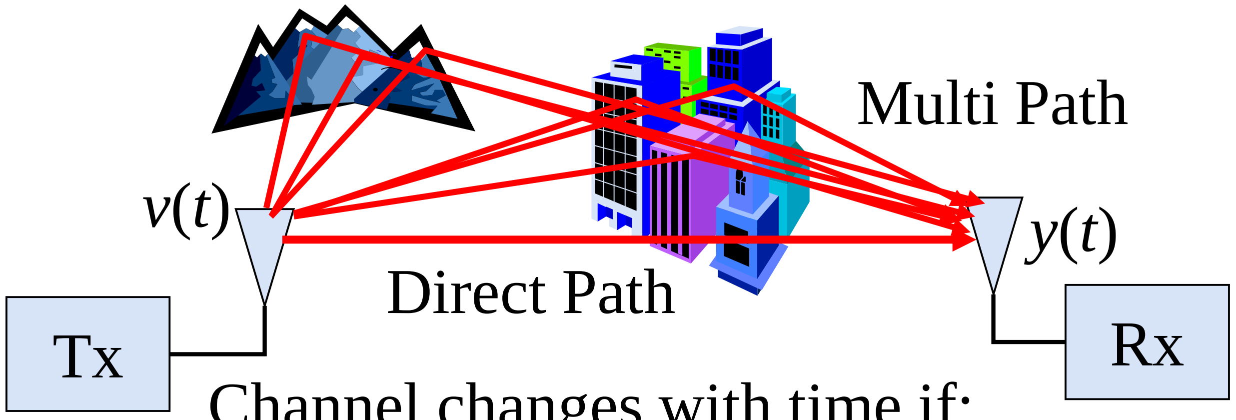
13.8 Signal Processing Examples

Ex. 13.3 Time-Varying Channel Estimation



Channel changes with time if:

- Relative motion between Rx, Tx
- Reflectors move/change with time

$$y(t) = \int_0^T h_t(\tau) v(t-\tau) d\tau$$

T is the maximum delay

Model using a time-varying D-T FIR system

$$y[n] = \sum_{k=0}^{p} h_n[k] v[n-k]$$

Coefficients change at each n
to model time-varying channel

In communication systems, <u>multipath</u> channels <u>degrade performance</u> (Inter-symbol interference (ISI), flat fading, frequency-selective fading, etc.)

<u>Need To</u>: First… estimate the channel coefficients
Second… Build an "Inverse Filter" or "Equalizer"

<u>2 Broad Scenarios</u>:

1. Signal $v(t)$ being sent is known ("Training Data")
2. Signal $v(t)$ being sent is not known ("Blind Channel Est.")

<u>One</u> method for scenario #1 is to use a Kalman Filter:

"State" to be estimated is $\mathbf{h}[n] = [h_n[0] \ \ldots \ h_n[p]]^T$

(Note: "**h**" here is no longer used to notate the "observation model" here)

<u>Need State Equation</u>:

Assume FIR tap coefficients <u>change slowly</u>

$$\mathbf{h}[n] = \mathbf{A}\mathbf{h}[n-1] + \mathbf{u}[n]$$

Assume FIR taps are uncorrelated with each other…

<“uncorrelated scattering”>

⇒ $\mathbf{A}$, $\mathbf{Q}$, $\mathbf{C}_\mathbf{h}$, are Diagonal

Assumed <u>Known</u>
That is a weakness!!

cov{$\mathbf{h}$[-1]} = $\mathbf{M}$[-1|-1]

cov{$\mathbf{u}$[n]}

Need Observation Equation:

Have measurement model from convolution view:

$$x[n] = \sum_{k=0}^{p} h_n[k]v[n-k] + w[n]$$

Known training signal

zero-mean, WGN, σ^2

$$x[n] = \mathbf{v}^T \mathbf{h}_n + w[n]$$

Observation "Matrix" is made up of the samples of the known transmitted signal

State Vector is the filter coefficients

Simple Specific Example: $p = 2$ (1 Direct Path, 1 Multipath)

$$\boxed{\mathbf{h}[n] = \mathbf{A}\mathbf{h}[n-1] + \mathbf{u}[n]}$$

$$\mathbf{A} = \begin{bmatrix} 0.99 & 0 \\ 0 & 0.999 \end{bmatrix} \quad \mathbf{Q} = \begin{bmatrix} 0.0001 & 0 \\ 0 & 0.0001 \end{bmatrix}$$

