

FILTROS de KALMAN POLINOMIALES

Para aplicar la teoría del Filtro de Kalman, nuestro modelo de mundo real se debe describir mediante un conjunto de ecuaciones diferenciales en forma matricial o de espacio de estados.

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{u} + \mathbf{w}$$

\mathbf{x} es un vector columna con los estados del sistema

\mathbf{F} es la matriz de la dinámica del sistema

\mathbf{u} es un vector conocido, llamado vector de control

\mathbf{w} es un proceso de ruido blanco expresado en forma de vector

Hay una matriz de ruido de proceso \mathbf{Q} que se relaciona con el vector de ruido de proceso \mathbf{w} según

$$\mathbf{Q} = E[\mathbf{w}\mathbf{w}^T]$$

Aunque el ruido de proceso no tenga un significado físico se utiliza como dispositivo para decirle al filtro lo impreciso que es nuestro conocimiento del mundo real.

La formulación del filtro de Kalman requiere que las mediciones estén linealmente relacionadas con los estados según

$$z = Hx + v$$

donde

z es el vector de medida

H es la matriz de medida

v es ruido gaussiano de medida en forma de vector

La matriz R de ruido de medida se relaciona con el vector de ruido de medida v según

$$R = E[vv^T]$$

Todas estas relaciones tendrán que ser discretizadas antes de construir el filtro de Kalman.

Si tomamos medidas cada T_s segundos, primero tendremos que encontrar la matriz fundamental Φ .

Para sistemas invariantes en el tiempo la matriz fundamental se puede encontrar a partir de la matriz de la dinámica del sistema según

$$\Phi(t) = \mathcal{L}^{-1}[(sI - F)^{-1}]$$

donde

I es la matriz identidad

\mathcal{L}^{-1} es la transformada inversa de Laplace

F es la matriz de la dinámica del sistema

Otra forma de encontrar la matriz fundamental es calculando el desarrollo en serie de Taylor

$$\Phi(t) = e^{Ft} = I + Ft + \frac{(Ft)^2}{2!} + \dots + \frac{(Ft)^n}{n!} + \dots$$

La matriz fundamental discreta o matriz de transición se obtiene fácilmente evaluando la matriz fundamental en los instantes de muestreo T_s

$$\Phi_k = \Phi(T_s)$$

La forma discreta de la ecuación de medida es

$$z_k = Hx_k + v_k$$

y

$$R_k = E(v_k v_k^T)$$

donde

R_k es una matriz formada por las varianzas de cada fuente de ruido de medida

La ecuación resultante de filtrado kalman está dada por

$$\hat{\mathbf{x}}_k = \Phi_k \hat{\mathbf{x}}_{k-1} + \mathbf{G}_k \mathbf{u}_{k-1} + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H} \Phi_k \hat{\mathbf{x}}_{k-1} - \mathbf{H} \mathbf{G}_k \mathbf{u}_{k-1})$$

donde

\mathbf{K}_k representa la matriz de ganancia Kalman

\mathbf{G}_k se obtiene como

$$\mathbf{G}_k = \int_0^{T_s} \Phi(\tau) \mathbf{G} \, d\tau$$

si se asume que \mathbf{u}_{k-1} es constante entre instantes de muestreo.

Las ganancias Kalman se calculan mediante las ecuaciones recursivas de Riccati mientras el filtro está funcionando

$$\begin{aligned} \mathbf{M}_k &= \Phi_k \mathbf{P}_{k-1} \Phi_k^T + \mathbf{Q}_k \\ \mathbf{K}_k &= \mathbf{M}_k \mathbf{H}^T (\mathbf{H} \mathbf{M}_k \mathbf{H}^T + \mathbf{R}_k)^{-1} \\ \mathbf{P}_k &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \mathbf{M}_k \end{aligned}$$

donde

\mathbf{P}_k es la matriz de covarianza que representa los errores de los entados estimados (esto es, varianza de valor real menos valor estimado) despues de actualizar.

\mathbf{M}_k es la matriz de covarianza representando los errores de los estados estimados antes de actualizar.

