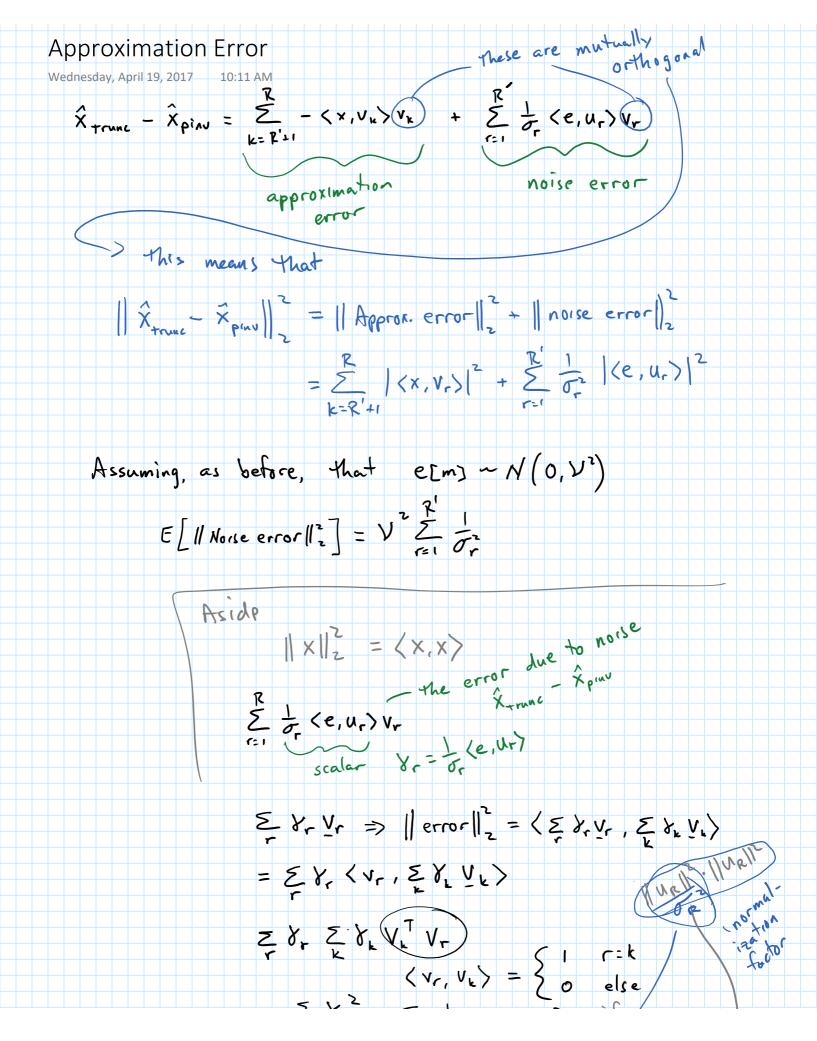
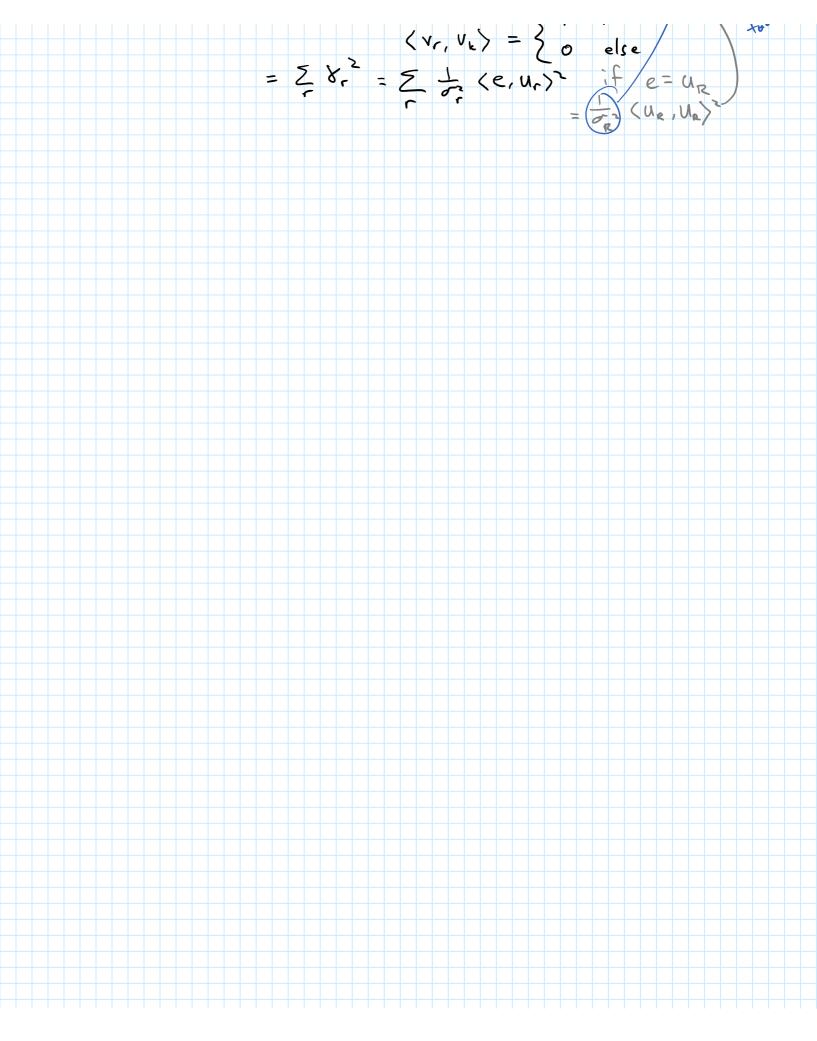
Kalman Filters Wednesday, April 19, 2017 9:58 AM minimize E { e2Cn] } = like LMS except. 1. KF does not observe den] directly 2. KF works in highly non-stationary environments The KF is state-based We have some system model and keep track of the system state example: we have a car moving at a constant speed state: car's position & speed $\times (n) = (P(n)) \leftarrow position$ $S(n) \leftarrow speed$ at time n+1 $\times (n+1)=$ $\left(\begin{array}{c} \rho(n) + \operatorname{st} \cdot S(n) \\ S(n) \end{array}\right)$ we can write this as $X(n+1) = A \times (n) + w(n)$ where $A = \begin{pmatrix} 1 & st \\ 0 & 1 \end{pmatrix}$ we usually have noise the driver - he can adjust the speed w(n)= (Something) $y(n) = X_1(n) + V(n)$ measurement noise y(n)= Cx(n) + V(n) we measure position but not speed and we have C= (10) measurement noise