$\mathbf{Q} = \text{cov}\{\mathbf{u}[n]\}$

Typical Realization of Channel Coefficients

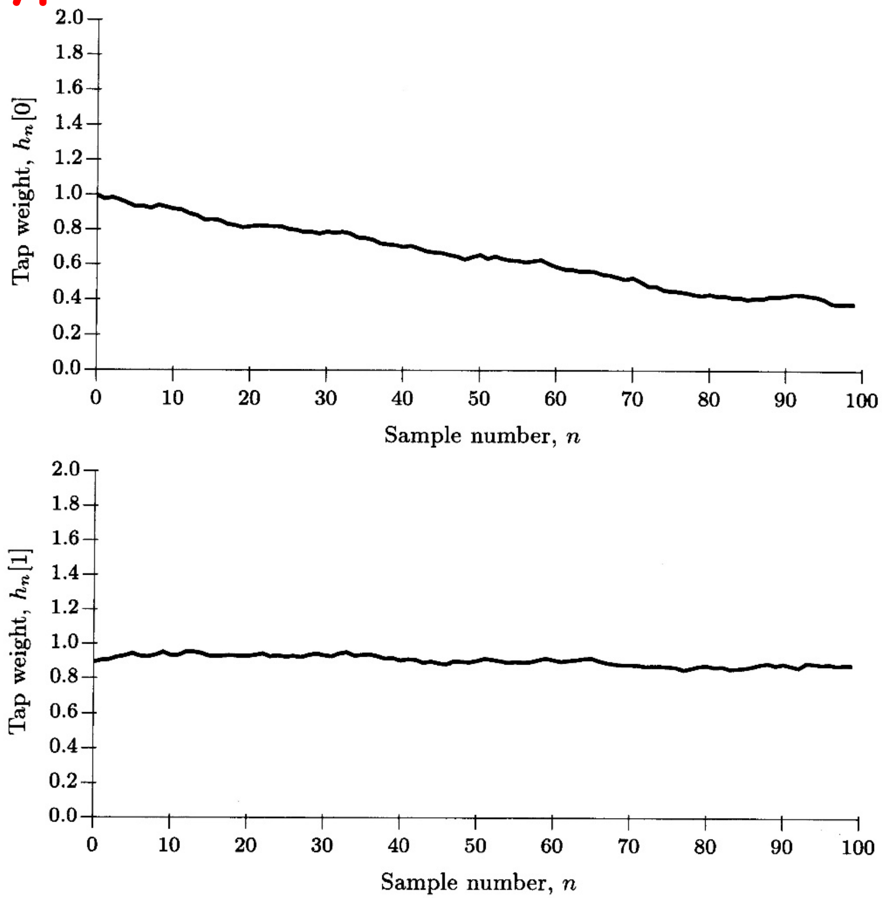
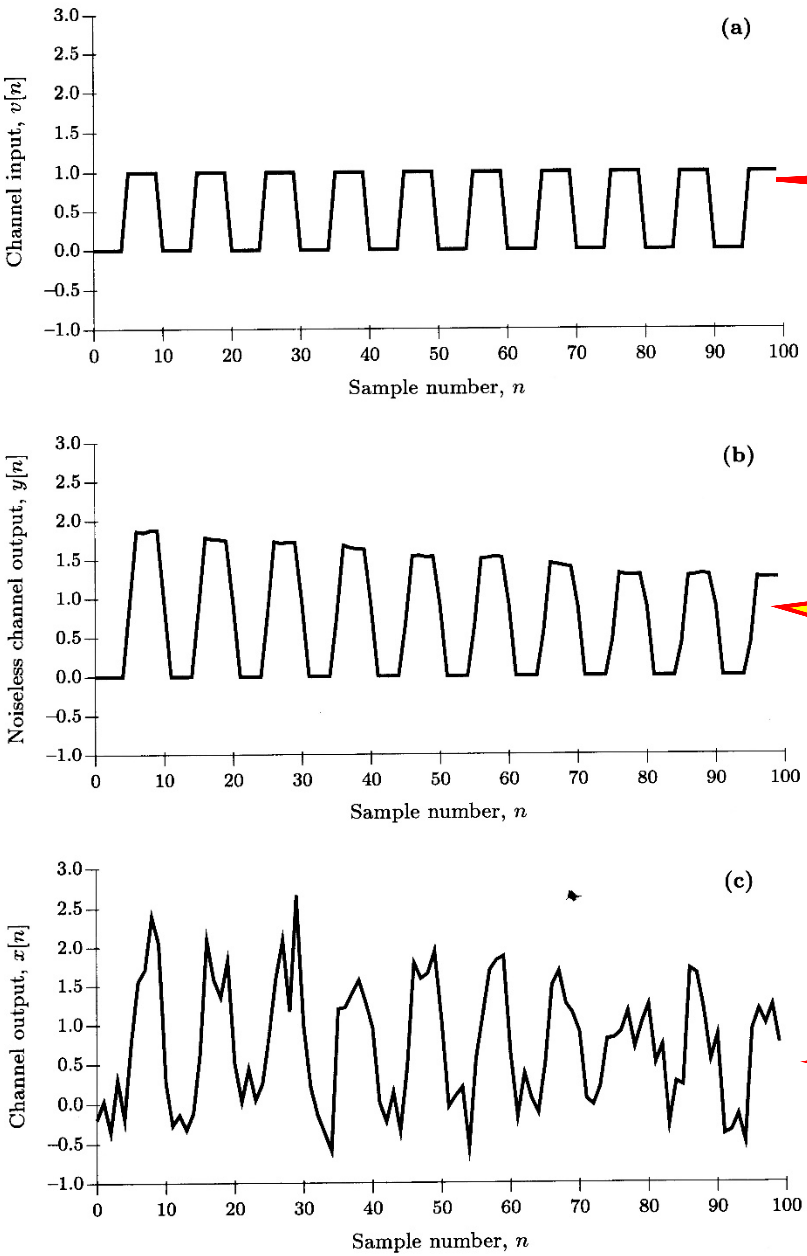


