

Optimal Estimation Methods

(Lecture 22 – Introduction to Stochastic Processes)

Dr. John L. Crassidis

University at Buffalo – State University of New York
Department of Mechanical & Aerospace Engineering
Amherst, NY 14260-4400
johnc@buffalo.edu
<http://www.buffalo.edu/~johnc>

Optimal Estimation Methods

(Lecture 21 – Target Tracking & Orbit Determination)

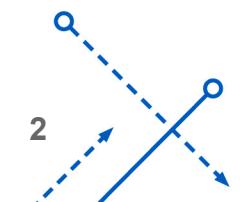
Dr. John L. Crassidis

University at Buffalo – State University of New York
Department of Mechanical & Aerospace Engineering
Amherst, NY 14260-4400
johnc@buffalo.edu
<http://www.buffalo.edu/~johnc>



Target Tracking

Target Tracking



- Early-day application of the Kalman filter
 - Tracking of aircraft from radar observations
 - Actually done before the Kalman filter was developed (using the Wiener filter), but its structure is well explained using the Kalman filter equations *↔ transfer function equivalent of WF.*
 - Two main purposes
 - The first involves actual filtering of the radar measurements to obtain accurate range estimates
 - The second involves the estimation of velocity (and possibly acceleration)
 - Need for accurate velocity estimation
 - Predict ahead of time where multiple targets are expected in future radar scans in order to make a correct association of each target
 - A 3σ bound from the error covariance can be used to assess the validity of the radar scan at future times
 - Used to ensure that the same target is actually tracked, thus avoiding incorrect target associations of multiple vehicles



- One of the simplest target trackers is the $\alpha-\beta$ filter
 - Used to estimate the position and velocity (usually range and range rate) of a vehicle
 - Truth model in continuous time is given by

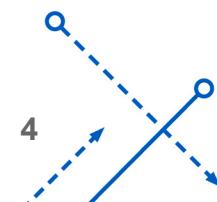
$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w(t), \quad \mathbf{x} \equiv \begin{bmatrix} r \\ \dot{r} \end{bmatrix} \quad (1)$$

where $w(t)$ is the process noise with spectral density q , and the states are position and velocity, denoted by r and \dot{r}

- Note that the first state does not contain any process noise in this formulation
 - This is due to the fact that this state represents a kinematic relationship that is valid in theory and in the real world, since velocity is always the derivative of position
- Discrete time measurements are assumed

$$\tilde{y}_k = [1 \quad 0] \mathbf{x}_k + v_k \equiv H\mathbf{x}_k + v_k, \quad v_k \sim N(0, \sigma_n^2)$$

$\tilde{y} = r$



- Since $F^2 = 0$, the state transition matrix is given by

$$\Phi = I + \Delta t F = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$$

$x_{k+1} = r_k + v_k \Delta t$
 $v_{k+1} = v_k$

where Δt is the sampling interval

- The discrete-time process noise covariance is given by

$$\Sigma_Q \Sigma^T = \int_0^{\Delta t} \Phi(\tau) G Q G^T \Phi^T(\tau) d\tau$$

where $G = [0 \ 1]^T$

- Carrying out the integral gives

$$\Sigma_Q \Sigma^T = q \int_0^{\Delta t} \begin{bmatrix} 1 & \tau \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} [0 \ 1] \begin{bmatrix} 1 & 0 \\ \tau & 1 \end{bmatrix} d\tau$$

$$= q \int_0^{\Delta t} \begin{bmatrix} \tau^2 & \tau \\ \tau & 1 \end{bmatrix} d\tau$$

$$= q \begin{bmatrix} \Delta t^3/3 & \Delta t^2/2 \\ \Delta t^2/2 & \Delta t \end{bmatrix}$$

→ discrete
in position
step
process noise
(2)
5

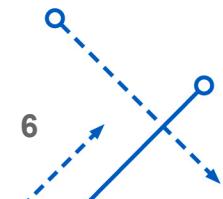
- Notice, unlike the continuous-time process noise term given by $q G G^T$, the discrete-time process noise has nonzero values in all elements
 - This is due to the effect of sampling of a continuous-time process
 - However, if Δt is small, then a first order approximation is given by (as stated in the derivation of the discrete-time KF)

$$\Upsilon Q \Upsilon^T \approx \Delta t q G G^T = q \begin{bmatrix} 0 & 0 \\ 0 & \Delta t \end{bmatrix}$$

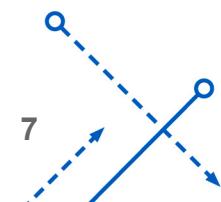
- The discrete-time Kalman filter equations reduce down to

$$\begin{aligned}\hat{r}_k^+ &= \hat{r}_k^- + \alpha [\tilde{y}_k - \hat{r}_k^-] \\ \dot{\hat{r}}_k^+ &= \dot{\hat{r}}_k^- + \frac{\beta}{\Delta t} [\tilde{y}_k - \hat{r}_k^-] \\ \hat{r}_{k+1}^- &= \hat{r}_k^+ + \dot{\hat{r}}_k^+ \Delta t \\ \dot{\hat{r}}_{k+1}^- &= \dot{\hat{r}}_k^+\end{aligned}$$

where the gain is given by $K_k = K \equiv [\alpha \ \beta/\Delta t]^T$



- The gains α and β are often treated as tuning parameters to enhance the tracking performance
 - However, conventional wisdom tells us that tuning these gains individually is incorrect
 - To understand this concept we must remember that the model in Eq. (1) shows a kinematic relationship
 - If α and β are chosen separately, then this kinematic relationship can be lost
 - This means the velocity estimate may not truly be the derivative of the position estimate, even though we know that this relationship is exact
 - A more true-to-physics approach involves tuning the continuous-time process noise parameter q
- From Eq. (2) changes in the velocity over the sampling interval are of the order $\sqrt{q\Delta t}$, which can be used as a guideline in the choice of q



- Define the following propagated and updated error-covariances

$$P^- \equiv \begin{bmatrix} p_{rr}^- & p_{r\dot{r}}^- \\ p_{\dot{r}r}^- & p_{\dot{r}\dot{r}}^- \end{bmatrix}, \quad P^+ \equiv \begin{bmatrix} p_{rr}^+ & p_{r\dot{r}}^+ \\ p_{\dot{r}r}^+ & p_{\dot{r}\dot{r}}^+ \end{bmatrix}$$

- Next, define the following variable

$$S_q = q^{1/2} \Delta t^{3/2} / \sigma_n$$

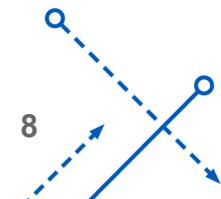
- The propagated error-covariance elements are given by

$$p_{rr}^- = \sigma_n^2 \left[\left(\frac{\xi}{S_q} \right)^2 - 1 \right]$$

$$p_{\dot{r}\dot{r}}^- = \left(\frac{\sigma_n}{\Delta t} \right)^2 \left[S_q^2 \left(\frac{1}{2} - \frac{1}{\xi} \right) + \xi \right]$$

$$p_{r\dot{r}}^- = \frac{\sigma_n^2 \xi}{\Delta t}$$

(3)



where

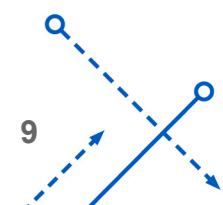
$$\xi = \frac{1}{2} \left[\left(\frac{S_q^2}{2} + \vartheta \right) + \sqrt{\left(\frac{S_q^2}{2} + \vartheta \right)^2 - 4S_q^2} \right]$$

$$\vartheta = \left[4S_q^2 + \frac{S_q^4}{12} \right]^{1/2}$$

- The Kalman gain is given by

$$K \equiv \begin{bmatrix} \alpha \\ \beta/\Delta t \end{bmatrix} = P^- H^T (H P^- H^T + R)^{-1} = \frac{1}{p_{rr}^- + \sigma_n^2} \begin{bmatrix} p_{rr}^- \\ p_{r\dot{r}}^- \end{bmatrix} \quad (4)$$

- This clearly shows that α and β are closely related to one another



- To determine this relationship, first will show the relationship between p_{rr}^- and $p_{r\dot{r}}^-$
- Substituting $\xi = \Delta t p_{r\dot{r}}^- / \sigma_n^2$ into Eq. (3) and solving the resulting equation for $p_{r\dot{r}}^-$ yields

$$p_{r\dot{r}}^- = \frac{\sigma_n S_q}{\Delta t} \sqrt{p_{rr}^- + \sigma_n^2} \quad (5)$$

- Solving for p_{rr}^- from the definition of α in Eq. (4) gives

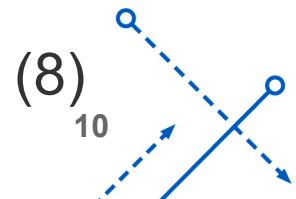
$$p_{rr}^- = \frac{\sigma_n^2 \alpha}{1 - \alpha} \quad (6)$$

- Solving for $p_{r\dot{r}}^-$ from the definition of β in Eq. (4) gives

$$p_{r\dot{r}}^- = \frac{\beta (p_{rr}^- + \sigma_n^2)}{\Delta t} \quad (7)$$

- Substituting Eq. (6) into Eq. (7) yields

$$p_{r\dot{r}}^- = \frac{\sigma_n^2 \beta}{\Delta t (1 - \alpha)} \quad (8)$$



- Substituting Eqs. (6) and (8) into Eq. (5) leads to

$$\frac{\sigma_n^2 \beta}{\Delta t (1 - \alpha)} = \frac{\sigma_n S_q}{\Delta t} \sqrt{\frac{\sigma_n^2 \alpha}{(1 - \alpha)} + \sigma_n^2}$$

$$\frac{\sigma_n^4 \beta^2}{\Delta t^2 (1 - \alpha)^2} = \frac{\sigma_n^2 S_q^2}{\Delta t^2} \left[\frac{\sigma_n^2 \alpha}{(1 - \alpha)} + \sigma_n^2 \right]$$

$$\frac{\sigma_n^2 \beta^2}{(1 - \alpha)^2} = S_q^2 \left[\frac{\sigma_n^2 \alpha + \sigma_n^2 (1 - \alpha)}{(1 - \alpha)} \right]$$

$$\frac{\sigma_n^2 \beta^2}{(1 - \alpha)^2} = S_q^2 \left[\frac{\sigma_n^2}{(1 - \alpha)} \right]$$

- Hence

$$S_q \sim \sqrt{\frac{\alpha}{\sigma_n^2}}$$

$$\boxed{\frac{\beta^2}{1 - \alpha} = S_q^2}$$

Satisfy S_q by
 choosing β & α

(9)

- The quantity S_q is known as the tracking index, since it is proportional to the ratio of the process noise standard deviation and the measurement noise standard deviation

- Kalata's Model

*May not
be consistent
kinematically*

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \Delta t^2/2 \\ \Delta t \end{bmatrix} w_k$$

- Leads to $S_q = q^{1/2} \Delta t^{1/2} / \sigma_n$
- Slightly different than the previously shown one (uses $\Delta t^{1/2}$ instead of $\Delta t^{3/2}$)
- This model assumes that the target undergoes a constant acceleration during the sampling interval and that the accelerations from period to period are independent
 - This model may ignore the kinematic relationship shown by the continuous-time model, and thus is not consistent kinematically

- Gain

$$\mathbf{P}^+ = [\mathbf{I} - \mathbf{K} \mathbf{H}] \mathbf{P}^-$$

$$K \equiv \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \underline{\mathbf{P}^+ H^T R^{-1}} = \sigma_n^{-2} \begin{bmatrix} p_{rr}^+ \\ p_{r\dot{r}}^+ \end{bmatrix}$$

where $k_1 = \alpha$ and $k_2 = \beta/\Delta t$



- Updated covariance is given by

$$p_{rr}^+ = \sigma_n^2 \left[1 - \left(\frac{S_q}{\xi} \right)^2 \right]$$

$$p_{\dot{r}\dot{r}}^+ = \left(\frac{\sigma_n}{\Delta t} \right)^2 \left[\xi - S_q^2 \left(\frac{1}{\xi} + \frac{1}{2} \right) \right]$$

- Equating the first equation above to $k_1 = \alpha$ gives

$$\boxed{\alpha = 1 - \left(\frac{S_q}{\xi} \right)^2}$$

- Using Eq. (9) directly gives

$$\boxed{\beta = S_q \sqrt{1 - \alpha}}$$

- A direct relationship between α and β is possible
 - Substituting $H = [1 \ 0]$ into $P^+ = (I - KH)P^-$ gives

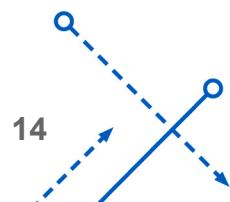
$$\begin{bmatrix} p_{rr}^+ & p_{r\dot{r}}^+ \\ p_{\dot{r}r}^+ & p_{\dot{r}\dot{r}}^+ \end{bmatrix} = \begin{bmatrix} p_{rr}^- (1 - k_1) & p_{r\dot{r}}^- (1 - k_1) \\ p_{\dot{r}r}^- - k_2 p_{rr}^- & p_{\dot{r}\dot{r}}^- - k_2 p_{\dot{r}r}^- \end{bmatrix} \quad (10)$$

- This must be symmetric so that

$$k_1 = \left(\frac{p_{rr}^-}{p_{\dot{r}r}^-} \right) k_2 \quad (11)$$

- Substituting quantities into $P^- = \Phi P^+ \Phi^T + \Upsilon Q \Upsilon^T$ gives

$$\begin{bmatrix} p_{rr}^- & p_{r\dot{r}}^- \\ p_{\dot{r}r}^- & p_{\dot{r}\dot{r}}^- \end{bmatrix} = \begin{bmatrix} p_{rr}^+ + 2p_{r\dot{r}}^+ \Delta t + p_{\dot{r}\dot{r}}^+ \Delta t^2 & p_{r\dot{r}}^+ + p_{\dot{r}\dot{r}}^+ \Delta t \\ p_{\dot{r}r}^+ + p_{\dot{r}\dot{r}}^+ \Delta t & p_{\dot{r}\dot{r}}^+ \end{bmatrix} + q \begin{bmatrix} \Delta t^3/3 & \Delta t^2/2 \\ \Delta t^2/2 & \Delta t \end{bmatrix} \quad (12)$$



- From Eqs. (10) and (12) the 2-2 element gives

$$k_2 = \frac{q \Delta t}{p_{r\dot{r}}^-} \quad (13)$$

- Solving Eq. (11) for p_{rr}^- and using Eq. (13) gives

$$p_{rr}^- = \frac{k_1 q \Delta t}{k_2^2} \quad (14)$$

- From Eqs. (10) and (12) the 1-2 element gives

$$p_{\dot{r}\dot{r}}^- = p_{r\dot{r}}^- \left(\frac{k_1}{\Delta t} + k_2 \right) - \frac{q \Delta t}{2} \quad (15)$$

- From Eqs. (10) and (12) the 1-1 element, with substitution of Eq. (15), leads to

$$p_{rr}^- k_1 + p_{r\dot{r}}^- \Delta t (k_1 - 2) + \frac{q \Delta t^3}{6} = 0$$

(16)
15

- Solving Eq. (13) for p_{rr}^- , and substituting the resulting equation and Eq. (14), into Eq. (16) yields

$$k_1^2 \Delta t + k_2 \Delta t^2 (k_1 - 2) + \frac{k_2^2 \Delta t^3}{6} = 0$$

- From $k_1 = \alpha$ and $k_2 = \beta/\Delta t$, this equation reduces down to

$$\alpha^2 + \beta(\alpha - 2) + \frac{\beta^2}{6} = 0$$

- Since β is always positive, which will be proven in the stability analysis, then α and β are related by

$$\alpha = -\frac{1}{2}\beta + \frac{1}{2}\sqrt{\beta [(\beta/3) + 8]}$$

- This equation clearly shows the relationship between α and β , which can be written without S_q directly

- Investigate discrete-time stability requirements
 - The eigenvalues of $\Phi(I - K H)$ can be found using

$$|zI - \Phi[I - K H]| = \det \begin{bmatrix} z + \alpha + \beta - 1 & -\Delta t \\ \beta/\Delta t & z - 1 \end{bmatrix} = 0$$

- This gives the following characteristic equation

Discrete
↓

$$z^2 + (\alpha + \beta - 2)z + (1 - \alpha) = 0$$

- All eigenvalues must lie within the unit circle for a stable system
 - Even though the characteristic equation is second-order in nature, using the unit circle condition directly to prove stability is arduous
 - Jury's test can be used to easily derive the stability conditions for α and β