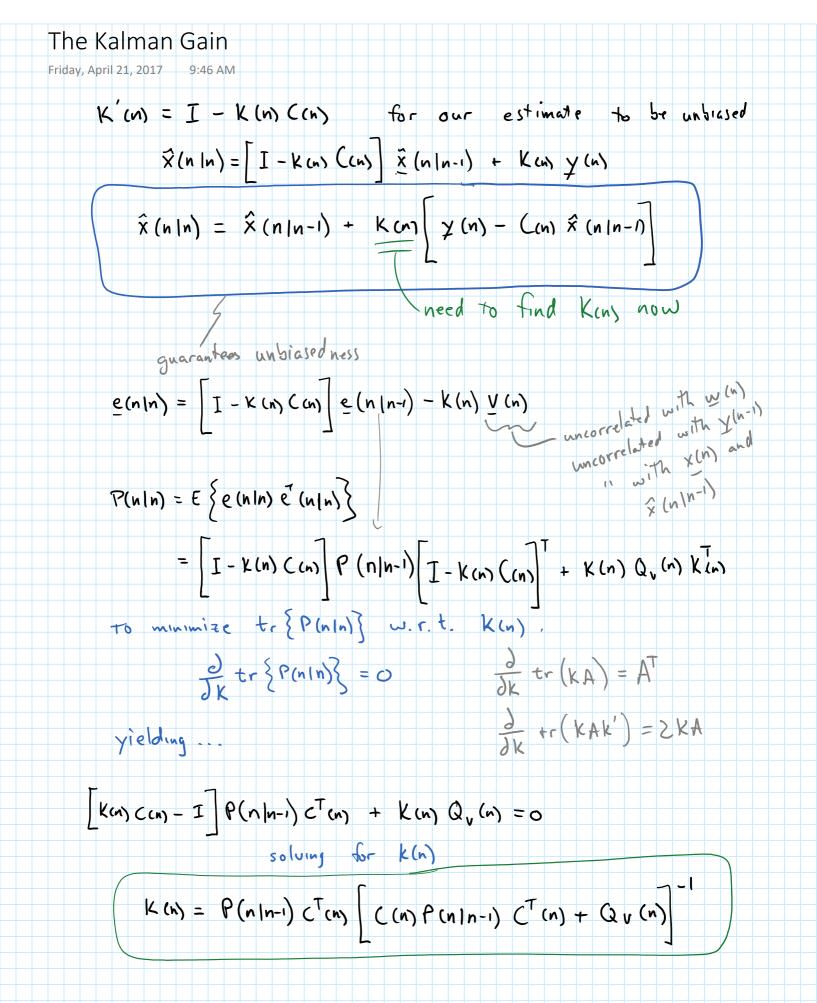
```
Kalman Filter Notation
    Wednesday, April 19, 2017
                State equation
                                                     x (n+1)= A (n+1, n) x (n) + B (n) w(n)
                                                                                                                                                      goal is to find x(n)
                 Observation equation
                                                     y(n) = C(n) \underline{X}(n) + D(n) \underline{V}(n)
                                                                                                                                                                    transforms state to
                                                                                                                                                                           observation
                        V(n) - observation noise
                    ( for this development, assume B & D are identity matrices)
                                                                                                                                                                         all other) at time)
                               E { V (n) V (n) } = Q o (n-k) each time step with is uncorrelated but
                                                                                                                                                                                     there correlations (in Qu)
                     w(n) - process noise
                              E { w (n) w (n) } = Q 6 (n-4)
        Let & (n/i) be the unbiased minimum mean-square estimate
                        x(n) at time n given all the measurements up to
           and including time i
       Let e (nIi) be the corresponding state estimation error
                                e (n/i) = x(n) - x(n/i)
                           P(n|i) = E { e(n|i) eT (n|i) } is the error covariance matrix
          Goal: minimize MSE
E\left\{ \| e(n|n) \|^{2} \right\} = Tr \left\{ P(n|n) \right\} = \sum_{k=0}^{p-1} E\left\{ e_{k}^{2}(n|n) \right\}
= \sum_{n \in rmel} State transition = \sum_{n \in rme
```