Figure 13.16 Realization of TDL coefficients

Note: $h_n[0]$ decays faster and that the random perturbation is small

Book doesn't state how the initial coefficients were chosen for this realization



Known Transmitted Signal

Noise-Free Received Signal

<It is a bit odd that the received signal is larger than the transmitted signal>

Noisy Received Signal

The variance of the noise in the measurement model is $\sigma^2 = 0.1$

Figure 13.17 Input-output waveforms of channel

Estimation Results Using Standard Kalman Filter

Initialization: $\hat{\mathbf{h}}[-1|-1] = [0 \quad 0]^T$ $\qquad \mathbf{M}[-1|-1] = 100\mathbf{I}$ $\qquad \sigma^2 = 0.1$

Chosen to reflect that little prior knowledge is known

Transient due to wrong IC

Eventually Tracks Well!!

$h_n[0]$

$h_n[1]$

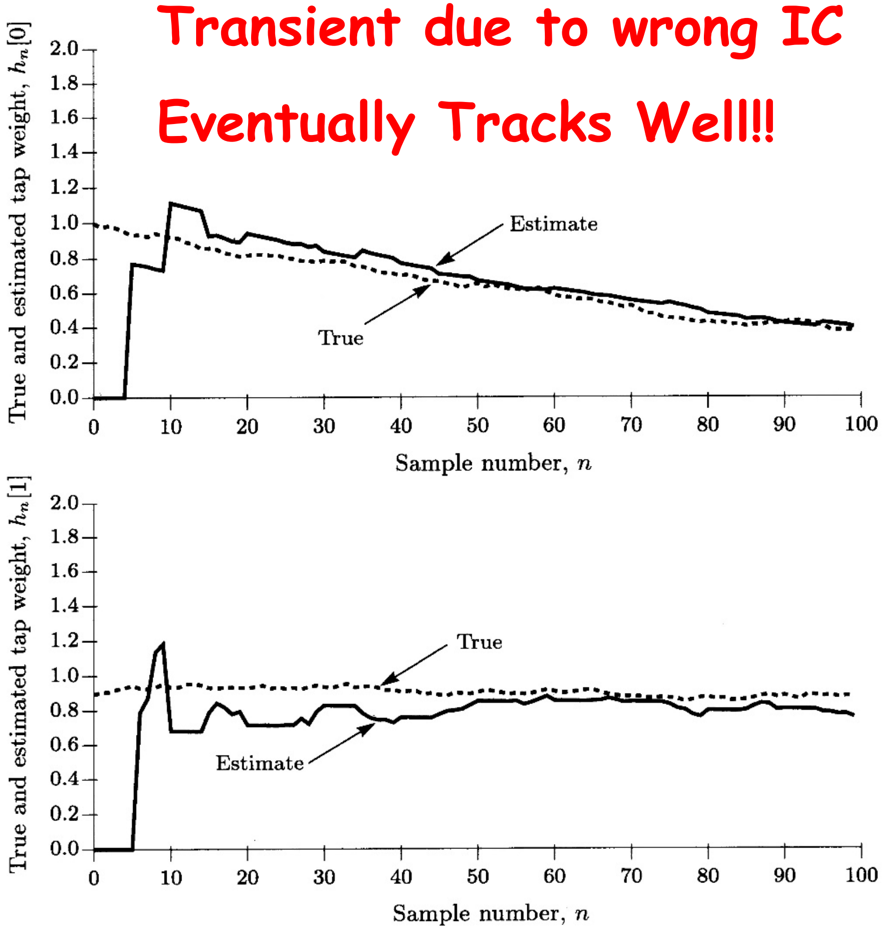


Figure 13.18 Kalman filter estimate

In theory we said that we initialize to the *a priori* mean… but in practice it is common to just pick some arbitrary initial value and set the initial covariance quite high… this forces the filter to start out trusting the data a lot!