- Consider the following second-order polynomial

$$z^2 + a_1 z + a_2 = 0$$

where $a_1 \equiv \alpha + \beta - 2$ and $a_2 \equiv 1 - \alpha$

- Jury's test for stability for this second-order equation involves satisfying the following three conditions

$$a_2 < 1$$

$$a_2 > a_1 - 1$$

$$a_2 > -(a_1 + 1)$$

- From the definitions of a_1 and a_2 , these conditions give $\alpha > 0$, $\beta > 0$, and $2\alpha + \beta < 4$
- From the equation $\alpha = 1 - (S_q/\xi)^2$, since $\alpha > 0$ and $(S_q/\xi)^2 > 0$, then the following conditions must be satisfied for stability

$$0 < \alpha \leq 1$$

$$0 < \beta < 2$$



- The stability conditions are valid even if α and β are chosen independently
- If q is tuned to determine α and β , then from the following equations

$$\frac{\beta^2}{1 - \alpha} = S_q^2$$
$$\alpha = -\frac{1}{2}\beta + \frac{1}{2}\sqrt{\beta [(\beta/3) + 8]}$$

the asymptotic limits are given by

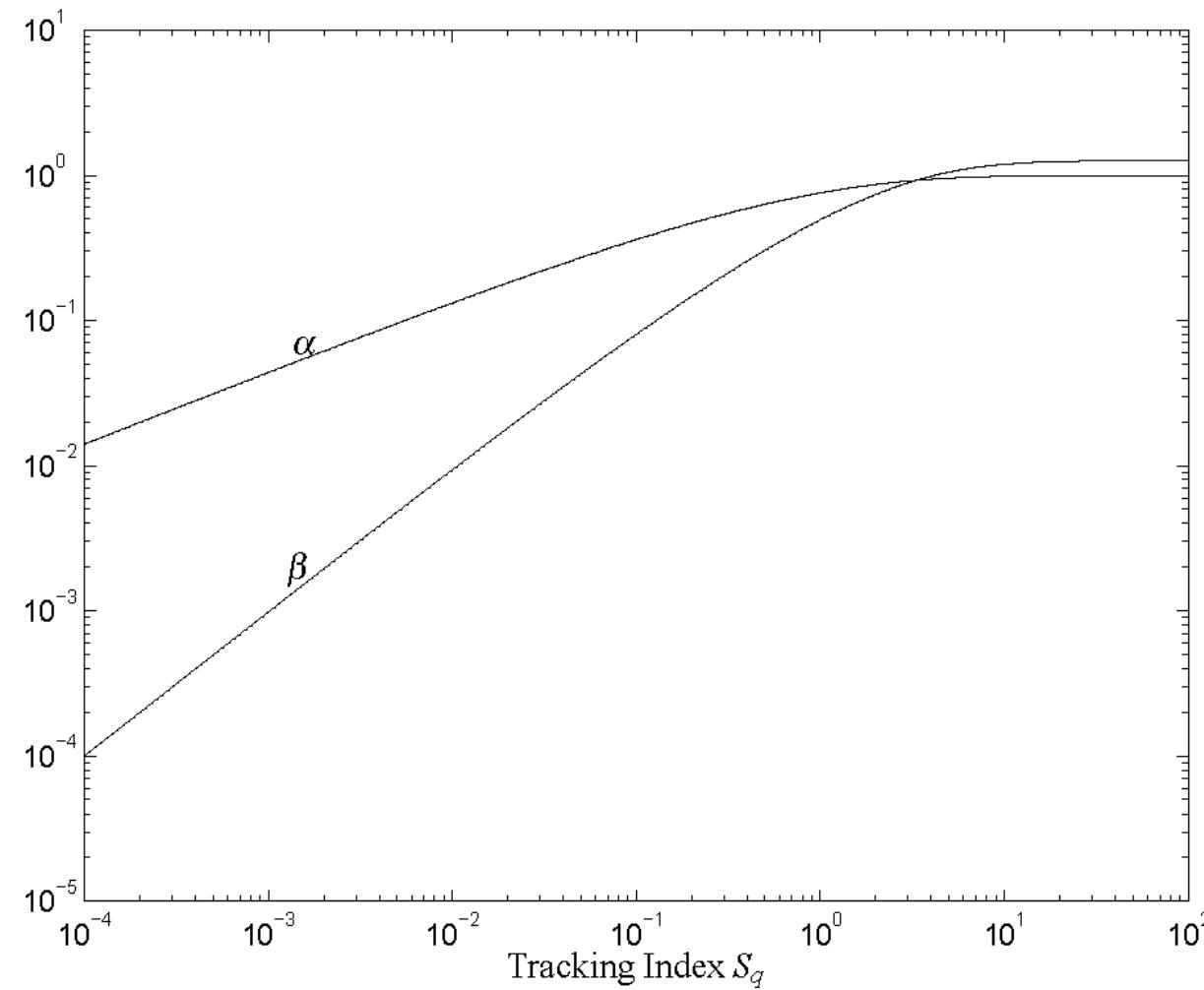
$$\alpha = 1$$

$$\beta = 3 - \sqrt{3} = 1.2679$$

- These limits are clearly within the upper bounds
- Always best to tune q then to tune α and β in order to retain the kinematic relationship (although this is not done in practice usually)



Gains vs. Tracking Index



- Higher-order filter with truth model given by

*(Velocity
Acceleration
Jerk)* = $\dot{\mathbf{x}}$

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t)$$

- Measurement model given by

$$\tilde{y}_k = [1 \ 0 \ 0] \mathbf{x}_k + v_k \equiv H\mathbf{x}_k + v_k$$

- The state transition matrix and covariance are given by

$$\Phi = I + \Delta t F + \frac{\Delta t^2}{2} F^2 = \begin{bmatrix} 1 & \Delta t & \Delta t^2/2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix}$$

$$\Upsilon Q \Upsilon^T = q \begin{bmatrix} \Delta t^5/20 & \Delta t^4/8 & \Delta t^3/6 \\ \Delta t^4/8 & \Delta t^3/3 & \Delta t^2/2 \\ \Delta t^3/6 & \Delta t^2/2 & \Delta t \end{bmatrix}$$



- Kalman update and propagation equations are given by

$$\begin{aligned}
 \hat{r}_k^+ &= \hat{r}_k^- + \alpha [\tilde{y}_k - \hat{r}_k^-] \\
 \dot{\hat{r}}_k^+ &= \dot{\hat{r}}_k^- + \frac{\beta}{\Delta t} [\tilde{y}_k - \hat{r}_k^-] \\
 \ddot{\hat{r}}_k^+ &= \ddot{\hat{r}}_k^- + \frac{\gamma}{2\Delta t^2} [\tilde{y}_k - \hat{r}_k^-] \\
 \hat{r}_{k+1}^- &= \hat{r}_k^+ + \dot{\hat{r}}_k^+ \Delta t + \frac{1}{2} \ddot{\hat{r}}_k^+ \Delta t^2 \\
 \dot{\hat{r}}_{k+1}^- &= \dot{\hat{r}}_k^+ + \ddot{\hat{r}}_k^+ \Delta t \\
 \ddot{\hat{r}}_{k+1}^- &= \ddot{\hat{r}}_k^+
 \end{aligned}$$

where the gain is given by $K_k = K \equiv [\alpha \ \beta/\Delta t \ \gamma/(2\Delta t^2)]^T$

- Changes in the acceleration over the sampling interval are of the order $\sqrt{q\Delta t}$, which can be used as a guideline in the choice of q
 - Can clearly see how this filter can produce better estimates
 - But requires initial acceleration, which may not be known accurately



- Characteristic equation found by

$$|zI - \Phi[I - K H]| = \det \begin{bmatrix} z + \alpha + \beta + \frac{1}{4}\gamma - 1 & -\Delta t & -\frac{1}{2}\Delta t^2 \\ \frac{1}{2\Delta t}(2\beta + \gamma) & z - 1 & -\Delta t \\ \frac{1}{2\Delta t^2}\gamma & 0 & z - 1 \end{bmatrix} = 0$$

which leads to

$$z^3 + \left(\alpha + \beta + \frac{1}{4}\gamma - 3\right)z^2 + \left(3 - 2\alpha - \beta + \frac{1}{4}\gamma\right)z + (\alpha - 1) = 0$$

- Jury's test leads to the following stability conditions

$$0 < \alpha \leq 1$$

$$0 < \beta < 2$$

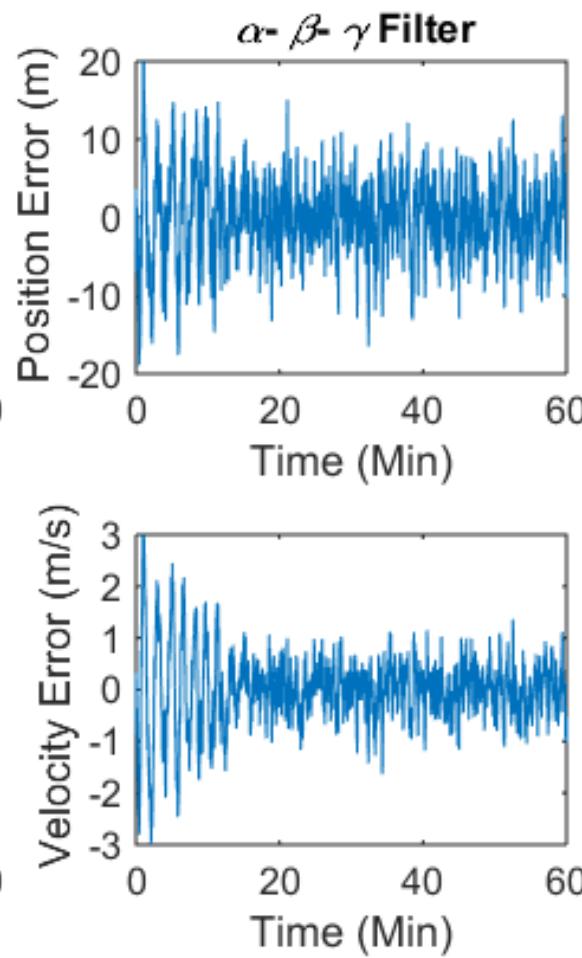
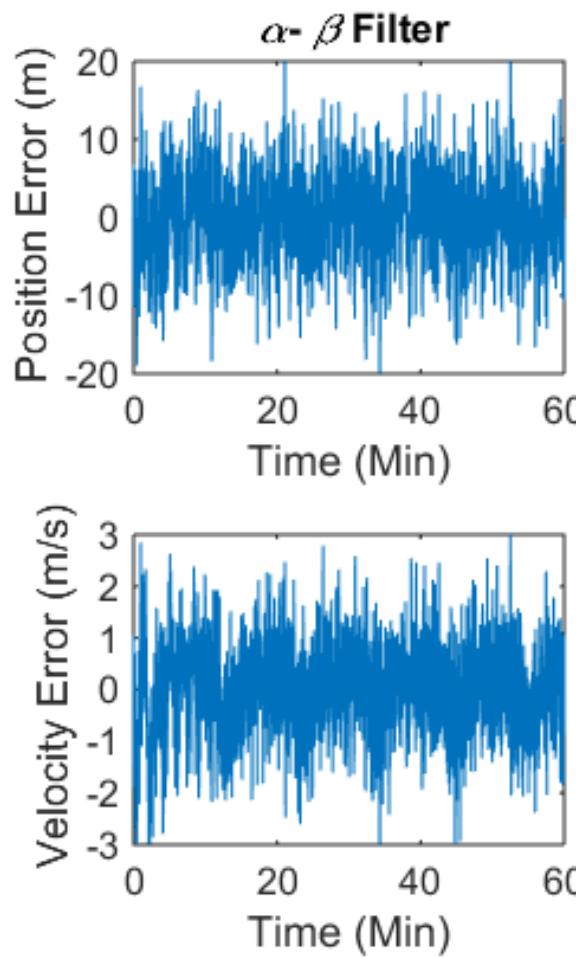
$$0 < \gamma < \frac{4\alpha\beta}{2 - \alpha}$$



- Track the vertical position of an aircraft
 - The vertical position has a standard deviation of 10 m for the measurement error
 - Measurements are sampled at 1-second intervals
 - Since the truth is known, then the variance parameter q in both the filters is tuned to ensure the best possible performance
 - This parameter is varied until transients begin to appear in the position errors
 - For the $\alpha-\beta$ filter the optimal parameter is given by $q = 0.5$
 - Gives $\alpha = 0.31344$ and $\beta = 0.05859$
 - For the $\alpha-\beta-\gamma$ filter the optimal parameter is given by $q = 0.0001$
 - Much smaller than $\alpha-\beta$ filter; affects changes in acceleration, which is smaller in magnitude than changes in velocity
 - Gives $\alpha = 0.18127$ and $\beta = 0.01811$, and $\gamma = 0.00181$



Example (ii)



The α - β - γ filter gives better estimates, especially for velocity, because it is a higher-order filter than the α - β filter



Code (i)

% Aircraft Data

```
cd0=0.0164;cda=0.20;cdde=0;  
cy0=0;cyb=-0.9;cydr=0.120;cyda=0;  
cl0=0.21;cla=4.4;clde=0.32;  
cll0=0;cllb=-.160;clldr=0.008;cllda=0.013;  
cm0=0;cma=-1.00;cmde=-1.30;  
cn0=0;cnb=0.160;cnbr=-0.100;cnnda=0.0018;  
cmq=-20.5;cllp=-0.340;cllr=0.130;cnp=-0.026;cnr=-0.280;
```

% Aircraft Data in SI Units

```
rho=.6536033;s=510.97;cbar=8.321;l=59.74;mass=2831897.6/9.81;  
in=diag([24675882 44877565 67384138]);in(1,3)=1315143.1;in(3,1)=in(1,3);  
g=9.81;
```

```
coef=[cd0;cda;cdde;cy0;cyb;cydr;cyda;cl0;cla;clde;cll0;cllb;clldr;cllda;cm0;cma;cmde;cn0  
cnb;cnbr;cnnda;cmq;cllp;cllr;cnp;cnr];  
other=[rho;s;cbar;l;mass;g];
```



```
% Initial Conditions
```

```
w10=0;w20=0;w30=0;  
xx0=0;yy0=0;zz0=6096;  
vmag=205.13;
```

```
% Trim Conditions
```

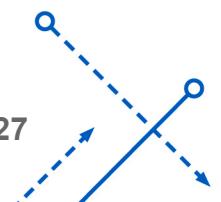
```
qtrim=0.5*rho*vmag^2;  
dtrim=cla*cmde-cma*clde;  
alptrim=((mass*g/qtrim/s-cl0)*cmde+cm0*clde)/dtrim;  
detrim=(-cla*cm0-cma*(mass*g/qtrim/s-cl0))/dtrim;  
dragtrim=(cd0+cda*alptrim+cdde*detrim)*qtrim*s;  
v10=sqrt(vmag^2/(1+tan(alptrim)^2));v20=0;v30=v10*tan(alptrim);
```

```
% Initial Angles
```

```
phi0=0;theta0=0*alptrim;psi0=0;
```

```
% Steady-State Values
```

```
w1_ss=w10;w2_ss=w20;w3_ss=w30;  
v_ss=vmag;
```



```
% True States
```

```
dt=1;t=[0:dt:3600]';m=length(t);  
x=zeros(m,12);  
x(1,1:3)=[v10 v20 v30];  
x(1,4:6)=[w10 w20 w30];  
x(1,7:9)=[xx0 yy0 zz0];  
x(1,10:12)=[phi0 theta0 psi0];
```

```
% Control Surface Inputs and Thrust
```

```
de=detrim*ones(m,1)+1*pi/180*sin(0.01*t);dr=0*ones(m,1);da=0*ones(m,1);  
thrust=dragtrim*ones(m,1);
```

```
% Main Loop for Aircraft Simulation
```

```
for i=1:m-1,
```

```
f1=dt*air_fun(x(i,:),de(i),dr(i),da(i),thrust(i),coef,other,in,w1_ss,w2_ss,w3_ss,v_ss);  
f2=dt*air_fun(x(i,:)+0.5*f1',de(i),dr(i),da(i),thrust(i),coef,other,in,w1_ss,w2_ss,w3_ss,v_ss);  
f3=dt*air_fun(x(i,:)+0.5*f2',de(i),dr(i),da(i),thrust(i),coef,other,in,w1_ss,w2_ss,w3_ss,v_ss);  
f4=dt*air_fun(x(i,:)+f3',de(i),dr(i),da(i),thrust(i),coef,other,in,w1_ss,w2_ss,w3_ss,v_ss);  
x(i+1,:)=x(i,:)+1/6*(f1'+2*f2'+2*f3'+f4');  
end;
```





```
% Velocity, Angle of Attack and Sideslip
```

```
vel=x(:,1:3);
```

```
velm=(vel(:,1).^2+vel(:,2).^2+vel(:,3).^2).^(0.5);
```

```
alp=atan(vel(:,3)./vel(:,1))*180/pi;
```

```
bet=asin(vel(:,2)./velm)*180/pi;
```

```
% Angles
```

```
w=x(:,4:6)*180/pi;
```

```
pos=x(:,7:9);
```

```
phi=x(:,10)*180/pi;theta=x(:,11)*180/pi;psi=x(:,12)*180/pi;
```

```
% Pre-Allocate Space
```

```
pos0=pos(1,3);vel0=vel(1,3);
```

```
m=length(pos);
```

```
pose=zeros(m,1);pose(1)=pos0;
```

```
vele=zeros(m,1);vele(1)=vel0;
```

```
% Noise Parameter
```