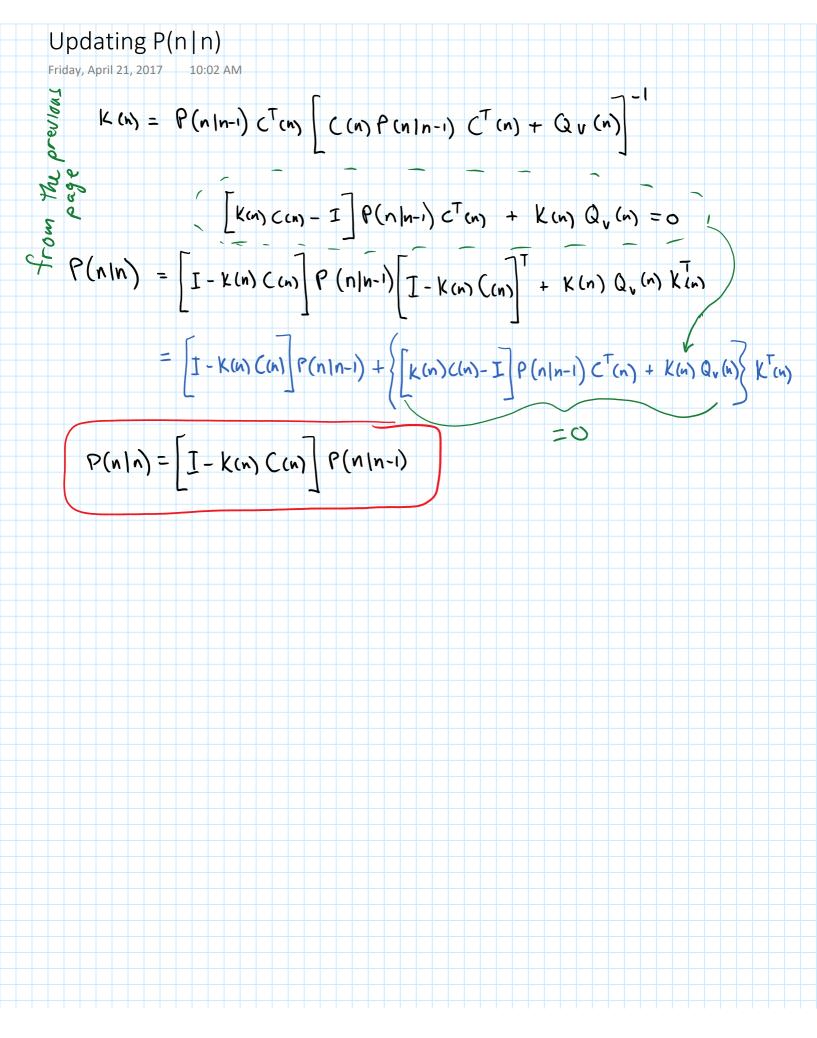
```
Kalman Filter Development
Friday, April 21, 2017
  Given & (n-1/n-1) and P (n-1/n-1), when y (n) becomes available,
  we want to find the unbiased estimate \hat{x} (nin) that
  minimizes
                Tr & P(nIn)}
   Proceed in two steps.
       i) Giver & (n-1/n-1), find & (n/n-1) z "Prediction step"
                - use state transition equations
       2) Given y(n) and \hat{x}(n|n-1), find \hat{x}(n|n)

or tail

detail
Prediction detail
       Since W(n) is zero mean
         \hat{x} (n(n-1) = A(n, n-1) \hat{x} (n-1(n-1)
              e(n|n-1) = \times (n) - \widehat{\times} (n|n-1)
                      = A(n,n-1) \times (n-1) + W(n) - A(n,n-1) \times (n-1)(n-1)
                      = A(n, n-1) \times (n-1) - \hat{x}(n-(|n-1|) + w(n)
                                       e(n-1(n-1)
         if \hat{x}(n-1|n-1) is unbiased then
              E { e (n | n-1) { = A (n, n-1) E { e (n-1 | n-1) } + E { w (n) }
                                                       W(n) is Zero Mean
                                         x (n-11n-1) is
                                          unbiastd
                              = 0
                                       so \hat{X} (n(n-1) is also
                                                unblased
```

```
Kalman Development cont.
   Friday, April 21, 2017
            P(n/n-1) = E { e (n/n-1) e (n/n-1) }
= E \left\{ \left( A(n, n-1) e(n-1/n-1) + w(n) \right) \left( A(n, n-1) e(n-1/n-1) + w(n) \right) \right\}
is uncorrelated with
  w (n)
       P(n|n-1) = A(n,n-1)P(n-1|n-1)A^{T}(n,n-1) + Q_{w}(n)
                                                   Ricatti Equation
                    x (n |n-1) = A(n, n-1) x (n-1 |n-1)
  Step 2 observation
                           we will use a linear estimator
                    \hat{x}(n|n) = K'(n) \hat{x}(n|n-i) + K(n) + K(n)
             we require \hat{x} (n/n) to be unblased => [= {e(n/n)} = 0
              and we want to choose the K' & K
              to minimize E { ||e(n|n)||2}
        e(n|n) = x(n) - K'(n) \hat{x}(n|n-1) - K(n) y(n)
                 = x(n) - k'(n) x(n) - e (n1n-1) - k(n) ((n) x(n) + v(n)
                 = \left[ I - K'(n) - K(n)C(n) \right] \underline{X}(n) + K(n)e(n|n-1) - K(n)V(n)
= \left\{ \begin{cases} 1 - K'(n) - K(n)V(n) \\ 0 \end{cases} \right\} = 0 \quad E \left\{ \begin{cases} 1 - K'(n) - K(n)V(n) \\ 0 \end{cases} \right\} = 0
                    for unbiased, we must E\{e(n|n-i)\}=0 E\{v(n)\}=0 I-V(n)-V(n)
```