Decay down… relies more on model
Gain is zero when signal is noise only

Kalman Filter Gains

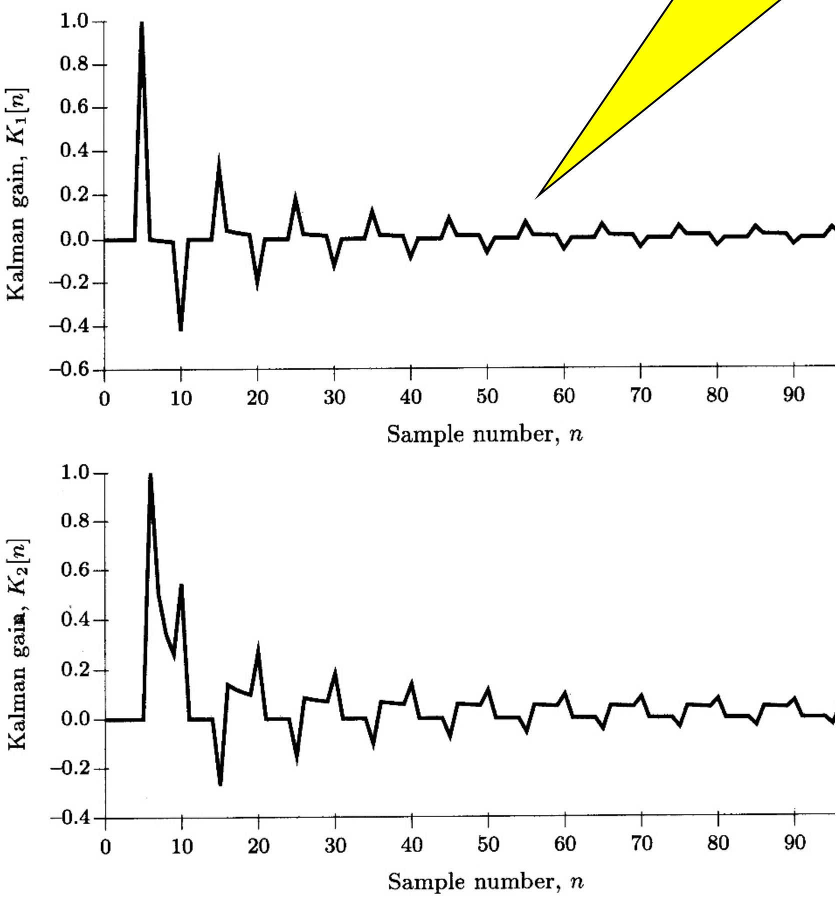


Figure 13.19 Kalman filter gains

Kalman Filter MMSE

Filter Performance
improves with time

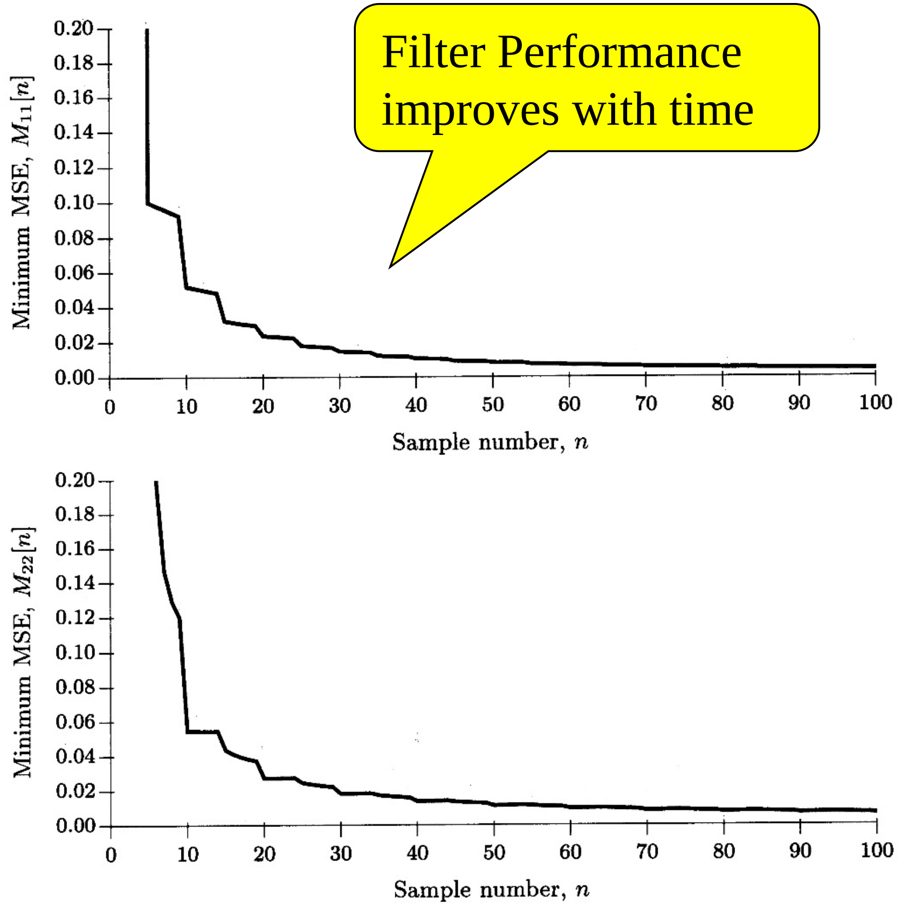


Figure 13.20 Kalman filter minimum MSE

Example: Radar Target Tracking

For this simple example…. assume:

State Model: Constant-Velocity A/C Model

$$\underbrace{\begin{bmatrix} r_x[n] \\ r_y[n] \\ v_x[n] \\ v_y[n] \end{bmatrix}}_{\mathbf{s}[n]} = \underbrace{\begin{bmatrix} 1 & 0 & \Delta & 0 \\ 0 & 1 & 0 & \Delta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} r_x[n-1] \\ r_y[n-1] \\ v_x[n-1] \\ v_y[n-1] \end{bmatrix}}_{\mathbf{s}[n-1]} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ u_x[n] \\ u_y[n] \end{bmatrix}}_{\mathbf{u}[n]}$$

$$\mathbf{Q} = \text{cov}\{\mathbf{u}\} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_u^2 & 0 \\ 0 & 0 & 0 & \sigma_u^2 \end{bmatrix}$$

Velocity perturbations due to wind, slight speed corrections, etc.