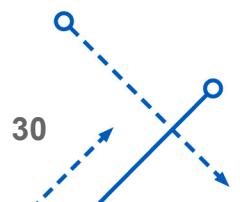
```
sigp=10;
```

```
ym=pos(:,3)+sigp*randn(m,1);
```



Code (v)

```
% Alpha-Beta Filter Variables  
xe_2=zeros(m,2);xe_2(1,:)=[pos0 vel0];  
phi=[1 dt;0 1];h=[1 0];  
  
% Process Noise Tuning and Covariance  
q=.5;  
qd=q*[1/3*dt^3 0.5*dt^2;0.5*dt^2 dt];  
pcov=dare(phi',h',qd,sigp^2,zeros(2,1),eye(2));  
gain=pcov*h'*inv(h*pcov*h'+sigp^2);  
  
sig3_alp_bet=diag(pcov).^(0.5)*3  
disp(' ')  
  
% Alpha-Beta Filter  
for i = 1:m-1  
    xe_2(i+1,:)=(phi*xe_2(i,:)' + phi*gain*(ym(i)-xe_2(i,1)))';  
end
```



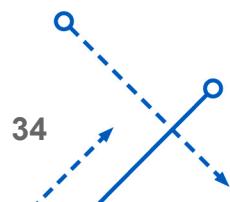
```
% Alpha-Beta-Gamma Filter Variables  
xe_3=zeros(m,3);xe_3(1,:)=[pos0 vel0 0];  
phi=[1 dt dt^2/2;0 1 dt;0 0 1];h=[1 0 0];  
  
% Process Noise Tuning and Covariance  
q=.0001;  
qd=q*[dt^5/20 dt^4/8 dt^3/6;dt^4/8 dt^3/3 dt^2/2;dt^3/6 dt^2/2 dt];  
pcov=dare(phi',h',qd,sigp^2,zeros(3,1),eye(3));  
gain=pcov*h'*inv(h*pcov*h'+sigp^2);  
  
sig3_alp_bet_gam=diag(pcov).^(0.5)*3  
  
% Alpha-Beta-Gamma Filter  
for i = 1:m-1  
    xe_3(i+1,:)=(phi*xe_3(i,:)' + phi*gain*(ym(i)-xe_3(i,1)))';  
end  
  
% Velocity  
zvel=diff(pos(:,3))/dt;zvel(m)=zvel(m-1);
```

```
% Plot Results
subplot(221)
plot(t/60,pos(:,3)-xe_2(:,1));
set(gca,'fontsize',12)
axis([0 60 -20 20]);
set(gca,'Ytick',[-20 -10 0 10 20]);
set(gca,'Xtick',[0 20 40 60]);
ylabel('Position Error (m)')
xlabel('Time (Min)')
title('{$\alpha$}-{$\beta$} Filter')

subplot(223)
plot(t/60,zvel-xe_2(:,2));
set(gca,'fontsize',12)
axis([0 60 -3 3]);
set(gca,'Ytick',[-3 -2 -1 0 1 2 3]);
set(gca,'Xtick',[0 20 40 60]);
ylabel('Velocity Error (m/s)')
xlabel('Time (Min)')
```

```
subplot(222)
plot(t/60,pos(:,3)-xe_3(:,1));
set(gca,'fontsize',12)
axis([0 60 -20 20]);
set(gca,'Ytick',[-20 -10 0 10 20]);
set(gca,'Xtick',[0 20 40 60]);
ylabel('Position Error (m)')
xlabel('Time (Min)')
title('{$\alpha$}-{$\beta$}-{$\gamma$} Filter')
```

```
subplot(224)
plot(t/60,zvel-xe_3(:,2));
set(gca,'fontsize',12)
axis([0 60 -3 3]);
set(gca,'Ytick',[-3 -2 -1 0 1 2 3]);
set(gca,'Xtick',[0 20 40 60]);
ylabel('Velocity Error (m/s)')
xlabel('Time (Min)')
```



```
function f=air_fun(x,de,dr,da,thrust,coef,other,in,w1_ss,w2_ss,w3_ss,v_ss);  
% Function Routine for General Aircraft Equations
```

```
% Inertias
```

```
ixx=in(1,1);iyy=in(2,2);izz=in(3,3);ixz=in(1,3);
```

```
% Velocities and Euler Angles
```

```
v1=x(1);v2=x(2);v3=x(3);  
w1=x(4);w2=x(5);w3=x(6);  
phi=x(10);theta=x(11);psi=x(12);
```

```
% Get Other Constants
```

```
f=zeros(12,1);  
m=other(5);  
g=other(6);
```

```
% Speed, Angle of Attack and Sideslip
```

```
vm=norm([x(1);x(2);x(3)]);  
alp=atan(x(3)/x(1));  
bet=asin(x(2)/vm);
```



Code (x)

```
% Dynamic Pressure
```

```
q=0.5*other(1)*vm^2;
```

```
% General Coefficients
```

```
cd=coef(1)+coef(2)*alp+coef(3)*de;
```

```
cy=coef(4)+coef(5)*bet+coef(6)*dr+coef(7)*da;
```

```
cl=coef(8)+coef(9)*alp+coef(10)*de;
```

```
dd=2*vm^2;
```

```
cld=coef(11)+coef(12)*bet+coef(13)*dr+coef(14)*da+coef(23)*(w1-w1_ss)*other(4)/2/v_ss...
+coef(24)*(w3-w3_ss)*other(4)/2/v_ss;
```

```
cmd=coef(15)+coef(16)*alp+coef(17)*de+coef(22)*(w2-w2_ss)*other(3)/2/v_ss;
```

```
cnd=coef(18)+coef(19)*bet+coef(20)*dr+coef(21)*da+coef(25)*(w1-w1_ss)*other(4)/2/v_ss...
+coef(26)*w3*other(4)/2/v_ss;
```

```
% Drag, Force and Lift
```

```
drag=cd*q*other(2);
```

```
yforce=cy*q*other(2);
```

```
lift=cl*q*other(2);
```



% Torques

```
la1=cll*q*other(2)*other(4);  
la2=cm*q*other(2)*other(3);  
la3=cn*q*other(2)*other(4);
```

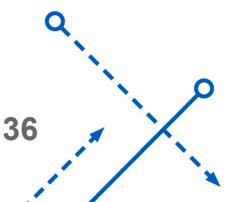
% Forces

```
force1=-drag*cos(alp)+lift*sin(alp)+thrust*cos(alp);  
force2=yforce;  
force3=-drag*sin(alp)-lift*cos(alp)+thrust*sin(alp);
```

% Functions

```
f(1)=-g*sin(theta)+force1/m+v2*w3-v3*w2;  
f(2)=g*cos(theta)*sin(phi)+force2/m+v3*w1-v1*w3;  
f(3)=g*cos(theta)*cos(phi)+force3/m+v1*w2-v2*w1;
```

```
k1=ixz*(iyy-ixx);  
k2=izz*(izz-iyy);  
k3=ixz*(izz-iyy);  
k4=ixx*(iyy-ixx);
```



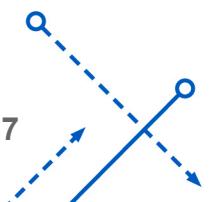


Code (xii)

```
f(4)=(ixz*la3+izz*la1-k1*w1*w2-ixz^2*w2*w3+ixz*izz*w1*w2-k2*w2*w3)/(ixx*izz-ixz^2);
f(5)=(la2-(ixx-izz)*w1*w2-ixz*(w1^2-w3^2))/iyy;
f(6)=(ixx*la3+ixz*la1+ixz^2*w1*w2-k3*w2*w3-k4*w1*w2-ixx*ixz*w2*w3)/(izz*ixx-ixz^2);

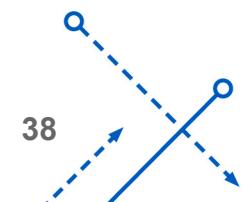
f(7)=cos(theta)*cos(psi)*v1...
+(sin(phi)*sin(theta)*cos(psi)-cos(phi)*sin(psi))*v2...
+(cos(phi)*sin(theta)*cos(psi)+sin(phi)*sin(psi))*v3;
f(8)=cos(theta)*sin(psi)*v1...
+(sin(phi)*sin(theta)*sin(psi)+cos(phi)*cos(psi))*v2...
+(cos(phi)*sin(theta)*sin(psi)-sin(phi)*cos(psi))*v3;
f(9)=-sin(theta)*v1+sin(phi)*cos(theta)*v2+cos(phi)*cos(theta)*v3;

f(10)=w1+sin(phi)*tan(theta)*w2+cos(phi)*tan(theta)*w3;
f(11)=cos(phi)*w2-sin(phi)*w3;
f(12)=sin(phi)*sec(theta)*w2+cos(phi)*sec(theta)*w3;
```

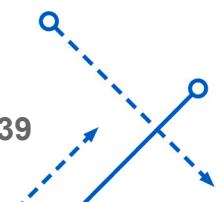


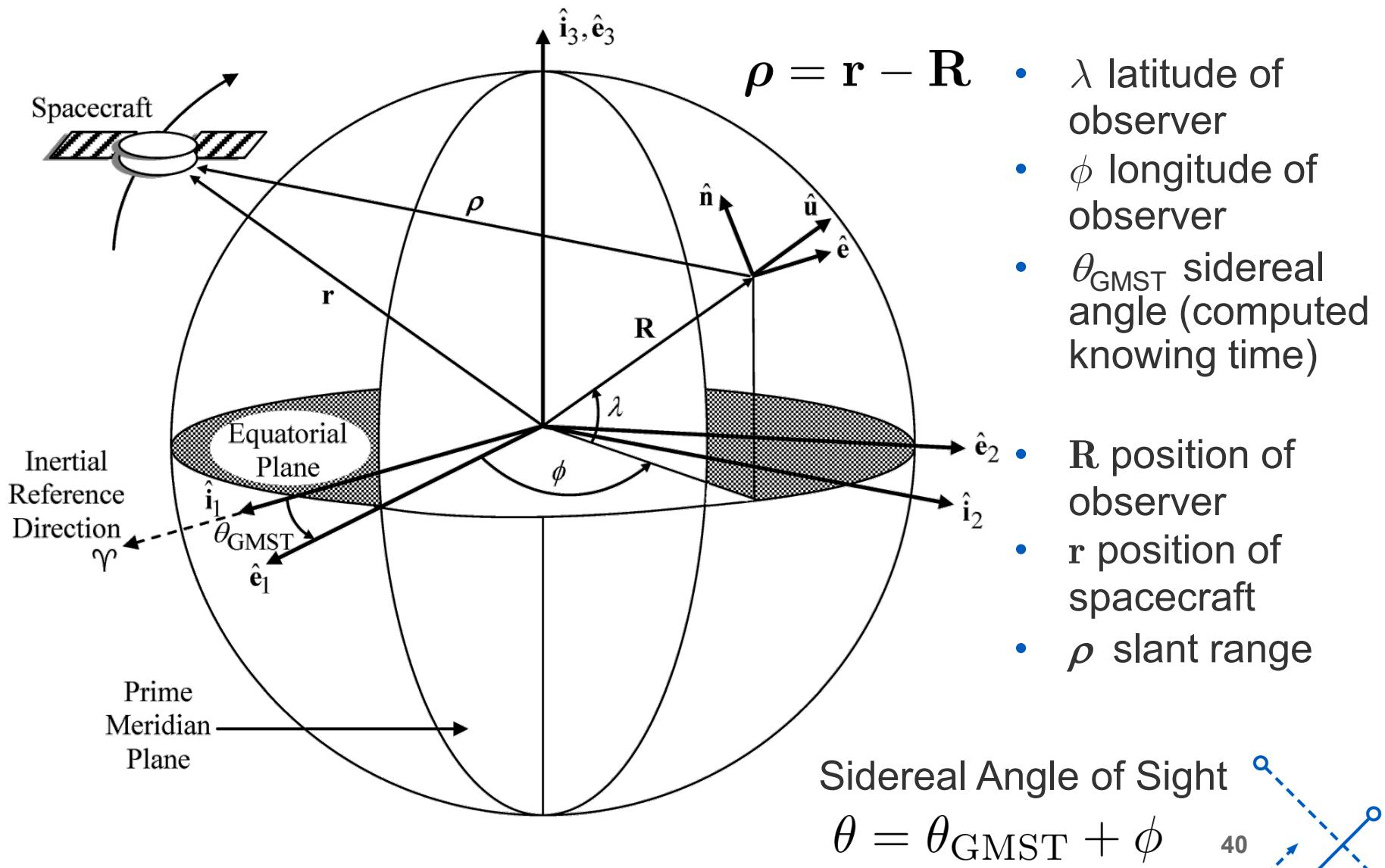
1

Orbit Estimation



- Two main approaches
 - Least Squares
 - Kalman filter
- Least Squares Approach
 - Determination of the planetary objects from telescope/sextant observations was Gauss' motivation to invent least squares!
 - Still used today
 - Goal
 - Given a number of observations during some time interval, determine the orbit position and velocity at some epoch
 - Nonlinear least squares must be used
 - A differential correction is used since differential equations are employed for the orbit estimation
 - Use initial orbit determination methods to initialize the nonlinear least squares process





- Slant range in inertial components is given by

$$\rho = \begin{bmatrix} x - \|\mathbf{R}\| \cos \lambda \cos \theta \\ y - \|\mathbf{R}\| \cos \lambda \sin \theta \\ z - \|\mathbf{R}\| \sin \lambda \end{bmatrix}$$

- Conversion from inertial to observer (“up, east and north) system is given by **VEN - for space craft**

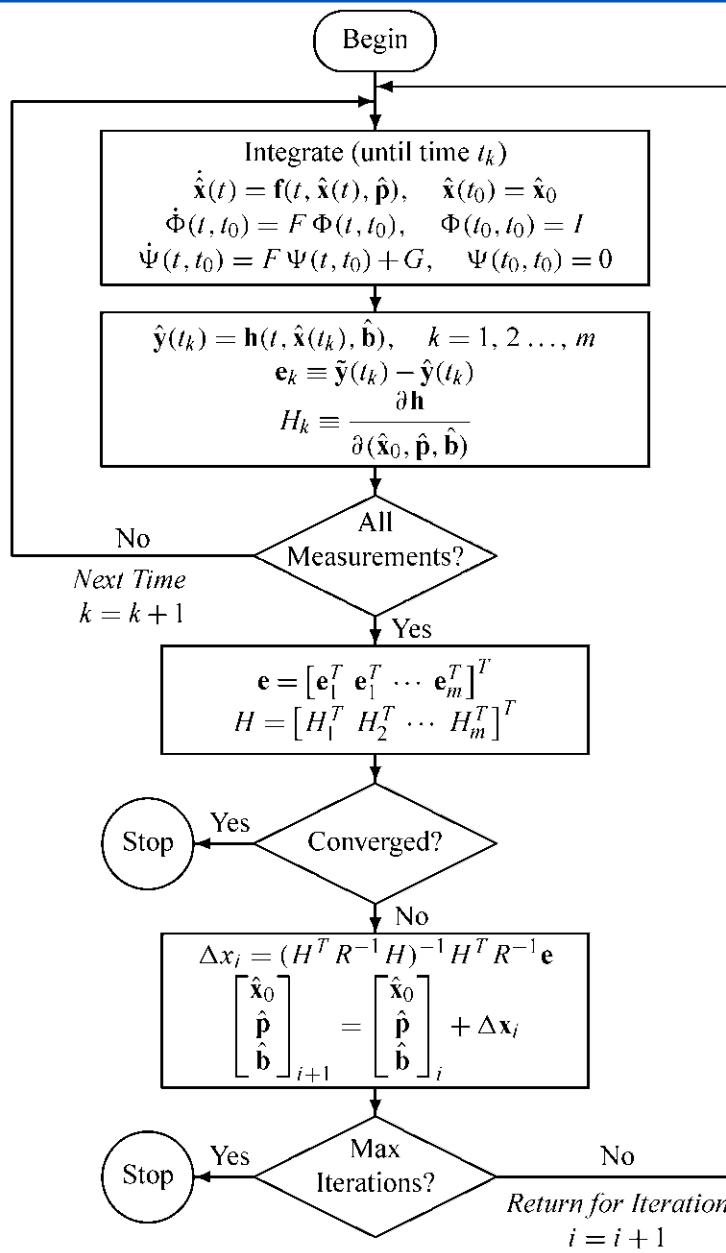
$$\begin{bmatrix} \rho_u \\ \rho_e \\ \rho_n \end{bmatrix} = \begin{bmatrix} \cos \lambda & 0 & \sin \lambda \\ 0 & 1 & 0 \\ -\sin \lambda & 0 & \cos \lambda \end{bmatrix} \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \rho$$