La matriz discreta del ruido de proceso \mathbf{Q}_k se puede encontrar a partir de la matriz continua del ruido de proceso \mathbf{Q} y la matriz fundamental según

$$\mathbf{Q}_k = \int_0^{T_s} \Phi(\tau) \mathbf{Q} \Phi^T(\tau) d\tau.$$

Para a utilizar las ecuaciones de Riccati se requiere de una matriz de inicial de covarianza \mathbf{P}_0 .

Obtención de las Ecuaciones de Riccati en forma Escalar

Si no hay perturbación determinista o vector de control, la forma que toma el filtrado de Kalman es

$$\hat{\mathbf{x}}_k = \Phi_k \hat{\mathbf{x}}_{k-1} + \mathbf{K}_k (z_k - \mathbf{H} \Phi_k \hat{\mathbf{x}}_{k-1})$$

mientras que las ecuaciones de Riccati siguen siendo

$$\begin{aligned} \mathbf{M}_k &= \Phi_k \mathbf{P}_{k-1} \Phi_k^T \\ \mathbf{K}_k &= \mathbf{M}_k \mathbf{H}^T (\mathbf{H} \mathbf{M}_k \mathbf{H}^T + \mathbf{R}_k)^{-1} \\ \mathbf{P}_k &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \mathbf{M}_k \end{aligned}$$

La obtención de estas ecuaciones es exageradamente complicada siendo bastante simple para el caso escalar.

Si no hay perturbación determinista o vector de control, nuestro modelo discreto de mundo real está dado por

$$x_k = \Phi_k x_{k-1} + w_k$$

donde Φ_k es el escalar que propaga los estados desde un instante de muestreo al siguiente, y w_k es ruido blanco de proceso.

A partir de

$$\hat{\mathbf{x}}_k = \Phi_k \hat{\mathbf{x}}_{k-1} + \mathbf{K}_k (z_k - \mathbf{H} \Phi_k \hat{\mathbf{x}}_{k-1})$$

podemos ver que la ecuación escalar de filtrado Kalman es

$$\hat{x}_k = \Phi_k \hat{x}_{k-1} + K_k (z_k - H \Phi_k \hat{x}_{k-1})$$

La ecuación de medida es

$$z_k = Hx_k + v_k$$

donde z_k es la medida, H es el escalar que relaciona el estado con la medida y v_k es el ruido de medida.

De las ecuaciones anteriores se tiene que el error en la estimación es

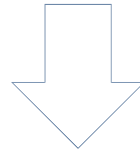
$$\tilde{\mathbf{x}}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k = \mathbf{x}_k - \Phi_k \hat{\mathbf{x}}_{k-1} - \mathbf{K}_k (z_k - \mathbf{H} \Phi_k \hat{\mathbf{x}}_{k-1})$$

Poniendo la medida en términos del estado se tiene

$$\tilde{\mathbf{x}}_k = \mathbf{x}_k - \Phi_k \hat{\mathbf{x}}_{k-1} - \mathbf{K}_k (H\mathbf{x}_k + v_k - \mathbf{H} \Phi_k \hat{\mathbf{x}}_{k-1})$$

Si se utiliza la forma alternativa de expresar un estado, en lugar de en el instante k , en el instante $k - 1$, tendremos

$$\tilde{x}_k = x_k - \Phi_k \hat{x}_{k-1} - K_k (Hx_k + v_k - H\Phi_k \hat{x}_{k-1})$$



$$x_k = \Phi_k x_{k-1} + w_k$$

$$\tilde{x}_k = \Phi_k x_{k-1} + w_k - \Phi_k \hat{x}_{k-1} - K_k (H\Phi_k x_{k-1} + Hw_k + v_k - H\Phi_k \hat{x}_{k-1})$$