Observation Model: Noisy Range/Bearing Radar Measurements

$$\mathbf{x}[n] = \begin{bmatrix} \sqrt{r_x^2[n] + r_y^2[n]} \\ \tan^{-1}\left[\dfrac{r_y[n]}{r_x[n]}\right] \end{bmatrix} + \begin{bmatrix} w_R[n] \\ w_\beta[n] \end{bmatrix}$$

in radians

$$\mathbf{C} = \text{cov}\{\mathbf{w}\} = \begin{bmatrix} \sigma_R^2 & 0 \\ 0 & \sigma_\beta^2 \end{bmatrix}$$

Extended Kalman Filter Issues

Need the following:

1. Linearization of the observation model (see book for details)
   - Calculate by hand, program into the EKF to be evaluated each iteration
2. Covariance of State Driving Noise
   - Assume wind gusts, etc. are as likely to occur in any direction w/ same magnitude ➔ model as indep. w/ common variance

$$\mathbf{Q} = \operatorname{cov}\{\mathbf{u}\} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_u^2 & 0 \\ 0 & 0 & 0 & \sigma_u^2 \end{bmatrix}$$

σ_u = what??? Note: $u_x[n]/\,\Delta$ = acceleration from n-1 to n

So choose σ_u in m/s so that $\sigma_u/\,\Delta$ gives a reasonable range of accelerations for the type of target expected to track

3. Covariance of Measurement Noise

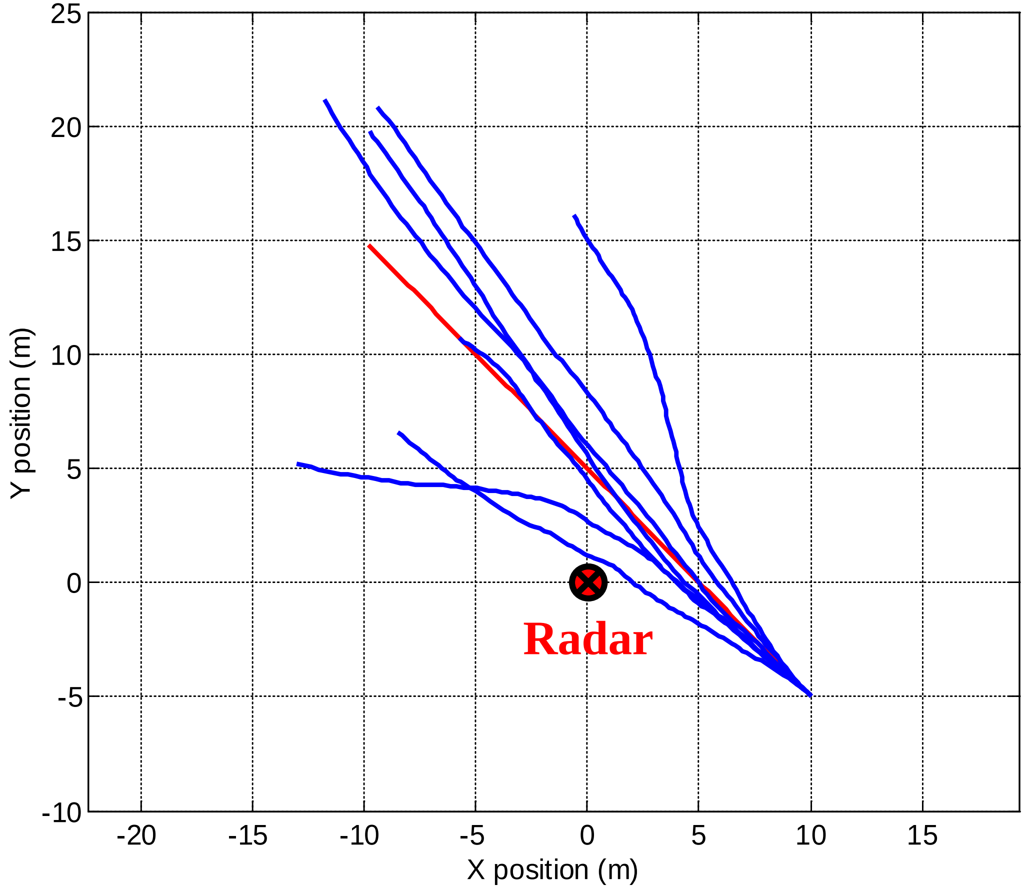
- The DSP engineers working on the radar usually specify this or build routines into the radar to provide time updated assessments of range/bearing accuracy
- Usually assume to be white and zero-mean
- Can use CRLBs for Range & Bearing
    - Note: The CRLBs depend on SNR so the Range & Bearing measurement accuracy should get worse when the target is farther away
- Often assume Range Error to be Uncorrelated with Bearing Error…
    - So… use $\mathbf{C}[n] = \text{diag}\{\sigma_R^2[n], \sigma_\beta^2[n]\}$
- But best to derive joint CRLB to see if they are correlated