- Assume radar with range, azimuth and elevation

$$\|\rho\| = (\rho_u^2 + \rho_e^2 + \rho_n^2)^{1/2}$$

$$\text{az} = \tan^{-1} \left(\frac{\rho_e}{\rho_n} \right), \quad \text{el} = \sin^{-1} \left(\frac{\rho_u}{\|\rho\|} \right)$$





$$\ddot{\mathbf{r}} = -\frac{\mu}{||\mathbf{r}||^3} \mathbf{r}, \quad \mathbf{r} = [x \ y \ z]^T$$

The goal is to determine $\mathbf{x}_0 = [\mathbf{r}_0^T \ \dot{\mathbf{r}}_0^T]^T$

Also includes other parameters if desired, given by p (e.g., the parameter μ can also be determined if desired)

Other quantities, such as measurement biases or force model parameters, can be appended to the measurement observation equation through the vector b

- Truth at epoch

$$\mathbf{r}_0 = [7,000 \quad 1,000 \quad 200]^T \text{ km}, \quad \dot{\mathbf{r}}_0 = [4 \quad 7 \quad 2]^T \text{ km/sec}$$

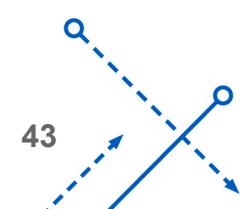
- Measurements

- The latitude of the observer is given by $\lambda = 5^\circ$
- Initial sidereal time is given by $\theta_0 = 5^\circ$
- Measurements are given at 10-second intervals over a 100-second simulation
- The measurement errors are zero-mean Gaussian with a standard deviation of the range measurement error given by 1 km, and a standard deviation of the angle measurements given by 0.01 degrees

- Herrick-Gibbs estimates and truth (second time-step)

$$\hat{\mathbf{r}} = [7,038 \quad 1,070 \quad 221]^T \text{ km}, \quad \mathbf{r} = [7,040 \quad 1,070 \quad 220]^T \text{ km}$$

$$\dot{\hat{\mathbf{r}}} = [3.92 \quad 7.00 \quad 2.00]^T \text{ km/sec}, \quad \dot{\mathbf{r}} = [3.92 \quad 7.00 \quad 2.00]^T \text{ km/sec}$$



Example (ii)

- Initialized least squares with

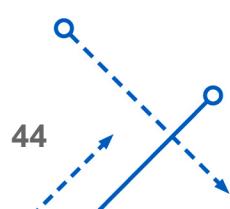
$$\hat{\mathbf{r}}_0 = [6,990 \quad 1 \quad 1]^T \text{ km}, \quad \dot{\hat{\mathbf{r}}}_0 = [1 \quad 1 \quad 1]^T \text{ km/sec}$$

- Test robustness of the algorithm

Iteration	Position (km)			Velocity (km/sec)		
0	6,990	1	1	1	1	1
1	7,496	1,329	-178	5.30	6.20	-18.42
2	7,183	609	27	12.66	22.63	12.69
3	6,842	905	490	6.65	13.73	-8.15
4	6,795	963	255	9.33	7.38	1.36
5	6,985	989	199	4.24	7.20	1.89
6	7,000	1,000	200	4.00	7.00	2.00
7	7,000	1,000	200	4.00	7.00	2.00

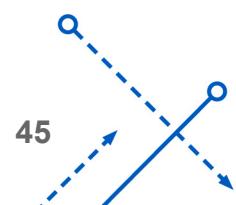
$$3\sigma_{\hat{\mathbf{r}}} = [1.26 \quad 0.25 \quad 0.51]^T \text{ km}$$

$$3\sigma_{\dot{\hat{\mathbf{r}}}} = [0.020 \quad 0.008 \quad 0.006]^T \text{ km/sec}$$



- Procedure

- Use the EKF to process the data forward with some initial condition guess *← May diverge, not robust as least squares*
- Then process the data backward to epoch
- Initial conditions for the state are then given by previous pass results
 - The backward pass uses the final state from the forward pass for its initial condition
- The covariance must be reset after each forward or backward pass
 - This is required since no “new” information is given with each pass
- The algorithm for orbit estimation is essentially equivalent to the nonlinear fixed-point smoother with a covariance reset
- Results show much better convergence properties than a nonlinear least squares approach



- Partialials
 - State matrix is given by

$$F = \begin{bmatrix} 0_{3 \times 3} & I_{3 \times 3} \\ F_{21} & 0_{3 \times 3} \end{bmatrix}$$

where

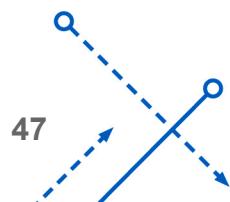
$$F_{21} = \begin{bmatrix} \frac{3\mu x^2}{\|\mathbf{r}\|^5} - \frac{\mu}{\|\mathbf{r}\|^3} & \frac{3\mu xy}{\|\mathbf{r}\|^5} & \frac{3\mu xz}{\|\mathbf{r}\|^5} \\ \frac{3\mu xy}{\|\mathbf{r}\|^5} & \frac{3\mu y^2}{\|\mathbf{r}\|^5} - \frac{\mu}{\|\mathbf{r}\|^3} & \frac{3\mu yz}{\|\mathbf{r}\|^5} \\ \frac{3\mu xz}{\|\mathbf{r}\|^5} & \frac{3\mu yz}{\|\mathbf{r}\|^5} & \frac{3\mu z^2}{\|\mathbf{r}\|^5} - \frac{\mu}{\|\mathbf{r}\|^3} \end{bmatrix}$$

- Output matrix

$$\frac{\partial \mathbf{h}}{\partial \mathbf{x}} = [H_{11} \quad 0_{3 \times 3}]$$

where

$$H_{11} = \begin{bmatrix} \frac{\partial ||\rho||}{\partial x} & \frac{\partial ||\rho||}{\partial y} & \frac{\partial ||\rho||}{\partial z} \\ \frac{\partial az}{\partial x} & \frac{\partial az}{\partial y} & \frac{\partial az}{\partial z} \\ \frac{\partial el}{\partial x} & \frac{\partial el}{\partial y} & \frac{\partial el}{\partial z} \end{bmatrix}$$



$$\frac{\partial \|\boldsymbol{\rho}\|}{\partial x} = (\rho_u \cos \phi \cos \Theta - \rho_e \sin \Theta - \rho_n \sin \phi \cos \Theta) / \|\boldsymbol{\rho}\|$$

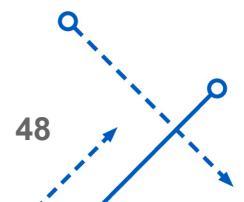
$$\frac{\partial \|\boldsymbol{\rho}\|}{\partial y} = (\rho_u \cos \phi \sin \Theta + \rho_e \cos \Theta - \rho_n \sin \phi \sin \Theta) / \|\boldsymbol{\rho}\|$$

$$\frac{\partial \|\boldsymbol{\rho}\|}{\partial z} = (\rho_u \sin \phi + \rho_n \cos \phi) / \|\boldsymbol{\rho}\|$$

$$\frac{\partial \text{az}}{\partial x} = \frac{1}{(\rho_n^2 + \rho_e^2)} (\rho_e \sin \phi \cos \Theta - \rho_n \sin \Theta)$$

$$\frac{\partial \text{az}}{\partial y} = \frac{1}{(\rho_n^2 + \rho_e^2)} (\rho_e \sin \phi \sin \Theta + \rho_n \cos \Theta)$$

$$\frac{\partial \text{az}}{\partial z} = -\frac{1}{(\rho_n^2 + \rho_e^2)} \rho_e \cos \phi$$



$$\frac{\partial \text{el}}{\partial x} = \frac{1}{||\boldsymbol{\rho}||(||\boldsymbol{\rho}||^2 - \rho_u^2)^{1/2}} \left(||\boldsymbol{\rho}|| \cos \phi \cos \Theta - \rho_u \frac{\partial ||\boldsymbol{\rho}||}{\partial x} \right)$$
$$\frac{\partial \text{el}}{\partial y} = \frac{1}{||\boldsymbol{\rho}||(||\boldsymbol{\rho}||^2 - \rho_u^2)^{1/2}} \left(||\boldsymbol{\rho}|| \cos \phi \sin \Theta - \rho_u \frac{\partial ||\boldsymbol{\rho}||}{\partial y} \right)$$
$$\frac{\partial \text{el}}{\partial z} = \frac{1}{||\boldsymbol{\rho}||(||\boldsymbol{\rho}||^2 - \rho_u^2)^{1/2}} \left(||\boldsymbol{\rho}|| \sin \phi - \rho_u \frac{\partial ||\boldsymbol{\rho}||}{\partial z} \right)$$

- Same starting conditions as least squares

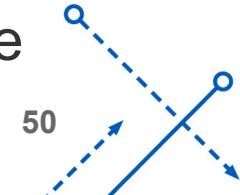
Iteration	Position (km)			Velocity (km/sec)		
0	6,990	1	1	1	1	1
1	7,121	1,046	192	-0.07	5.70	1.67
2	7,000	1,000	200	4.00	7.00	2.00
3	7,000	1,000	200	4.00	7.00	2.00

Identical results
and covariance
as least squares
at final iteration

$$3\sigma_{\hat{\mathbf{r}}} = [1.26 \quad 0.25 \quad 0.51]^T \text{ km}, \quad 3\sigma_{\dot{\mathbf{r}}} = [0.020 \quad 0.008 \quad 0.006]^T \text{ km/sec}$$

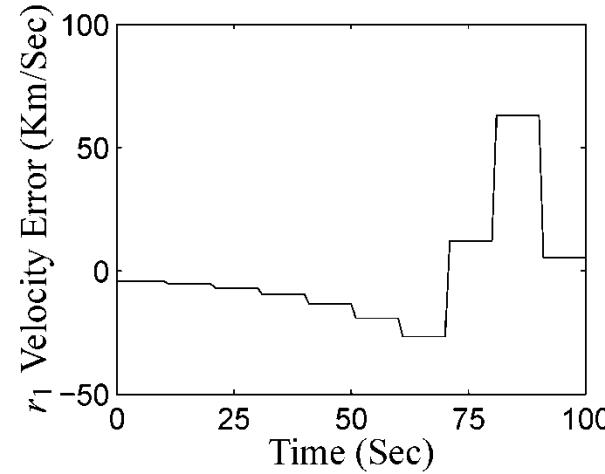
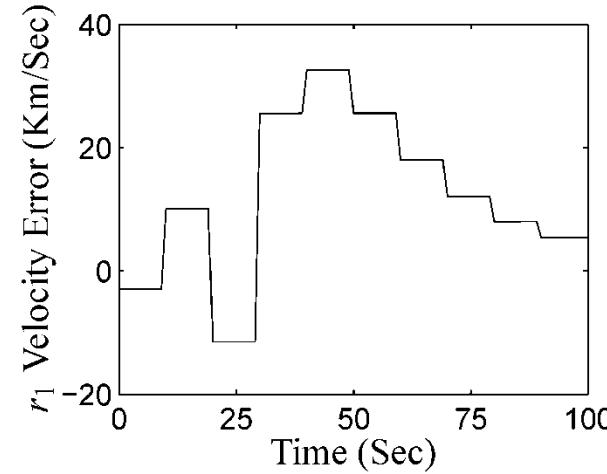
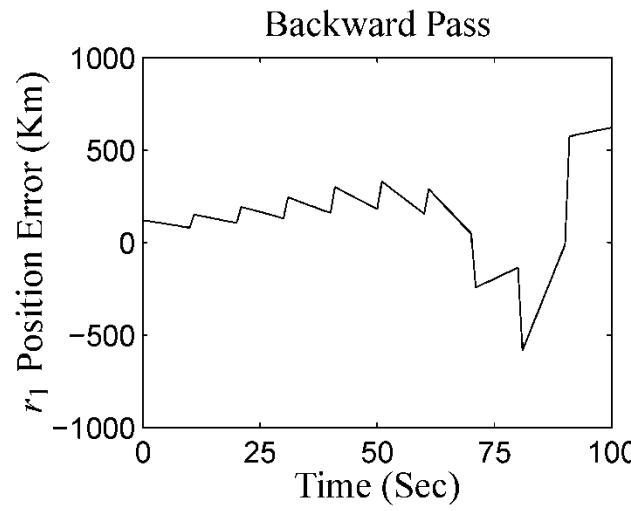
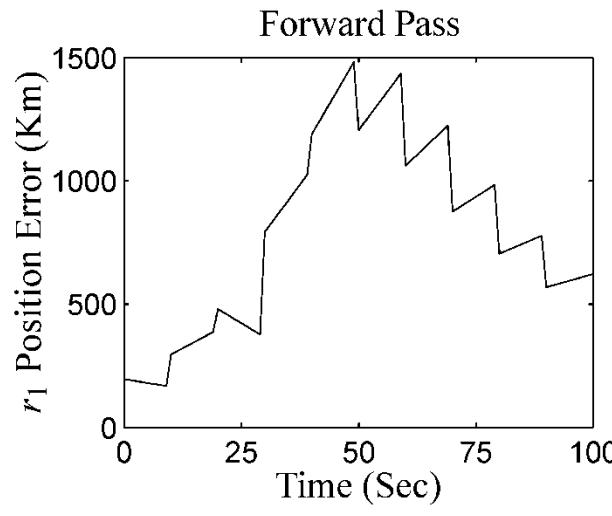
- Notes

- EKF can be executed in real-time to provide a navigation solution, i.e., it provides sequential estimates
- EKF can handle process noise, which is typically needed to handle unmodeled disturbances
- EKF must append state vector to estimate biases while least squares approach doesn't need to do this

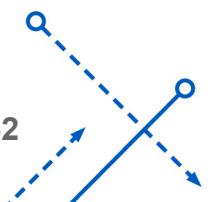


Example (ii)

First EKF Iteration



```
function [xe,xecov]=orbit_kal(x0,t0,tf,dt,y,tm,lat,sid,max,ymcov,p0,q);  
  
% [xe,xecov]=orbit_kal(x0,t0,tf,dt,y,tm,lat,dis,max,ymcov,p0,q);  
%  
% Determines the initial orbit conditions at Epoch using  
% an iterated Kalman filter approach. The inputs are:  
% x0 = initial guess (6x1)  
% t0 = initial time (1x1)  
% tf = final time (1x1)  
% dt = integration interval in sec. (1x1)  
% y = [range azimuth elevation] in km and rad (mx3)  
% tm = measurement times (assumed multiples of dt, and tm(1) = t0) (mx3)  
% lat = geocentric latitude of radar in rad (1x1)  
% sid = sidereal time in sec (mx1)  
% max = maximum number of iterations  
% ymcov = measurement covariance (3x3)  
% p0 = initial error covariance (6x6)  
% q = discrete process noise covariance (6x6)
```

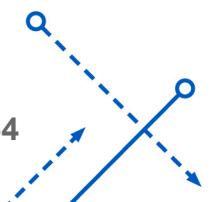


```
% Find Measurement Times  
t=[t0:dt:tf]';  
clear k;for i=1:length(tm), k(i)=find(t==tm(i));end;k=k(:);  
  
% Measurements  
ym=[y(:,1) y(:,2) y(:,3)];  
  
% Initialize  
m=length(t);  
ye1=zeros(length(y),1);ye2=zeros(length(y),1);ye3=zeros(length(y),1);  
mu=398600.64;  
pcov=p0;  
kkk=1;  
xe=zeros(length(t),6);xe(1,:)=x0(:)';  
pf=zeros(length(t),6);pb=zeros(length(t),6);  
pf(1,:)=diag(pcov)';  
clear xiter piter  
rearth=6378;  
phia=lat;theta=sid;
```



```
% Main Loop
for jjjj=1:max,
    for i=1:m-1;
        if (i==k(kkk)),
            xw=xe(i,:);

% Estimated Quantities
rhoud=cos(phiia)*cos(theta(kkk)).*(xw(1)-rearth*cos(phiia)*cos(theta(kkk))) ...
+cos(phiia)*sin(theta(kkk)).*(xw(2)-rearth*cos(phiia)*sin(theta(kkk))) ...
+sin(phiia)*(xw(3)-rearth*sin(phiia));
rhoed=-sin(theta(kkk)).*(xw(1)-rearth*cos(phiia)*cos(theta(kkk))) ...
+cos(theta(kkk)).*(xw(2)-rearth*cos(phiia)*sin(theta(kkk)));
rhond=-sin(phiia)*cos(theta(kkk)).*(xw(1)-rearth*cos(phiia)*cos(theta(kkk))) ...
-sin(phiia)*sin(theta(kkk)).*(xw(2)-rearth*cos(phiia)*sin(theta(kkk))) ...
+cos(phiia)*(xw(3)-rearth*sin(phiia));
rhod=norm([rhoud rhoed rhond]);
```