State equation: x(n) = A(n,n-1)x(n-1) + w(n)

observation equation: y(n) = C(n) x(n) + y(n)

Noise statistics: E { w (n) wt (n) } = Qw

E { V (n) V (n) } = Q

Initialization: $\hat{X}(0|0) = E\{X(0)\}$

 $P(010) = E\left\{ \left(\times (0) - E\left\{ \times (0) \right\} \right) \left(\times (0) - E\left\{ \times (0) \right\} \right)^{T} \right\}$

Computation: for n=1,...

x (n/n-1) = A(n,n-1) x (n-1/n-1)

P(n1n-1) = A(n,n-1) P(n-1|n-1) AT(n,n-1) + Qw

K(n) = P(n | n-1) CT(n) [C(n) P(n | n-1) CT(n) + Qu]-1

 \hat{x} (n(n) = \hat{x} (n | n-1) + k (n) $\left[\hat{y} (n) - C(n) \hat{x} (n | n-1) \right]$

P(n1n) = [1 - K(n) C(n)] P(n1n-1)

comments: . often we know Q, based on characterization of our sensors

· Qw is often hard to measure, overestimating is

better than underestimating it.

· A large Qw can compensate for a poor model

· We often tune the filter by tweaking Qw ; Qu

· Make P(010) very large it you are uncertain about x(010)