4. Initialization Issues

- Typically… Convert first range/bearing into initial r_x & r_y values
- If radar provides no velocity info (i.e. does not measure Doppler) can assume zero velocities
- Pick a large initial MSE to force KF to be unbiased
  - If we follow the above two ideas, then we might pick the MSE for r_x & r_y based on statistical analysis of conversion of range/bearing accuracy into r_x & r_y accuracies
- Sometimes one radar gets a "hand-off" from some other radar or sensor
  - The other radar/sensor would likely hand-off its <u>last</u> track values… so use those as ICs for the initializing the new radar
  - The other radar/sensor would likely hand-off a MSE measure of the quality its <u>last</u> track… so use that as $\mathbf{M}[-1|-1]$

State Model Example Trajectories: Constant-Velocity A/C Model

Red Line is Non-Random Constant Velocity Trajectory



$\Delta = 1 \text{ sec}$

$\sigma_u = 0.01 \text{ m/s} \quad \left(\sigma_u^2 = 0.0001 \text{ m}^2/\text{s}^2\right)$

$r_x[-1] = 10 \quad r_y[-1] = -5 \text{ m}$

$v_x[-1] = -0.2 \quad v_y[-1] = 0.2 \text{ m/s}$

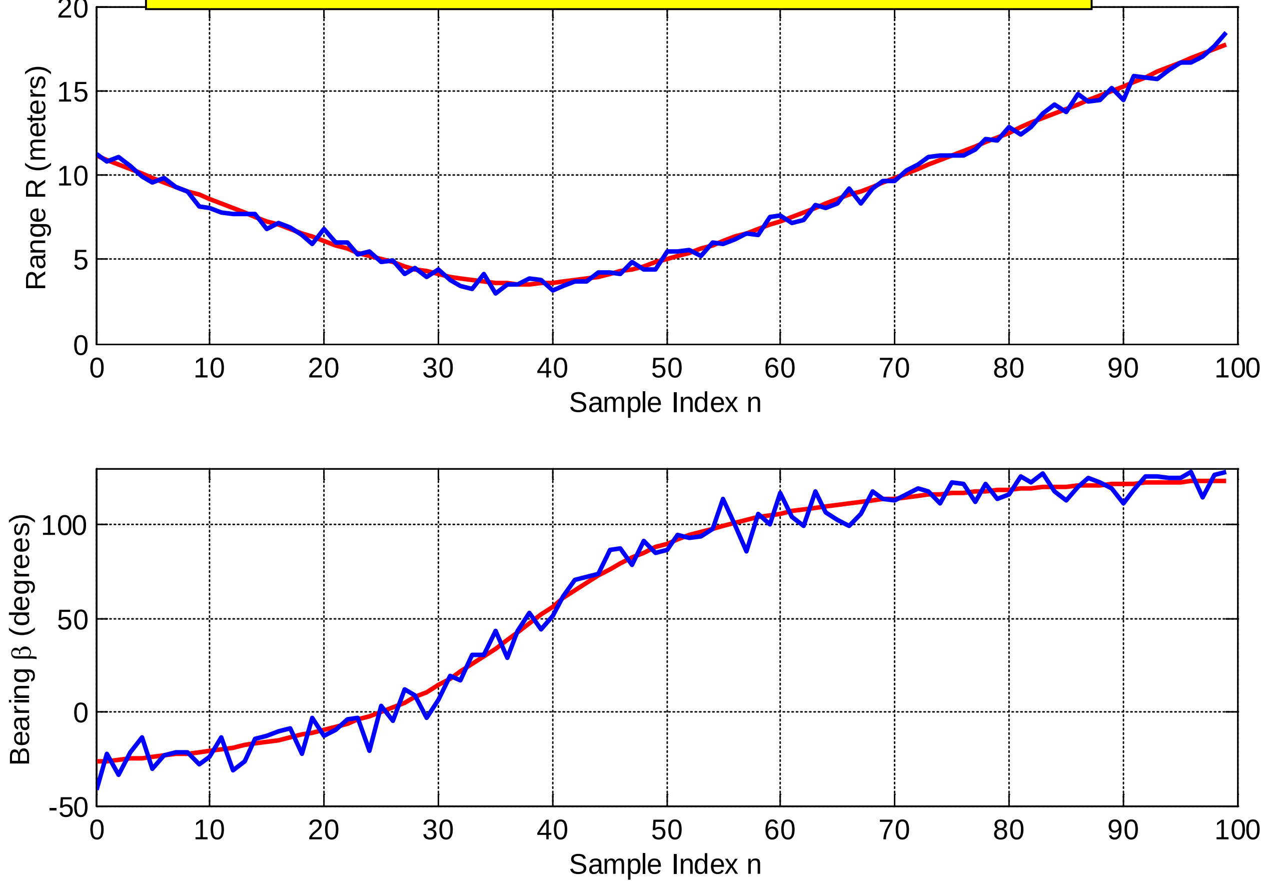
Observation Model Example Measurements

$$\sigma_R = 0.3162 \text{ m } (\sigma_R^2 = 0.1 \text{ m}^2)$$

$$\sigma_\beta = 0.1 \text{ rad} = 5.7 \text{ deg } (\sigma_R^2 = 0.01 \text{ rad}^2)$$

In reality, these would get worse when the target is far away due to a weaker returned signal

Red Lines are Noise-Free Measurements



Measurements Directly Give a Poor Track

If we tried to directly convert the noisy range and bearing measurements into a track… this is what we'd get.

Not a very accurate track!!!! ➔ Need a Kalman Filter!!!