% Partials

```
drdx=(rhoud*cos(phi)*cos(theta(kkk))-rhoed*sin(theta(kkk)) ...  
-rhond*sin(phi)*cos(theta(kkk))/rhod;
```

```
drdy=(rhoud*cos(phi)*sin(theta(kkk))+rhoed*cos(theta(kkk)) ...  
-rhond*sin(phi)*sin(theta(kkk))/rhod;
```

```
drdz=(rhoud*sin(phi)+rhond*cos(phi))/rhod;
```

```
fac1=1/(1+(rhoed/rhond)^2);
```

```
dadx=fac1*(-sin(theta(kkk))+rhoed*sin(phi)*cos(theta(kkk))/rhond/rhond;
```

```
dady=fac1*(cos(theta(kkk))+rhoed*sin(phi)*sin(theta(kkk))/rhond/rhond;
```

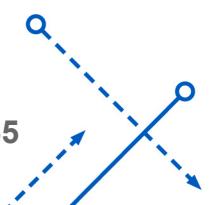
```
dadz=fac1*(-rhoed*cos(phi)/rhond/rhond);
```

```
fac2=1/sqrt(1-(rhoud/rhod)^2);
```

```
dedx=fac2*(cos(phi)*cos(theta(kkk))-rhoud*drdx/rhod/rhod;
```

```
dedy=fac2*(cos(phi)*sin(theta(kkk))-rhoud*drdy/rhod/rhod;
```

```
dedz=fac2*(sin(phi)-rhoud*drdz/rhod/rhod);
```



% Sensitivity

```
hpre=[drdx drdy drdz 0 0 0  
      dadx dady dadz 0 0 0  
      dedx dedy dedz 0 0 0];
```

% Gain

```
gain=pcov*hpre'*inv(hpre*pcov*hpre'+ymcov);
```

% Estimates

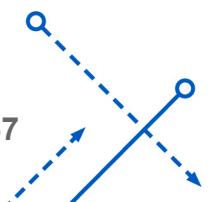
```
ye1=rhod;  
ye2=atan2(rhoed,rhond);  
ye3=asin(rhoud/rhod);  
if (ye2-y(kkk,2) > pi), ye2=ye2-2*pi; end  
if (y(kkk,2)-ye2 > pi), ye2=ye2+2*pi; end
```

```
ye=[ye1 ye2 ye3]';
```

```
% Unupdate
xe(i,:)=xe(i,:)+(gain*(ym(kkk,:)-ye))';
kkk=kkk+1;
pcov=(eye(6)-gain*hpre)*pcov;

end

% Propagate
re=norm(xe(i,1:3));
mu=398600.64;
asub=mu/re^5*[3*xe(i,1)^2-re^2 3*xe(i,1)*xe(i,2) 3*xe(i,1)*xe(i,3)
               3*xe(i,1)*xe(i,2) 3*xe(i,2)^2-re^2 3*xe(i,2)*xe(i,3)
               3*xe(i,1)*xe(i,3) 3*xe(i,2)*xe(i,3) 3*xe(i,3)^2-re^2];
biga=[zeros(3) eye(3);asub zeros(3)];
ad=c2d(biga,zeros(6,1),dt);
pcov=ad*pcov*ad'+q;
```



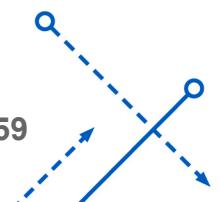
```
% Integrate
f1=dt*orbitfun(xe(i,:),mu);
f2=dt*orbitfun(xe(i,:)+0.5*f1',mu);
f3=dt*orbitfun(xe(i,:)+0.5*f2',mu);
f4=dt*orbitfun(xe(i,:)+f3',mu);
xe(i+1,:)=xe(i,:)+1/6*(f1'+2*f2'+2*f3'+f4');

pf(i+1,:)=diag(pcov)';
end

xf=xe;pb(m,:)=pf(m,:);pcov=p0;
kkk=kkk-1;
```

```
% Backwards
for i=m:-1:2,
if (i==k(kkk)),
% Estimate quantities
xw=xe(i,:);

rhoud=cos(phiia)*cos(theta(kkk)).*(xw(1)-rearth*cos(phiia)*cos(theta(kkk))) ...
+cos(phiia)*sin(theta(kkk)).*(xw(2)-rearth*cos(phiia)*sin(theta(kkk))) ...
+sin(phiia)*(xw(3)-rearth*sin(phiia));
rhoed=-sin(theta(kkk)).*(xw(1)-rearth*cos(phiia)*cos(theta(kkk))) ...
+cos(theta(kkk)).*(xw(2)-rearth*cos(phiia)*sin(theta(kkk)));
rhond=-sin(phiia)*cos(theta(kkk)).*(xw(1)-rearth*cos(phiia)*cos(theta(kkk))) ...
-sin(phiia)*sin(theta(kkk)).*(xw(2)-rearth*cos(phiia)*sin(theta(kkk))) ...
+cos(phiia)*(xw(3)-rearth*sin(phiia));
rhod=norm([rhoud rhoed rhond]);
```



% Partial Derivatives

```
drdx=(rhoud*cos(phi)*cos(theta(kkk))-rhoed*sin(theta(kkk)) ...  
-rhond*sin(phi)*cos(theta(kkk))/rhod;
```

```
drdy=(rhoud*cos(phi)*sin(theta(kkk))+rhoed*cos(theta(kkk)) ...  
-rhond*sin(phi)*sin(theta(kkk))/rhod;
```

```
drdz=(rhoud*sin(phi)+rhond*cos(phi))/rhod;
```

```
fac1=1/(1+(rhoed/rhond)^2);
```

```
dadx=fac1*(-sin(theta(kkk))+rhoed*sin(phi)*cos(theta(kkk))/rhond/rhond;
```

```
dady=fac1*(cos(theta(kkk))+rhoed*sin(phi)*sin(theta(kkk))/rhond/rhond;
```

```
dadz=fac1*(-rhoed*cos(phi)/rhond)/rhond;
```

```
fac2=1/sqrt(1-(rhoud/rhod)^2);
```

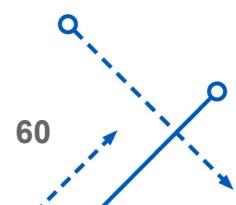
```
dedx=fac2*(cos(phi)*cos(theta(kkk))-rhoud*drdx/rhod)/rhod;
```

```
dehy=fac2*(cos(phi)*sin(theta(kkk))-rhoud*drdy/rhod)/rhod;
```

```
dedz=fac2*(sin(phi)-rhoud*drdz/rhod)/rhod;
```

% Sensitivity

```
hpre=[drdx drdy drdz 0 0 0; dadx dady dadz 0 0 0; dedxdehy dedz 0 0 0];
```



```
% Estimates  
ye1=rhod;  
ye2=atan2(rhoed,rhond);  
ye3=asin(rhoud/rhod);  
if (ye2-y(kkk,2) > pi), ye2=ye2-2*pi; end  
if (y(kkk,2)-ye2 > pi), ye2=ye2+2*pi; end
```

```
% Gain  
gain=pcov*hpre'*inv(hpre*pcov*hpre'+ymcov);
```

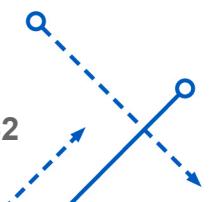
```
% Update  
ye=[ye1 ye2 ye3]';  
xe(i,:)=xe(i,:)+(gain*(ym(kkk,:)-ye))';  
kkk=kkk-1;  
pcov=(eye(6)-gain*hpre)*pcov;  
  
end
```

```
% Propagate
re=norm(xe(i,1:3));
mu=398600.64;
asub=mu/re^(5)*[3*xe(i,1)^2-re^2 3*xe(i,1)*xe(i,2) 3*xe(i,1)*xe(i,3)
                 3*xe(i,1)*xe(i,2) 3*xe(i,2)^2-re^2 3*xe(i,2)*xe(i,3)
                 3*xe(i,1)*xe(i,3) 3*xe(i,2)*xe(i,3) 3*xe(i,3)^2-re^2];
biga=[zeros(3) eye(3);asub zeros(3)];
biga=[zeros(3) eye(3);asub zeros(3)];

ad=c2d(-biga,zeros(6,1),dt);
pcov=ad*pcov*ad'+q;
pb(i-1,:)=diag(pcov)';

% Integrate
f1=-dt*orbitfun(xe(i,:),mu);
f2=-dt*orbitfun(xe(i,:)+0.5*f1',mu);
f3=-dt*orbitfun(xe(i,:)+0.5*f2',mu);
f4=-dt*orbitfun(xe(i,:)+f3',mu);
xe(i-1,:)=xe(i,:)+1/6*(f1'+2*f2'+2*f3'+f4');

end
```





Code (xii)

```
plast=pcov;
```

```
xb=xe;pcov=p0;
```

```
xiter(jjjj,:)=xe(1,:);
```

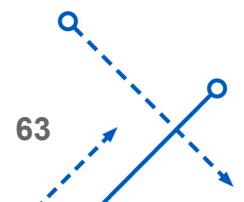
```
piter(jjjj,:)=pb(1,:);
```

```
xe(1,:)
```

```
end
```

```
xe=xiter;
```

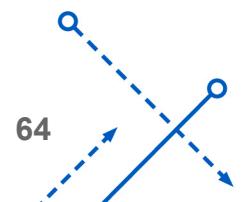
```
xecov=plast;
```



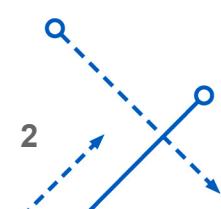


Code (xiii)

```
function f=orbitfun(x,mu);  
  
f=zeros(6,1);  
r=sqrt(x(1)^2+x(2)^2+x(3)^2);  
r3=r^3;  
f(1)=x(4);f(2)=x(5);f(3)=x(6);  
f(4)=-mu/r3*x(1);f(5)=-mu/r3*x(2);f(6)=-mu/r3*x(3);
```



- A *stochastic process* is simply a collection of random vectors defined on the same probability space
- Some basic definitions are now given
 - Let $\{\mathbf{x}(t_k)\}$ denote a sample function, which is a particular sequence of values taken as a result of an experiment
 - The variable $\mathbf{x}(t_k)$ is the random variable obtained at time t_k
 - Consider the pdf $p(\mathbf{x}(t_1), \mathbf{x}(t_2), \dots, \mathbf{x}(t_m))$
 - The mean $\mu(t_k)$ is denoted by $E\{\mathbf{x}(t_k)\}$
 - The autocorrelation is the set of quantities $E\{\mathbf{x}(t_i)\mathbf{x}^T(t_j)\}$
 - The covariance is defined by $E\{[\mathbf{x}(t_i) - \mu(t_i)][\mathbf{x}(t_j) - \mu(t_j)]^T\}$ for all t_i and t_j
 - Two processes, $\{\mathbf{x}(t_k)\}$ and $\{\mathbf{y}(t_k)\}$, are uncorrelated if $E\{\mathbf{x}(t_i)\mathbf{y}^T(t_j)\} = E\{\mathbf{x}(t_i)\}E\{\mathbf{y}^T(t_j)\}$ for all t_i and t_j
 - They are orthogonal if $E\{\mathbf{x}(t_i)\mathbf{y}^T(t_j)\} = 0$ for all t_i and t_j



- A process is said to be *stationary* if its random variable statistics do not vary in time, i.e., for arbitrary N we have

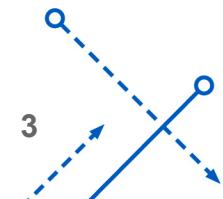
$$p(\mathbf{x}(t_1), \mathbf{x}(t_2), \dots, \mathbf{x}(t_m)) = p(\mathbf{x}(t_{1+N}), \mathbf{x}(t_{2+N}), \dots, \mathbf{x}(t_{m+N}))$$

- A process is *asymptotically stationary* if the following exists

$$\lim_{N \rightarrow \infty} p(\mathbf{x}(t_{1+N}), \dots, \mathbf{x}(t_{m+N}))$$

- A *wide-sense stationary process* exists if its first and second moments are invariant under time translation
- An *ergodic process* is a stationary process where the time averages can be replaced by an expectation
 - For Gaussian processes ergodicity is simply given by the following sufficient condition

$$\sum_{t_k=-\infty}^{+\infty} \|R(t_k)\| < \infty$$



- A *Markov process* is defined by the past having no influence on the present, which can be mathematically stated as

1st order

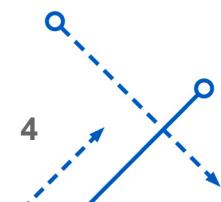
$$p(\mathbf{x}(t_1) | \mathbf{x}(t_2), \dots, \mathbf{x}(t_m)) = p(\mathbf{x}(t_1) | \mathbf{x}(t_2)) \quad (1)$$

- A second order Markov process is one where the most recent two pieces of information are all that affect the present
 - Right-hand side of (1) is replaced by $p(\mathbf{x}(t_1) | \mathbf{x}(t_2), \mathbf{x}(t_3))$
- A sequence of random variables \mathbf{x}_k converges to a random variable \mathbf{x} *with probability one* if every outcome is an event whose probability is one
- We now define convergence *in the mean sense*
 - A sequence of random variables \mathbf{x}_k is said to converge to \mathbf{x} in the mean square case if

$$\lim_{k \rightarrow \infty} E \{ \| \mathbf{x}_k - \mathbf{x} \|^2 \} = 0$$

- The function \mathbf{x} is called the limit in the mean and is often written as $\mathbf{x} = \text{l.i.m. } \mathbf{x}_k$, where l.i.m. is always defined as the limit in the mean square sense
- A random variable is *mean square continuous* if

$$\text{l.i.m. } \mathbf{x}(\tau) = \mathbf{x}(t) \quad \text{or} \quad \lim_{\tau \rightarrow t} \text{Tr} (E \{ [\mathbf{x}(\tau) - \mathbf{x}(t)][\mathbf{x}(\tau) - \mathbf{x}(t)]^T \}) = 0$$



- A transformation of the pdf is often done to help calculate the mean, covariance and higher moments
 - For a random variable \mathbf{x} the *moment generating function* is

$$M_{\mathbf{x}}(\mathbf{s}) = E \{ \exp(\mathbf{s}^T \mathbf{x}) \}$$

where $\mathbf{s} = [s_1 \ s_2 \ \cdots \ s_n]^T$ is a general vector

- The cross moments are defined by $E \{ x_1^{k_1} x_2^{k_2} \cdots x_n^{k_n} \}$, which can be computed by

$$E \{ x_1^{k_1} x_2^{k_2} \cdots x_n^{k_n} \} = \left. \frac{\partial^k M_{\mathbf{x}}(\mathbf{s})}{\partial s_1^{k_1} \cdots \partial s_n^{k_n}} \right|_{s_1=s_2=\cdots=s_n=0}$$

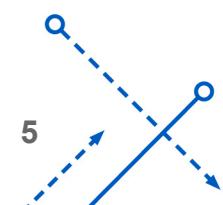
where $k = k_1 + k_2 + \cdots + k_n$

- For a scalar random variable x , the k^{th} -order moment is simply defined by

$$E \{ x^k \} = [d^k M_x(s)/ds^k]_{s=0}$$

- For a random vector \mathbf{x} , the second moment is simply

$$\nabla_{\mathbf{s}}^2 M_{\mathbf{x}}(\mathbf{s})|_{\mathbf{s}=0}$$



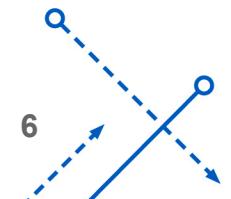
- The characteristic function, denoted by $\phi_{\mathbf{x}}(\mathbf{s})$, is related to the moment generating function by

$$\begin{aligned}\phi_{\mathbf{x}}(\mathbf{s}) &= M_{\mathbf{x}}(j \mathbf{s}) \\ &= E \{ j \exp(\mathbf{s}^T \mathbf{x}) \} \\ &= \int_{-\infty}^{\infty} j \exp(\mathbf{s}^T \mathbf{x}) p(\mathbf{x}) d\mathbf{x}\end{aligned}$$

where j is the imaginary unit with $j^2 = -1$

- Note that this function can also be viewed as the Fourier transform of the probability density function
- The pdf can be computed using the inverse Fourier transform

$$\begin{aligned}p(\mathbf{x}) &= \frac{1}{(2\pi)^n} \int_{-\infty}^{\infty} \exp(-j \mathbf{s}^T \mathbf{x}) \phi_{\mathbf{x}}(\mathbf{s}) ds \\ &= \int_{-\infty}^{\infty} \exp(-j 2\pi \mathbf{s}^T \mathbf{x}) \phi_{\mathbf{x}}(\mathbf{s}) ds\end{aligned}$$



- If \mathbf{x} is a random vector whose density function $p(\mathbf{x})$ is known, and if $\mathbf{y} = \mathbf{f}(\mathbf{x})$ is an arbitrary (generally nonlinear) one-to-one transformation, then it can be shown that the density function of \mathbf{y} is given by