Como

$$\tilde{x}_k = x_k - \hat{x}_k$$

también podemos decir que

$$\tilde{x}_{k-1} = x_{k-1} - \hat{x}_{k-1}$$

Por ello, combinando términos similares en las ecuaciones de error y de estimación, se tiene

$$\tilde{x}_k = (1 - K_k H) \tilde{x}_{k-1} \Phi_k + (1 - K_k H) w_k - K_k v_k$$

Si definimos la covarianza P_k como

$$P_k = E(\tilde{x}_k^2)$$

y reconociendo que

$$Q_k = E(w_k^2)$$

$$R_k = E(v_k^2)$$

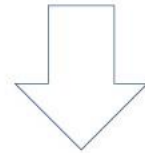
podemos elevar al cuadrado y tomar valor esperado en ambos miembros de la ecuación del error estimado, obteniendo

$$P_k = (1 - K_k H)^2 (P_{k-1} \Phi_k^2 + Q_k) + K_k^2 R_k$$

Para simplificar esta ecuación definimos $M_k = P_{k-1}\Phi_k^2 + Q_k$ la cual es análoga a la primera ecuación de Riccati.

$$P_k = (1 - K_k H)^2 (P_{k-1} \Phi_k^2 + Q_k) + K_k^2 R_k$$

Teniendose la simplificación



$$M_k = P_{k-1} \Phi_k^2 + Q_k$$

$$P_k = (1 - K_k H)^2 M_k + K_k^2 R_k$$

Si queremos encontrar la ganancia que minimiza la varianza del error de estimación, derivaremos esta expresión con respecto a la ganancia e igualaremos a cero, esto es

$$\frac{\partial P_k}{\partial K_k} = 0 = 2(1 - K_k H)M_k(-H) + 2K_k R_k$$

resolviendo esta ecuación se tiene

$$K_k = \frac{M_k H}{H^2 M_k + R_k} = M_k H (H^2 M_k + R_k)^{-1}$$

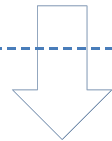
la cual es análoga a la segunda ecuación de Riccati.

Sustituyendo la ganancia óptima en la ecuación de la covarianza se tiene

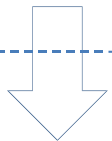
$$P_k = \left(1 - \frac{M_k H^2}{H^2 M_k + R_k}\right) M_k + \left(\frac{M_k H}{H^2 M_k + R_k}\right)^2 R_k$$

Invirtiendo la ecuación de la ganancia óptima y sustituyendola en la ecuación de la varianza

$$K_k = \frac{M_k H}{H^2 M_k + R_k} = M_k H (H^2 M_k + R_k)^{-1}$$



$$K_k R_k = M_k H - H^2 M_k K_k$$



$$P_k = \left(1 - \frac{M_k H^2}{H^2 M_k + R_k}\right) M_k + \left(\frac{M_k H}{H^2 M_k + R_k}\right)^2 R_k$$

se tiene

$$P_k = \frac{R_k K_k}{H} = \frac{M_k H - H^2 M_k K_k}{H} = M_k - H M_k K_k$$

o de forma más simplificada

$$P_k = (1 - K_k H) M_k$$

la cual es análoga a la primera ecuación de

Riccati.

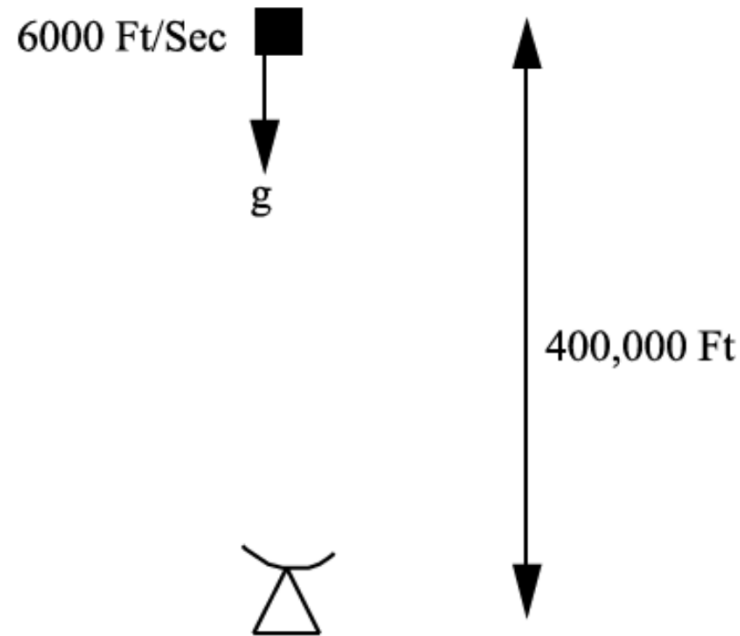
La ganancia del filtro de Kalman se elige de manera que minimice la varianza del error de estimación.