But… Nonlinear Observation Model… so use Extended KF!

Note how the track gets worse when far from the radar (angle accuracy converts into position accuracy in a way that depends on range)

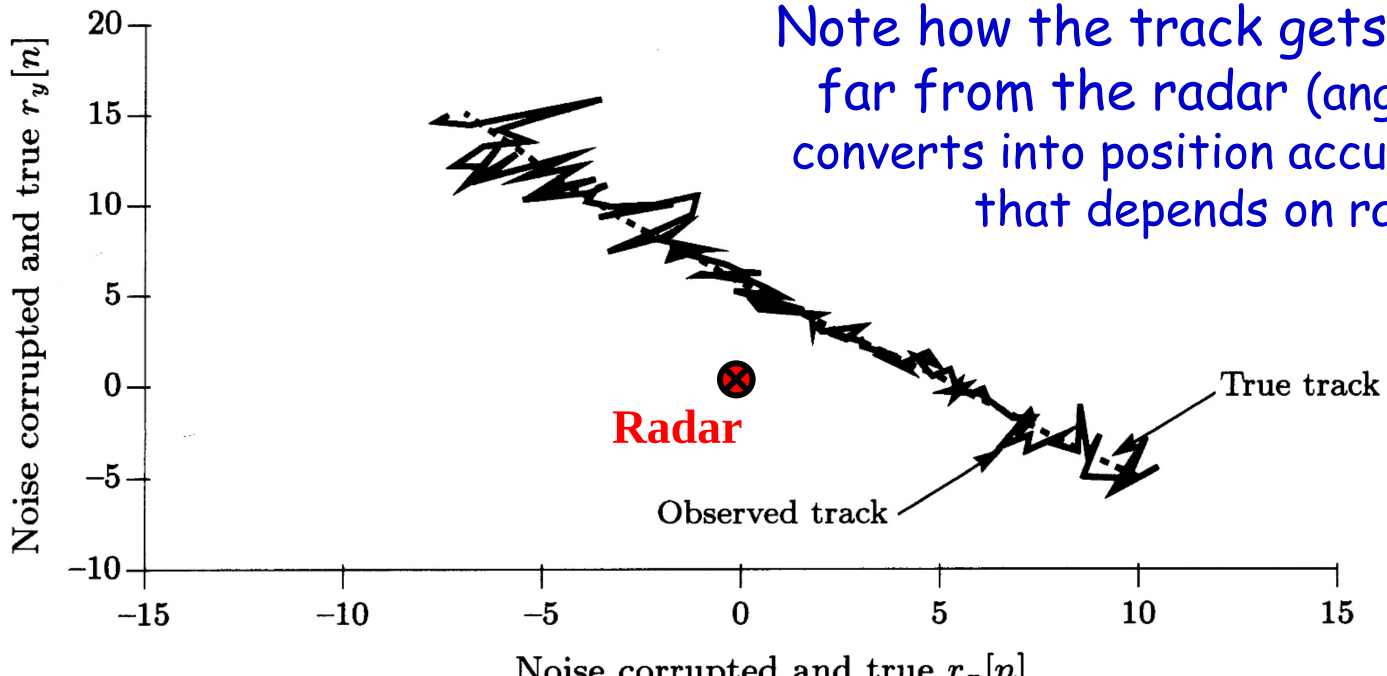


Figure 13.24 True and observed vehicle tracks

Extended Kalman Filter Gives Better Track

Note: The EKF was run with the *correct* values for **Q** and **C** (i.e., the **Q** and **C** used to simulate the trajectory and measurements was used to implement the Kalman Filter)

Initialization: $\mathbf{s}[-1|-1] = [5\ 5\ 0\ 0]^T$ $\mathbf{M}[-1|-1] = 100\mathbf{I}$

Picked "Arbitrarily"