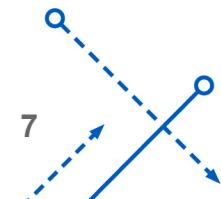
$$p(\mathbf{y}) = p(\mathbf{x}) \left| \det \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right) \right|^{-1} \quad (2)$$

with \mathbf{x} on the right-hand side given by

$$\mathbf{x} = \mathbf{f}^{-1}(\mathbf{y})$$

where $\mathbf{f}^{-1}(\mathbf{y})$ denotes the “reverse” relationship

- Thus, to convert the density function of \mathbf{x} to the density function of \mathbf{y} , simply write the density of \mathbf{x} in terms of \mathbf{y} and multiply by the inverse determinant of the Jacobian matrix
- Difficult part is to determine the inverse relationship
 - May not be possible



- Assume the following quadratic model

$$y = a x^2$$

- Note that for each value of y there are two x -values
- Assume that x has the following Gaussian density function

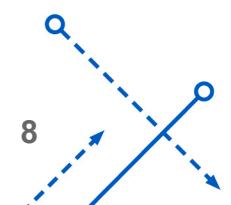
$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-x^2}{2\sigma^2}\right)$$

- When applying (2) we need to account for all multiplicities for multiple roots (this essentially means that the right-hand side of (2) needs to be multiplied by 2), which leads to

$$p(y) = \frac{1}{\sigma\sqrt{2\pi a y}} \exp\left(\frac{-y}{2a\sigma^2}\right), \quad \text{for } y > 0$$

and $p(y) = 0$ for $y < 0$

- Note that $p(y)$ is not Gaussian - PDF for $a x^2$



- Recall that the multidimensional case is given by

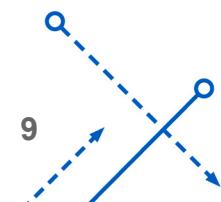
$$p(\mathbf{x}) = \frac{1}{[\det(2\pi R)]^{1/2}} \exp \left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T R^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right]$$

- Let's prove that its integral is one
 - First we need to prove the following “Gaussian Integral”

$$g \equiv \int_{-\infty}^{\infty} \exp \left(-\frac{1}{2} x^2 \right) dx = \sqrt{2\pi}$$

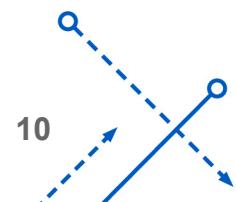
- Use Poisson's approach by squaring g to simplify the integral

$$\begin{aligned} \left[\int_{-\infty}^{\infty} \exp \left(-\frac{1}{2} x^2 \right) dx \right]^2 &= \int_{-\infty}^{\infty} \exp \left(-\frac{1}{2} x^2 \right) dx \int_{-\infty}^{\infty} \exp \left(-\frac{1}{2} y^2 \right) dy \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp \left[-\frac{1}{2} (x^2 + y^2) \right] dx dy \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp \left[-\frac{1}{2} (x^2 + y^2) \right] dy dx \end{aligned}$$



- Two basic ways to compute the integral
 - Cartesian approach
 - Polar approach
 - This approach requires the “Jacobian determinant,” which is easier than the Cartesian approach once the determinant is understood
 - This involves the transformation of variables approach
 - Choose to use the Cartesian approach here
- Cartesian approach goes back to Laplace (1812)
 - Let
- Since the limits on s as $y \rightarrow \pm\infty$ depend on the sign of x , use the fact that $\exp(-x^2/2)$ is an even function to simplify the calculation
 - Therefore, the integral over all real numbers is just twice the integral from zero to infinity

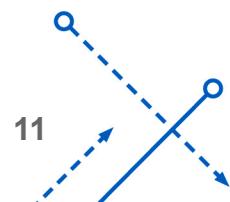
$$\int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}x^2\right) dx = 2 \int_0^{\infty} \exp\left(-\frac{1}{2}x^2\right) dx$$



- Then

$$\begin{aligned}
 g^2 &= 4 \int_0^\infty \int_0^\infty \exp\left[-\frac{1}{2}(x^2 + y^2)\right] dy dx \\
 &= 4 \int_0^\infty \left(\int_0^\infty \exp\left[-\frac{1}{2}x^2(1+s^2)\right] x ds \right) dx \quad \begin{matrix} \text{Substituted} \\ y = xs \\ dy = x ds \end{matrix} \\
 &= 4 \int_0^\infty \left(\int_0^\infty \exp\left[-\frac{1}{2}x^2(1+s^2)\right] x dx \right) ds \\
 &= 4 \int_0^\infty \left\{ -\frac{1}{(1+s^2)} \exp\left[-\frac{1}{2}x^2(1+s^2)\right] \right\} \Big|_{x=0}^{x=\infty} ds \\
 &= 4 \int_0^\infty -\frac{1}{(1+s^2)} (0-1) ds \\
 &= 4 \arctan(s) \Big|_{x=0}^{x=\infty} \\
 &= 4 \left(\frac{\pi}{2} - 0 \right) \\
 &= 2\pi
 \end{aligned}$$

- So $g = \sqrt{2\pi}$ ✓



- Take the eigenvalue/eigenvector decomposition of R^{-1}

$$R^{-1} = V\Lambda V^T, \quad \text{with} \quad \Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$$

- Define $\mathbf{z} \equiv \mathbf{x} - \boldsymbol{\mu}$ and $\mathbf{y} \equiv V^T \mathbf{z}$
- Then

↖ diagonal matrix of
eigenvalues

$$\begin{aligned} (\mathbf{x} - \boldsymbol{\mu})^T R^{-1} (\mathbf{x} - \boldsymbol{\mu}) &\equiv \mathbf{z}^T R^{-1} \mathbf{z} \\ &= \mathbf{z}^T V \Lambda V^T \mathbf{z} \\ &\equiv \mathbf{y}^T \Lambda \mathbf{y} \\ &= \sum_{i=1}^n \lambda_i y_i^2 \end{aligned}$$

- Also, using $\det(V\Lambda V^T) = \det(V)\det(\Lambda)\det(V^T)$, and since $\det(V) = \det(V^T) = \pm 1$, then it's obvious that

$$\det(R^{-1}) = \prod_{i=1}^n \lambda_i \quad \text{and} \quad \det(R) = \prod_{i=1}^n \lambda_i^{-1}$$



- Integrate the exponential term of the Gaussian pdf

$$\int_{-\infty}^{\infty} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T R^{-1}(\mathbf{x} - \boldsymbol{\mu})\right] d\mathbf{x} = \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}\mathbf{z}^T R^{-1}\mathbf{z}\right) d\mathbf{z}$$

- Now look at the density function transformation from \mathbf{z} to $\mathbf{y} = V^T \mathbf{z}$

$$p(\mathbf{y}) = p(\mathbf{z}) \left| \det \left(\frac{\partial V^T \mathbf{z}}{\partial \mathbf{z}} \right) \right|^{-1} = p(\mathbf{z}) |\det(V^T)|^{-1} = p(\mathbf{z}) |\pm 1|^{-1} = p(\mathbf{z})$$

- Then

$$\begin{aligned}
 \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}\mathbf{z}^T R^{-1}\mathbf{z}\right) d\mathbf{z} &= \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2} \sum_{i=1}^n \lambda_i y_i^2\right) dy \quad \text{Used variable transformation} \\
 &= \prod_{i=1}^n \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2} \lambda_i y_i^2\right) dy_i \\
 &= (2\pi)^{n/2} \prod_{i=1}^n \lambda_i^{-1/2} = [\det(2\pi R)]^{1/2} \quad \checkmark
 \end{aligned}$$

Used Gaussian Integral

Denominator of $p(\mathbf{x})$

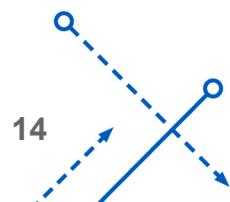
13

- Find the characteristic function
 - Begin with scalar normal distribution with $z_i \sim N(0, 1)$
 - Then

$$\begin{aligned}
 M_{z_i}(s_i) &= E\{e^{s_i z_i}\} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{s_i z_i} e^{-z_i^2/2} dz_i \\
 &= e^{s_i^2/2} \underbrace{\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-(z_i - s_i)^2/2} dz_i}_{N(s_i, 1)} = e^{s_i^2/2}
 \end{aligned}$$

- If $\mathbf{z} = (z_1, z_2, \dots, z_n)$ is a random sample with each z_i from $N(0, 1)$, then

$$\begin{aligned}
 M_{\mathbf{z}}(\mathbf{s}) &= E \left\{ \exp \left(\sum_{i=1}^n s_i z_i \right) \right\} = E \left\{ \prod_{i=1}^n \exp(s_i z_i) \right\} \stackrel{\text{ind}}{=} \prod_{i=1}^n E \{ \exp(s_i z_i) \} \\
 &= \prod_{i=1}^n M_{z_i}(s_i) = \exp \left\{ \sum_{i=1}^n s_i^2 / 2 \right\} = \exp \{ \mathbf{s}^T \mathbf{s} / 2 \}
 \end{aligned}$$

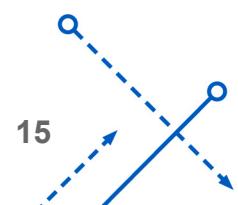


- From $R = \mathcal{L}\mathcal{L}^T$, then $\mathbf{x} = \mathcal{L}\mathbf{z} + \boldsymbol{\mu}$, so $E\{\mathbf{x}\} = \boldsymbol{\mu}$ and $\text{cov}\{\mathbf{x}\} = R$
- Then the moment generating function for a Gaussian is given by

$$\begin{aligned}
 M_{\mathbf{x}}(\mathbf{s}) &= E \left\{ \exp(\mathbf{s}^T \mathbf{x}) \right\} = E \left\{ \exp \left[\mathbf{s}^T (\mathcal{L} \mathbf{z} + \boldsymbol{\mu}) \right] \right\} \\
 &= \exp(\mathbf{s}^T \boldsymbol{\mu}) E \left\{ \exp \left(\mathbf{s}^T \mathcal{L} \mathbf{z} \right) \right\} \\
 &= \exp(\mathbf{s}^T \boldsymbol{\mu}) M_{\mathbf{z}}(\mathcal{L}^T \mathbf{s}) \\
 &= \exp(\mathbf{s}^T \boldsymbol{\mu}) \exp \left[\frac{1}{2} (\mathcal{L}^T \mathbf{s})^T (\mathcal{L}^T \mathbf{s}) \right] \\
 &= \exp(\mathbf{s}^T \boldsymbol{\mu}) \exp \left(\frac{1}{2} \mathbf{s}^T \mathcal{L} \mathcal{L}^T \mathbf{s} \right) \\
 &= \exp(\mathbf{s}^T \boldsymbol{\mu}) \exp \left(\frac{1}{2} \mathbf{s}^T R \mathbf{s} \right) \\
 &= \exp \left(\mathbf{s}^T \boldsymbol{\mu} + \frac{1}{2} \mathbf{s}^T R \mathbf{s} \right)
 \end{aligned}$$

- Characteristic function is then given by

$$\phi_{\mathbf{x}}(\mathbf{s}) = M_{\mathbf{x}}(j\mathbf{s}) = \exp \left(j\mathbf{s}^T \boldsymbol{\mu} - \frac{1}{2} \mathbf{s}^T R \mathbf{s} \right)$$

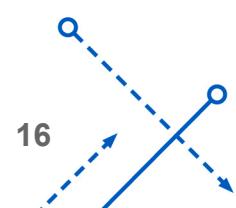
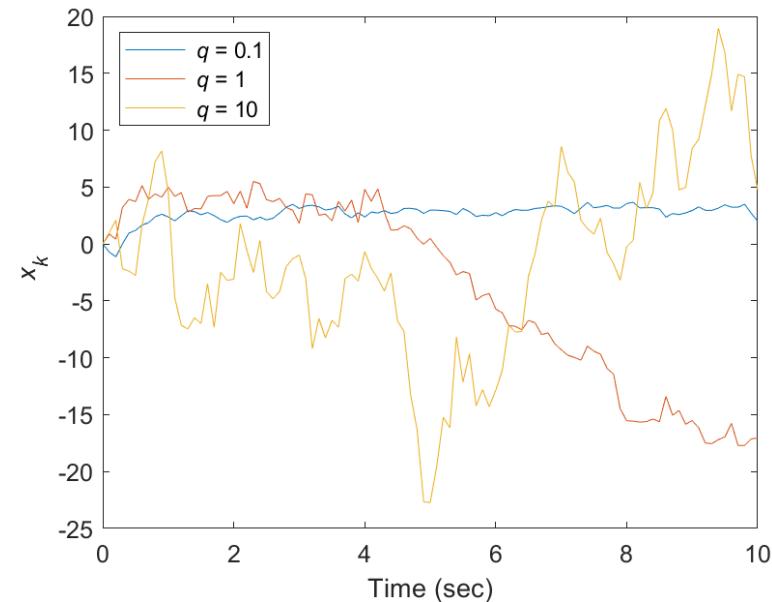


- A random walk process is defined as a process where the current value of a variable is composed of the past value plus an error term defined by a white-noise process
 - Consider the following discrete-time process

$$x_{k+1} = x_k + w_k, \quad w_k \sim N(0, q)$$

where w_k is a zero-mean Gaussian white-noise process

- This indicates that the change $x_{k+1} - x_k = w_k$ is a random process
- Thus, the best prediction of x for the next period is the current value
- It can be shown that the mean of a random walk process is constant but its variance is not
- Therefore, a random walk process is non-stationary and its variance increases with k



- The Wiener process is the limiting form of the random walk
 - This process, denoted by $\beta(t)$ for a single variable, has the following pdf $p(\beta(t)) = N(0, qt)$, where q is a constant
 - It relates to a zero-mean white noise, denoted by $w(t)$, through
$$\beta(t) = \int_0^t w(\tau) d\tau$$
where $E\{w(t_1) w(t_2)\} = q \delta(t_1 - t_2)$
 - Also, it can be shown that $E\{\beta(t_1) \beta(t_2)\} = q \min(t_1, t_2)$
- Assume that $q = 1$, and study the behavior of the Wiener process using the Fokker-Planck equation (more on this later)
 - The drift coefficient is zero, and the diffusion coefficient is 1
 - So the Fokker-Planck equation is given by

$$\frac{\partial}{\partial t} p(\beta(t)|\beta(t_0)) = \frac{1}{2} \frac{\partial^2}{\partial \beta^2} p(\beta(t)|\beta(t_0))$$

with initial condition $p(\beta(t_0)|\beta(t_0)) = \delta(\beta(t) - \beta(t_0))$



- This can be solved using the following characteristic function

$$\phi_\beta(s, t) = \int_{-\infty}^{\infty} \exp(js\beta) p(\beta(t)|\beta(t_0)) d\beta$$

which satisfies

$$\frac{\partial \phi}{\partial t} = -\frac{1}{2}s^2\phi$$

so that

$$\phi_\beta(s, t) = \exp\left[-\frac{1}{2}s^2(t - t_0)\right] \phi_\beta(s, t_0)$$

- From the initial condition we have $\phi_\beta(s, t_0) = \exp(js\beta(t_0))$ so that

$$\phi_\beta(s, t) = \exp\left[js\beta(t_0) - \frac{1}{2}s^2(t - t_0)\right]$$

- Using this equation the following pdf can be derived

$$p(\beta(t)|\beta(t_0)) = \frac{1}{\sqrt{2\pi(t - t_0)}} \exp\left[-\frac{1}{2}\frac{(\beta(t) - \beta(t_0))^2}{t - t_0}\right]$$



- This represents a Gaussian variable with

$$E \{ \beta(t) \} = \beta(t_0)$$

$$E \{ [\beta(t) - \beta(t_0)]^2 \} = t - t_0$$

- This indicates that an initially sharp distribution spreads in time
- The vector case with $\beta(t)$ has the following properties

$$E \{ \beta(t) \} = \beta(t_0)$$

$$E \{ [\beta_i(t) - \beta_i(t_0)][\beta_j(t) - \beta_j(t_0)] \} = (t - t_0)\delta_{ij}$$

- The one-variable Wiener process is often called *Brownian motion*. The Wiener process is a Markov process. This follows from the fact that it is the integral of white noise

$$\beta(t) = \beta(t_1) + \int_{t_1}^t w(\tau) d\tau$$

and $w(\tau)$, $\tau \in [t_1, t]$, is independent of $\beta(t_1)$.