Set large to assert that little is known a priori

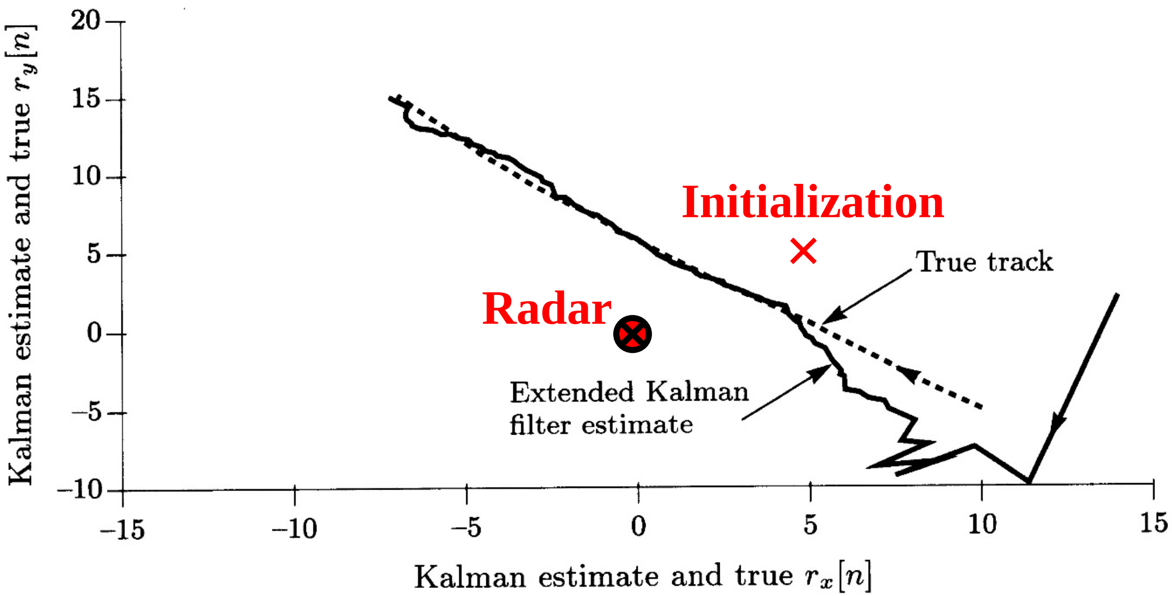


Figure 13.25 True and extended Kalman filter estimate

- After about 20 samples the EKF "attains track" even with poor ICs and the linearization.
- Track gets worse near end where measurements are "worse"
- MSE show "obtain track" and show that things get worse at the end

MSE Plots Show Performance

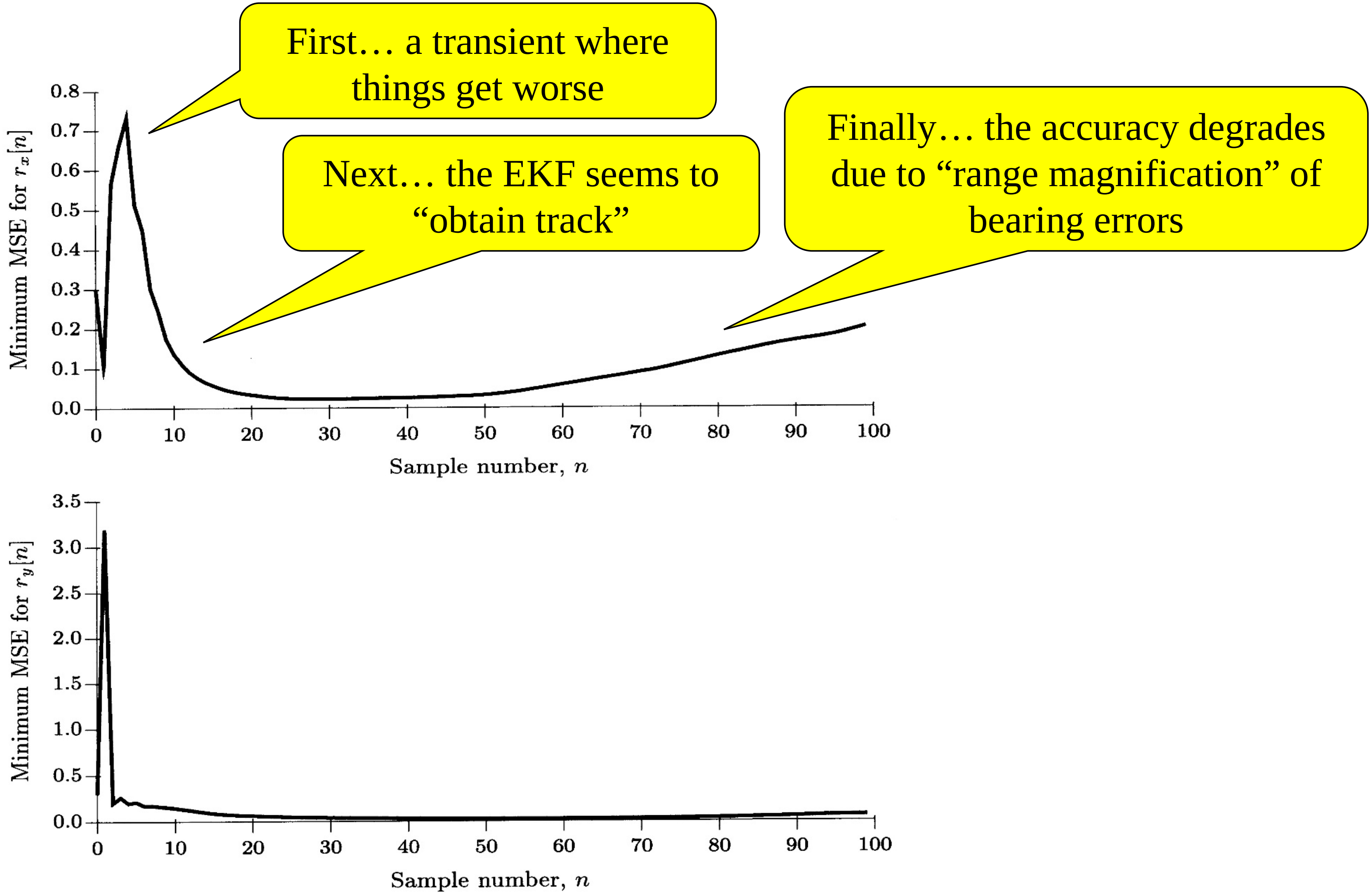


Figure 13.26 "Minimum" MSEs for $r_x[n]$ and $r_y[n]$