- Furthermore, the state $x(t)$ of a time-varying dynamic system driven by white noise is also a Markov process



- The general vector case has independent Gaussian increments with

$$E \{ \beta(t_2) - \beta(t_1) \} = \mathbf{0}$$

$$E \{ [\beta(t_2) - \beta(t_1)][\beta(t_2) - \beta(t_1)]^T \} = \int_{t_1}^{t_2} Q(t) dt \quad (3)$$

where $Q(t)$ is a matrix

- Note that a Wiener process is continuous, but it is not differentiable

- This is easily seen by attempting to find the limit of $(x_{k+1} - x_k)/\Delta t = w_k/\Delta t$ as $\Delta t \rightarrow 0$ for the random walk process
 - Clearly, this is not possible because w_k is a random process
- Thus, writing the equation

~~Doesn't exist~~ $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), t) + G(\mathbf{x}(t), t) \mathbf{w}(t)$

is not rigorously correct because $\mathbf{w}(t)$ can only be considered as the hypothetical derivative of $\beta(t)$



- A stochastic integral can be defined through

$$\mathbf{I}(t) = \int_{t_0}^t A(\tau) d\beta(\tau) \quad (4)$$

- Now attempt to define the integral as a mean square limit of

$$\mathbf{s}_N = \sum_{i=1}^N A(\tau_i) [\beta(t_i) - \beta(t_{i-1})] \quad (5)$$

for $t_{i-1} \leq \tau_i \leq t_{i+1}$ when the partition $t_0 < t_1 < \dots < t_N$ of $[t_0, t]$ is refined so that $\Delta_N = \max_{1 \leq i \leq N} (t_i - t_{i-1}) \rightarrow 0$, i.e.

$$\mathbf{I}(t) = \lim_{\Delta_N \rightarrow 0} \mathbf{s}_N$$

- Before attempting to evaluate this limit, we discuss the *Lévy oscillation property* of Brownian motion
 - This is also known as the *quadratic variation property*
- Consider a unit-diffusion Brownian motion process, denoted by β , and the time partition $t_0 < t_1 < \dots < t_N = t$



- The sums

$$\xi_N = \sum_{i=1}^N [\beta(t_i) - \beta(t_{i-1})]^2 - (t - t_0)$$

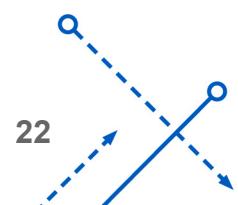
are random variables with means zero and variances given by $2 \sum_{i=1}^N (t_i - t_{i-1})^2$, which can be bounded by

$$2 \max_{1 \leq i \leq N} |t_i - t_{i-1}| \sum_{i=1}^N (t_i - t_{i-1}) = 2\Delta_N(t - t_0)$$

- Therefore, we now have $\underset{\Delta_N \rightarrow 0}{\text{l.i.m.}} \xi_N = 0$ and

$$\underset{\Delta_N \rightarrow 0}{\text{l.i.m.}} \sum_{i=1}^N [\beta(t_i) - \beta(t_{i-1})]^2 = t - t_0 \quad (6)$$

- Another way to state this equation is the symbolic notation $[d\beta(t)]^2 = dt$ with probability one in the mean square sense



- For the non-unit diffusion case, we simply have $[d\beta(t)]^2 = q(t) dt$, and for the general vector case we have

$$[d\beta(t) d\beta^T(t)] = Q(t) dt$$

- Using (3) we also have $E\{d\beta(t) d\beta^T(t)\} = Q(t) dt$
 - This is an important result, which states that not only is $E\{d\beta(t) d\beta^T(t)\} = Q(t) dt$ true but $[d\beta(t) d\beta^T(t)] = Q(t) dt$ itself is true for all samples except possibly a set of total probability zero
- Turn our attention now to the scalar version of (5)
 - Evaluating this equation where τ_i is any point on the interval $[t_{i-1}, t_i]$ will cause the value and properties of (4) to depend on the specific choice of τ_i
 - Thus, evaluation of truncated Taylor series expansions of nonlinear functions of $\beta(t)$ will invalidate the applicability of formal rules of differentials
 - To see this, assume that $A(\tau) = \beta(\tau)$ in (4) so that

$$\int_{t_0}^t \beta(\tau) d\beta(\tau) = \frac{1}{2} [\beta^2(t) - \beta^2(t_0)] \quad (7)$$



- This result is valid if the sums of the scalar version of (5), denoted by s_N , converge to a unique limit as $\Delta_N \rightarrow 0$ for any intermediate points τ_i
- Investigate a few useful choices for τ_i , letting $\tau_i = t_{i-1}$, then

$$\begin{aligned}
 s_N &= \sum_{i=1}^N \beta(t_{i-1})[\beta(t_i) - \beta(t_{i-1})] \\
 &= \sum_{i=1}^N \left[\frac{\beta(t_i) + \beta(t_{i-1})}{2} - \frac{\beta(t_i) - \beta(t_{i-1})}{2} \right] [\beta(t_i) - \beta(t_{i-1})] \\
 &= \frac{1}{2} \sum_{i=1}^N [\beta^2(t_i) - \beta^2(t_{i-1})] - \frac{1}{2} \sum_{i=1}^N [\beta(t_i) - \beta(t_{i-1})]^2 \\
 &= \frac{1}{2} [\beta^2(t) - \beta^2(t_0)] - \frac{1}{2} \sum_{i=1}^N [\beta(t_i) - \beta(t_{i-1})]^2
 \end{aligned}$$

- From (6) we have

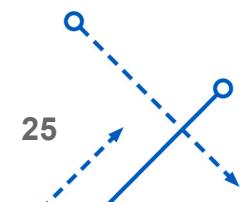
$$\text{l.i.m.}_{\Delta_N \rightarrow 0} s_N = \frac{1}{2} [\beta^2(t) - \beta^2(t_0)] - \frac{t - t_0}{2}$$



- Thus, using $\tau_i = t_{i-1}$ does not provide a convergence of the series to (7)
- Evaluating the series when $\tau_i = (1 - \theta)t_{i-1} + \theta t_i$ for $0 \leq \theta \leq 1$ gives

$$\text{l.i.m.}_{\Delta_N \rightarrow 0} s_N = \frac{1}{2}[\beta^2(t) - \beta^2(t_0)] + \left(\theta - \frac{1}{2}\right) \left(\frac{t - t_0}{2}\right)$$

- Setting $\tau_i = (t_{i-1} + t_i)/2$ does provide a convergence of the series to (7)
- This choice corresponds to the *Stratonovich integral*, while the choice $\tau_i = t_{i-1}$ corresponds to the *Itô integral*
- The Stratonovich integral has the advantage that formal rules of integration can be applied
- Despite this attractive feature, though, the Stratonovich integral lacks some important properties possessed by the Itô integral that are essential to Markov process descriptions
 - For example, the Itô integral efficiently uses the property that the Wiener process has independent increments



- The *stochastic differential* $d\mathbf{I}(t)$ of $\mathbf{I}(t)$ is now simply

$$d\mathbf{I}(t) = A(\tau) d\beta$$

- Gives us a mechanism to write a rigorously derived stochastic equation

$$d\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t), t) dt + G(\mathbf{x}(t), t) d\beta(t)$$

where $\beta(t)$ has diffusion strength $Q(t)$

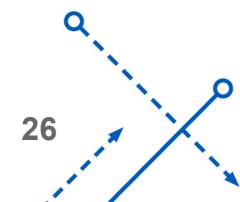
- This equation is often referred to as the Itô differential equation
- Consider the scalar version

$$dx(t) = f(x(t), t) dt + g(x(t), t) d\beta(t)$$

with diffusion strength $q(t)$

- Stratonovich's stochastic integral, denoted by the subscript S , is related to Itô's integral by

$$\left\{ \int_{t_0}^t a(\tau) d\beta(\tau) \right\}_S = \int_{t_0}^t a(\tau) d\beta(\tau) + \frac{1}{2} \int_{t_0}^t q(\tau) \frac{\partial a(\tau)}{\partial \beta} dt$$



- Defining $a(t) \equiv g(x(t), t)$ and since $\partial a / \partial \beta = (\partial g / \partial x) (\partial x / \partial \beta)$ we have the following equivalent Stratonovich differential equation

$$dx(t) = \left[f(x(t), t) - \frac{1}{2} q(t) g(x(t), t) \frac{\partial g(x(t), t)}{\partial x} \right] dt + g(x(t), t) d\beta(t)$$

- The multidimensional form can be written as

$$dx_i(t) = \left[f_i(\mathbf{x}(t), t) - \frac{1}{2} \mathbf{g}_i^T(\mathbf{x}(t), t) Q(t) \frac{\partial \mathbf{g}_i(\mathbf{x}(t), t)}{\partial x_i} \right] dt + \mathbf{g}_i^T(\mathbf{x}(t), t) d\beta(t)$$

for $i = 1, 2, \dots, n$, where $\mathbf{g}_i^T(\mathbf{x}(t), t)$ is the i^{th} row of $G(\mathbf{x}(t), t)$, and $f_i(\mathbf{x}(t), t)$ and $x_i(t)$ are the i^{th} elements of $\mathbf{f}(\mathbf{x}(t), t)$ and $\mathbf{x}(t)$, respectively

- Both the Itô and Stratonovich forms are equivalent when $G(\mathbf{x}(t), t)$ is not a function of \mathbf{x}



- Begin with the following stochastic differential equation

$$d\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t), t) dt + G(\mathbf{x}(t), t) d\beta(t) \quad (8)$$

- Again, the vector $\beta(t)$ represents Brownian motion of zero mean and diffusion $Q(t)$ so that

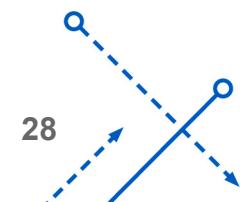
$$E \{ d\beta(t) d\beta^T(t) \} = Q(t) dt$$

$$E \{ [\beta(t) - \beta(\tau)][\beta(t) - \beta(\tau)]^T \} = \int_{\tau}^t Q(t) dt$$

- The solution can now be characterized in a form given by

$$\mathbf{x}(t) = \mathbf{x}(t_0) + \int_{t_0}^t \mathbf{f}(\mathbf{x}(\tau), \tau) d\tau + \int_{t_0}^t G(\mathbf{x}(\tau), \tau) \frac{d\beta(\tau)}{d\tau} d\tau \quad (9)$$

- The first integral is easily understood, but the second one is in the Itô form whose formal rules of integration and differentiation no longer apply, as established previously
- Itô calculus develops a more rigorous approach for differentiation and integration of moments for stochastic processes



- A good summary of the sufficient conditions for the existence and uniqueness of the solutions to (9) in the mean square sense is given by (similar to the ones for ordinary differential equations)

1. The functions $\mathbf{f}(\mathbf{x}(t), t)$ and $G(\mathbf{x}(t), t)$ are real functions that are uniformly *Lipschitz*. This means that a scalar k that is independent of time can be found such that

$$\begin{aligned} \|\mathbf{f}(\mathbf{x} + \Delta\mathbf{x}, t) - \mathbf{f}(\mathbf{x}, t)\| &\leq k\|\Delta\mathbf{x}\| \\ \|G(\mathbf{x} + \Delta\mathbf{x}, t) - G(\mathbf{x}, t)\|_F &\leq k\|\Delta\mathbf{x}\| \end{aligned}$$

for all \mathbf{x} and $\Delta\mathbf{x}$ and all t in the interval $[t_0, t_f]$ of interest.

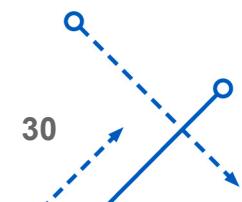
2. The functions $\mathbf{f}(\mathbf{x}(t), t)$ and $G(\mathbf{x}(t), t)$ are continuous in their second (time) argument over the interval $[t_0, t_f]$ of interest.
3. The functions $\mathbf{f}(\mathbf{x}(t), t)$ and $G(\mathbf{x}(t), t)$ are uniformly bounded according to $\|\mathbf{f}(\mathbf{x}, t)\| \leq k(1 + \|\mathbf{x}\|^2)$ and $\|G(\mathbf{x}, t)\|_F \leq k(1 + \|\mathbf{x}\|^2)$.
4. The vector $\mathbf{x}(t_0)$ is a random vector, with finite second moment, which is independent of the Brownian motion.



- In Itô's proof the solution for $\mathbf{x}(t)$ is given by assuming the existence of $\mathbf{x}_k(t)$ with $\mathbf{x}_k(t_0) = \mathbf{x}(t_0)$ and then forming $\mathbf{x}_{k+1}(t)$ through (9) with

$$\mathbf{x}_{k+1}(t) = \mathbf{x}(t_0) + \int_{t_0}^t \mathbf{f}(\mathbf{x}_k(\tau), \tau) d\tau + \int_{t_0}^t G(\mathbf{x}_k(\tau), \tau) \frac{d\beta(\tau)}{d\tau} d\tau$$

- If the four sufficient conditions are met, then the sequence of $\mathbf{x}_k(t)$ converges in the mean sense and with probability one on any finite interval $[t_0, t_f]$ to the solution $\mathbf{x}(t)$
- The solution $\mathbf{x}(t)$ has the following useful properties
 1. It is mean square continuous, i.e., $\lim_{\tau \rightarrow t} \mathbf{x}(\tau) = \mathbf{x}(t)$.
 2. The variables $\mathbf{x}(t) - \mathbf{x}(t_0)$ and $\mathbf{x}(t)$ are both independent of the future increments of $\beta(t)$.



3. It is a Markov process. Consider $t \geq t'$, so that

$$\mathbf{x}(t) = \mathbf{x}(t') + \int_{t'}^t \mathbf{f}(\mathbf{x}(\tau), \tau) d\tau + \int_{t'}^t G(\mathbf{x}(\tau), \tau) \frac{d\beta(\tau)}{d\tau} d\tau \quad (10)$$

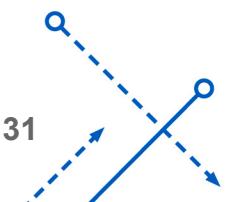
This clearly shows that $\mathbf{x}(t)$ depends on $\mathbf{x}(t')$ and $d\beta(\tau)$, $t' \leq \tau \leq t$, and the latter is independent of $\mathbf{x}(s)$ with $s \leq t'$. Thus, the conditional probability for $\mathbf{x}(t)$ given $\mathbf{x}(t')$ and $\mathbf{x}(s)$, $s \leq t'$, equals the distribution conditioned only on $\mathbf{x}(t')$. This proves that it is Markov.

4. The mean squared value of each component, i.e., $E \{x_i^2(t)\}$, is bounded by some finite value. Also, $\int_{t_0}^{t_f} E \{x_i^2(t)\} dt < \infty$.
5. The probability of a change in $\mathbf{x}(t)$ in a small interval Δt is of higher order than Δt :

$$\lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_{-\infty}^{\infty} p(\boldsymbol{\xi}(t + \Delta t) | \boldsymbol{\rho}(t)) d\boldsymbol{\xi} = 0 \quad (11)$$

$||\boldsymbol{\xi} - \boldsymbol{\rho}|| \geq \delta$

where the notation means that the integration over $\boldsymbol{\xi}$ is to be called outside the ball of radius δ about $\boldsymbol{\rho}$.



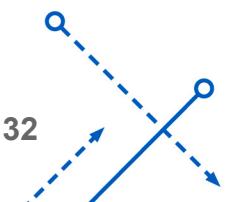
6. The *drift* of $\mathbf{x}(t)$ is $\mathbf{f}(\mathbf{x}(t), t)$. Using (8) we have

$$\begin{aligned}
 & \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_{-\infty}^{\infty} (\boldsymbol{\xi} - \boldsymbol{\rho}) p(\boldsymbol{\xi}(t + \Delta t) | \boldsymbol{\rho}(t)) d\boldsymbol{\xi} \\
 &= \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} E \{ \mathbf{x}(t + \Delta t) - \mathbf{x}(t) | \mathbf{x}(t) = \boldsymbol{\rho} \} \quad (12) \\
 &= \mathbf{f}(\boldsymbol{\rho}, t)
 \end{aligned}$$

This states that the mean rate of change in $\mathbf{x}(t)$ going from t to $t + \Delta t$ is $\mathbf{f}(\mathbf{x}(t), t)$ as $\Delta t \rightarrow \infty$.

7. The *diffusion* of $\mathbf{x}(t)$ is $G(\mathbf{x}(t), t) Q(t) G^T(\mathbf{x}(t), t)$:

$$\begin{aligned}
 & \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_{-\infty}^{\infty} (\boldsymbol{\xi} - \boldsymbol{\rho})(\boldsymbol{\xi} - \boldsymbol{\rho})^T p(\boldsymbol{\xi}(t + \Delta t) | \boldsymbol{\rho}(t)) d\boldsymbol{\xi} \\
 &= \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} E \{ [\mathbf{x}(t + \Delta t) - \mathbf{x}(t)][\mathbf{x}(t + \Delta t) - \mathbf{x}(t)]^T | \mathbf{x}(t) = \boldsymbol{\rho} \} \quad (13) \\
 &= G(\boldsymbol{\rho}, t) Q(t) G^T(\boldsymbol{\rho}, t)
 \end{aligned}$$



-
- 8. The higher-order infinitesimals in the progression of (11)-(13) are all zero:

$$\lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_{-\infty}^{\infty} (\xi_i - \rho_i)^k p(\boldsymbol{\xi}(t + \Delta t) | \boldsymbol{\rho}(t)) d\boldsymbol{\xi} = 0$$

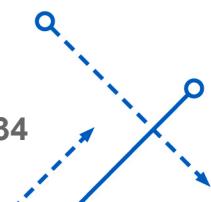
for $k \geq 2$. A similar relation exists for general products greater than second degree as well. This implies that the process does not diffuse “too fast.”

- Suppose we wish to apply Itô's calculus on a scalar function $\psi(\mathbf{x}(t), t)$ that has continuous first and second partial derivatives with respect to $\mathbf{x}(t)$ and is continuously differentiable with respect to time
 - As stated previously, formal rules of integration and differentiation are not valid in Itô's calculus
 - To develop a stochastic differential equation for $\psi(\mathbf{x}(t), t)$ the *Ito formula* must be used
 - First calculate

$$d\psi(\mathbf{x}(t), t) = \psi(\mathbf{x}(t) + d\mathbf{x}(t), t + dt) - \psi(\mathbf{x}(t), t)$$

- Now consider expanding the right-hand side using a Taylor series expansion, so that

$$d\psi(\mathbf{x}(t), t) = \frac{\partial\psi}{\partial t} dt + \frac{\partial\psi}{\partial\mathbf{x}^T} d\mathbf{x}(t) + \frac{1}{2} \frac{\partial^2\psi}{\partial t^2} dt^2 + \frac{1}{2} d\mathbf{x}^T(t) \frac{\partial^2\psi}{\partial\mathbf{x}\partial\mathbf{x}^T} d\mathbf{x}(t) + \dots$$



- Substituting (8) and retaining only terms up to first order in dt and second order in $d\beta$ yields

$$\begin{aligned} d\psi(\mathbf{x}(t), t) &= \frac{\partial\psi}{\partial t} dt + \frac{\partial\psi}{\partial\mathbf{x}^T} d\mathbf{x}(t) \\ &\quad + \frac{1}{2} \text{Tr} \left[G(\mathbf{x}(t), t) d\beta(t) d\beta^T(t) G^T(\mathbf{x}(t), t) \frac{\partial^2\psi}{\partial\mathbf{x}\partial\mathbf{x}^T} \right] \end{aligned}$$

where the identity $\text{Tr}(\mathbf{x}\mathbf{y}^T) = \mathbf{x}^T\mathbf{y}$ has been used

- Now, using the Levy property $[d\beta(t) d\beta^T(t)] = Q(t) dt$ gives

$$\begin{aligned} d\psi(\mathbf{x}(t), t) &= \frac{\partial\psi}{\partial t} dt + \frac{\partial\psi}{\partial\mathbf{x}^T} d\mathbf{x}(t) \\ &\quad + \frac{1}{2} \text{Tr} \left[G(\mathbf{x}(t), t) Q(t) G^T(\mathbf{x}(t), t) \frac{\partial^2\psi}{\partial\mathbf{x}\partial\mathbf{x}^T} \right] dt \quad (14) \end{aligned}$$

- Equation (8) is often combined with (14) and written in the form

$$d\psi(\mathbf{x}(t), t) = \frac{\partial \psi}{\partial t} dt + \mathcal{L}[\psi(\mathbf{x}(t), t)] dt + \frac{\partial \psi}{\partial \mathbf{x}^T} G(\mathbf{x}(t), t) d\beta(t) \quad (15)$$

where

$$\mathcal{L}[\psi(\mathbf{x}(t), t)] \equiv \frac{\partial \psi}{\partial \mathbf{x}^T} \mathbf{f}(\mathbf{x}(t), t) + \frac{1}{2} \text{Tr} \left[G(\mathbf{x}(t), t) Q(t) G^T(\mathbf{x}(t), t) \frac{\partial^2 \psi}{\partial \mathbf{x} \partial \mathbf{x}^T} \right]$$

- The term $\mathcal{L}[\psi(\mathbf{x}(t), t)]$ is the differential generator of the process

- Consider the following system

$$dx(t) = d\beta(t)$$

with $q(t)$ being a constant denoted by q

- This equation states that $x(t)$ is itself Brownian motion, which is heuristically written as $\dot{x}(t) = w(t)$
- Now consider the following nonlinear function

$$\psi(x(t), t) = e^{x(t)} = e^{\beta(t)}$$

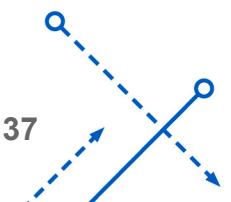
- From (15) this satisfies the following stochastic differential equation

$$d\psi(x(t), t) = e^{x(t)} dx(t) + \frac{1}{2} q e^{x(t)} dt$$

or

$$d \left[e^{\beta(t)} \right] = e^{\beta(t)} d\beta(t) + \frac{1}{2} q e^{\beta(t)} dt$$

- Because of the last term, this does not satisfy formal rules for differentials



- This can be overcome by defining $\gamma(t) \equiv e^{\beta(t)}$, which yields

$$d\gamma(t) = \frac{1}{2}q \gamma(t) dt + \gamma(t) d\beta(t)$$

with $\gamma(t_0) = 1$ with probability one

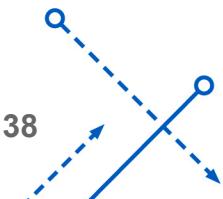
- This is now the appropriate stochastic differential equation in the form of (8) to yield a solution in the form of $e^{\beta(t)}$, since $\beta(t_0) = 0$ with probability one
- This example clearly shows that stochastic differential equations do not obey formal rules of integration either
- The differential equation that would have to be proposed by formal rules is

$$dz(t) = z(t) d\beta(t)$$

with $z(t_0) = 1$ with probability one

- The solution for this equation with $t_0 = 0$ using (15) is given by

$$z(t) = e^{\beta(t)-qt/2}$$



- All the material to this point has been leading to answering the question “how does the pdf propagate in time?” for Gaussian inputs into a nonlinear system
 - The answer lies in the *Fokker-Planck equation*, also known as the *forward Kolmogorov equation*
 - To derive this equation, begin with the scalar version of (8)

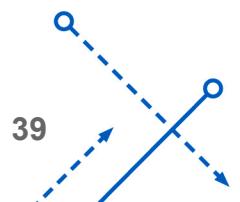
$$dx(t) = f(x(t), t) dt + g(x(t), t) d\beta(t)$$

- From the Itô formula in (15), dropping the explicit notation for time and state dependence for now, we have

$$d\psi = \left[\frac{\partial \psi}{\partial t} + f \frac{\partial \psi}{\partial x} + \frac{1}{2} g^2 q \frac{\partial^2 \psi}{\partial x^2} \right] dt + g \frac{\partial \psi}{\partial x} d\beta$$

- Taking the expectation of both sides yields

$$\frac{d}{dt} E \{ \psi \} = E \left\{ f \frac{\partial \psi}{\partial x} + \frac{1}{2} g^2 q \frac{\partial^2 \psi}{\partial x^2} \right\}$$



- Using the definition of expectation gives

$$\frac{d}{dt} \int_{-\infty}^{\infty} \psi p(x(t)|x(t')) dx = \int_{-\infty}^{\infty} \left[f \frac{\partial \psi}{\partial x} + \frac{1}{2} g^2 q \frac{\partial^2 \psi}{\partial x^2} \right] p(x(t)|x(t')) dx$$

where $p(x(t)|x(t'))$ denotes the conditional probability of $x(t)$ given $x(t')$ with $t > t'$

- Integrating by parts, and using $p \equiv p(x(t)|x(t'))$, yields

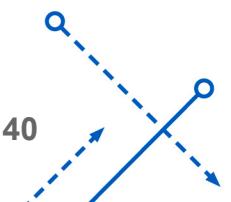
$$\int_{-\infty}^{\infty} \psi \frac{\partial p}{\partial t} dx = \int_{-\infty}^{\infty} \left[-\frac{\partial f p}{\partial x} + \frac{1}{2} \frac{\partial g^2 p}{\partial x^2} \right] \psi dx$$

when $p(x(t)|x(t'))$ and $\partial p(x(t)|x(t'))/\partial x$ vanish as $x \rightarrow \pm\infty$

- Since ψ is arbitrary, and simplifying the notation for $p(x(t)|x(t'))$ to be just $p(x(t), t)$, we now have

$$\frac{\partial p(x(t), t)}{\partial t} = -\frac{\partial}{\partial x} [f(x(t), t) p(x(t), t)] + \frac{1}{2} \frac{\partial^2}{\partial x^2} [g^2(x(t), t) p(x(t), t)]$$

- This is the Fokker-Planck equation for scalar systems



- The general Fokker-Planck equation for multidimensional systems is given by

$$\begin{aligned} \frac{\partial}{\partial t} p(\mathbf{x}(t), t) = & - \sum_{i=1}^n \frac{\partial}{\partial x_i} [f_i(\mathbf{x}(t), t) p(\mathbf{x}(t), t)] \\ & + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \frac{\partial^2}{\partial x_i \partial x_j} \left[\{G(x(t), t) Q(t) G^T(x(t), t)\}_{ij} p(\mathbf{x}(t), t) \right] \end{aligned}$$

where $f_i(\mathbf{x}(t), t)$ is the i^{th} element of $\mathbf{f}(\mathbf{x}(t), t)$, and $\{G Q G^T\}_{ij}$ is the ij^{th} element element of $G Q G^T$

- This equation only has a closed-form solution for a small number of cases
- Note that this form uses the Itô interpretation, which is commonly used among mathematicians

- The Stratonovich interpretation uses the following replacement

$$f_i(\mathbf{x}(t), t) \leftarrow f_i(\mathbf{x}(t), t) - \frac{1}{2} \mathbf{g}_i^T(\mathbf{x}(t), t) Q(t) \frac{\partial \mathbf{g}_i(\mathbf{x}(t), t)}{\partial x_i}$$

where \leftarrow denotes replacement, and where $\mathbf{g}_i^T(\mathbf{x}(t), t)$ is the i^{th} row of $G(\mathbf{x}(t), t)$

- The Stratonovich interpretation is more popular among engineers and physicists because standard rules of integration can be applied
 - For most systems of interest to engineers, the matrix G is usually not a function of $\mathbf{x}(t)$ so the interpretation issue is usually not a concern

- Consider the following first order system

$$dx(t) = f x(t) dt + d\beta(t)$$

or written heuristically as $\dot{x}(t) = f x(t) + w(t)$, with $E\{w(t)\} = 0$ and $E\{w(t)w(\tau)\} = q(t) \delta(t - \tau)$

- The Fokker-Planck equation is given with $f(x(t), t) = fx(t)$ and $g(x(t), t) = 1$
 - Note that the Itô and Stratonovich interpretations are equivalent in this case
- Assume a Gaussian solution with

$$p(\mathbf{x}(t), t) = \frac{1}{\sqrt{2\pi p(t)}} \exp \left\{ -\frac{[x(t) - \mu(t)]^2}{2 p(t)} \right\}$$

where $\mu(t)$ is the mean and $p(t)$ is the variance

- The Fokker-Planck equation now becomes

$$\begin{aligned} & \frac{1}{2} p^{-3/2}(t) \{ p^{-1}(t) [x(t) - \mu(t)]^2 - 1 \} \dot{p}(t) + p^{-3/2}(t) [x(t) - \mu(t)] \dot{\mu}(t) \\ &= -f p^{-1/2}(t) + f x(t) [x(t) - \mu(t)] p^{-3/2}(t) + \frac{1}{2} q(t) p^{-3/2}(t) \{ p^{-1}(t) [x(t) - \mu(t)]^2 - 1 \} \end{aligned}$$

- Equating terms independent of $[x(t) - \mu(t)]$ yields

$$\dot{p}(t) = 2f p(t) + q(t)$$

which is exactly the Kalman covariance propagation equation

- Equating terms that depend on $[x(t) - \mu(t)]$ yields

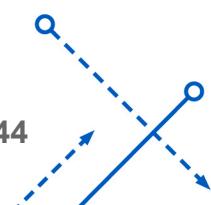
$$\dot{\mu}(t) + \frac{1}{2} \frac{x(t) - \mu(t)}{p(t)} \dot{p}(t) = f x(t) + q(t) \frac{1}{2} \frac{x(t) - \mu(t)}{p(t)}$$

- Substituting $\dot{p}(t) = 2f p(t) + q(t)$ yields

$$\dot{\mu}(t) = f \mu(t)$$

which is exactly the Kalman propagation equation

- This clearly shows that the scalar Kalman propagation equations are consistent with the Fokker-Planck equation₄₄



- The equation by Kushner modifies the Fokker-Planck equation with continuous measurements
 - Note that another form was introduced by Zakai
 - However, Kushner's equation is a nonlinear stochastic partial differential equation and satisfies the normalization requirement for a pdf, while Zakai's equation is a linear stochastic partial differential equation for the un-normalized pdf
 - The continuous-time measurements are modeled using the process described by

$$d\mathbf{z}(t) = \mathbf{h}(\mathbf{x}(t), t) dt + d\mathbf{b}(t)$$

where $\mathbf{b}(t)$ is Brownian motion, independent of $\beta(t)$ from (8), with diffusion given by

$$E \{ d\mathbf{b}(t) d\mathbf{b}^T(t) \} = R(t) dt$$

- This corresponds heuristically to

$$\dot{\mathbf{z}}(t) \equiv \mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), t) + \mathbf{v}(t)$$

with $E\{\mathbf{v}(t) \mathbf{v}^T(\tau)\} = R(t) \delta(t - \tau)$



- We wish to establish the equation for the time history of the conditional density of the state $\mathbf{x}(t)$, but now conditioned on the entire history of measurements observed up to time t
- The conditional density now becomes

$$p(\mathbf{x}(t), t | \mathbf{z}(\tau), t_0 \leq \tau \leq t)$$

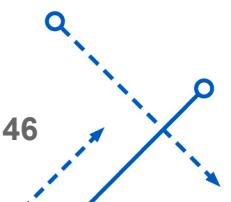
or in simple shorthand notation $p(\mathbf{x}|\mathbf{z})$

- The conditional density satisfies the Kushner equation, which is given by

$$\frac{\partial p(\mathbf{x}|\mathbf{z})}{\partial t} = \mathcal{L}(\mathbf{x}|\mathbf{z}) + [\mathbf{h}(\mathbf{x}(t), t) - \mathbf{m}(\mathbf{x}(t), t)]^T R^{-1}(t) [\mathbf{y}(t) - \mathbf{m}(\mathbf{x}(t), t)] p(\mathbf{x}|\mathbf{z})$$

where

$$\mathbf{m}(\mathbf{x}(t), t) \equiv \int_{-\infty}^{\infty} \mathbf{h}(\mathbf{x}(t), t) p(\mathbf{x}|\mathbf{z}) d\mathbf{x}$$



and

$$\begin{aligned}\mathcal{L}(\mathbf{x}|\mathbf{z}) \equiv & - \sum_{i=1}^n \frac{\partial}{\partial x_i} [f_i(\mathbf{x}(t), t) p(\mathbf{x}|\mathbf{z})] \\ & + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \frac{\partial^2}{\partial x_i \partial x_j} \left[\{G(x(t), t) Q(t) G^T(x(t), t)\}_{ij} p(\mathbf{x}|\mathbf{z}) \right]\end{aligned}$$

- Note that this equation corresponds to the Fokker-Planck equation
- If no measurement information exists, i.e. $R^{-1}(t) = 0$, then the Kushner equation reduces directly to the Fokker-Planck equation

Author(s) (year)	Method	Solution	Comments
Kolmogorov (1941)	innovations	exact	L, S
Wiener (1942)	spectral factorization	exact	L, S, IM
Levinson (1947)	lattice filter	approximate	L, S, FM
Bode & Shannon (1950)	innovations, whitening	exact	L, S
Zadeh & Ragazzini (1950)	innovations, whitening	exact	L, NS
Kalman (1960)	orthogonal projection	exact	LQG, NS, D
Kalman & Bucy (1961)	recursive Riccati	exact	LQG, NS, C
Stratonovich (1960)	conditional Markov	exact	NL, NS
Kushner (1967)	PDE	exact	NL, NS
Zakai (1969)	PDE	exact	NL, NS
Handschin & Mayne (1969)	Monte Carlo	approximate	NL, NS, NG
Bucy & Senne (1971)	point-mass, Bayes	approximate	NL, NS, NG
Kailath (1971)	innovations	exact	L, NS, NG
Beneš (1981)	Beneš	ES of Zakai Equation	NL, FD
Daum (1986)	virtual measurement	ES of FPK Equation	NL, FD
Gordon, Salmond, & Smith (1993)	bootstrap	approximate	NL, NS, NG
Julier & Uhlmann (1997)	unscented transform	approximate	NL, NG

LQG denotes “linear quadratic-Gaussian,” PDE denotes “partial differential equation,” FPK denotes “Fokker-Planck-Kolmogorov,” ES denotes “exact solution,” L denotes “linear,” S denotes “stationary,” NS denotes “non-stationary,” C denotes “continuous,” D denotes “discrete,” IM denotes “infinite memory,” FM denotes “finite memory,” NG denotes “non-Gaussian,” NL denotes “nonlinear,” and FD denotes “finite-dimensional”