Las ecuaciones de Riccati son simplemente una forma iterativa de encontrar en cada paso la ganancia óptima

La obtención de las ecuaciones de Riccati en forma matricial es mucho más compleja dado que las matrices no siempre obedecen a las mismas reglas como los escalares. Por ejemplo, con matrices es importante el orden en una multiplicación. Además, la inversa se calcula de manera muy diferente respecto a un escalar.

Filtro de Kalman Siguiendo un Objeto en Caída (no hay rozamiento)

Para ilustrar la utilidad de un filtro de Kalman polinomial con ruido de proceso, consideremos el ejemplo unidimensional de un objeto cayendo bastante aprisa sobre un radar de seguimiento como se muestra en la figura



El objeto está inicialmente a 400,000 ft por encima del radar y tiene una velocidad de 6000 ft/s hacia el radar, el cual está localizado sobre la superficie de una tierra plana.

En este ejemplo estamos despreciando el rozamiento o resistencia del aire de manera que solo actúa la gravedad g ($g = 32.2 \text{ ft/s}^2$) sobre el objeto.

Pretendemos que el radar mida la distancia desde el radar hasta el blanco (esto es, la altura del blanco) con una precisión en la medida dada por una desviación típica de 1000 ft.

El radar adquiere medidas 10 veces por segundo durante 30 s.

Nos gustaría construir un filtro para estimar la altura y la velocidad del objeto sin ninguna Información a priori (esto es, sabiendo únicamente la altura y la velocidad iniciales del objeto).

Sabemos de la física básica que si x es la distancia desde el radar hasta el objeto, el valor de x en cualquier instante de tiempo t está dado por

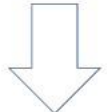
$$x = 400,000 - 6000t - \frac{gt^2}{2}$$

Como comprobación podemos ver que para un tiempo cero la expresión anterior nos da el valor inicial de x de 400,000 ft.

La velocidad del objeto en cualquier instante se puede encontrar tomando la derivada con respecto al tiempo de la expresión anterior,

$$x = 400,000 - 6000t - \frac{gt^2}{2}$$

Derivando
respecto al tiempo



$$\dot{x} = -6000 - gt$$

De nuevo comprobamos que haciendo el tiempo igual a cero esta expresión nos da la velocidad inicial de -6000 ft/s.

Podemos ver como la expresión de la altura x es un polinomio de segundo orden en el tiempo.

Como en este ejemplo el mundo real es realmente un polinomio de segundo orden, se podrá utilizar un filtro de Kalman polinomial de segundo orden sin ruido de proceso.

Compruebase que este modelo polinomial de segundo orden tan simple se obtiene como solución analítica cerrada al integrar la ecuación diferencial

$$\ddot{x} = -g$$

con las condiciones iniciales dadas de altura y velocidad.

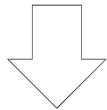
(Filtro de Kalman polinomial de tres estados)

Ecuación en espacio de estados $\dot{\mathbf{x}} = \mathbf{F}\mathbf{x}$ (sin ruido de proceso)

siendo

$$\mathbf{x} = \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix} \quad \text{Vector de estados}$$
$$\mathbf{F} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{Matriz de la dinámica del sistema}$$

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x}$$



$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix} \quad \Rightarrow \quad \ddot{x} = 0 \quad \Rightarrow \quad \ddot{x} = \text{constante}$$

A partir de $\Phi(t) = \mathcal{L}^{-1}[(sI - F)^{-1}]$

o bien de $\Phi(t) = e^{Ft} = I + Ft + \frac{(Ft)^2}{2!} + \dots + \frac{(Ft)^n}{n!} + \dots$

obtendremos la matriz fundamental en la que haciendo $t = T_s$ dará lugar a su forma discretizada o matriz de transición

$$\Phi_k = \begin{bmatrix} 1 & T_s & 0.5T_s^2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$$

La ecuación de medida discretizada es $z_k = Hx_k + v_k$

donde

$$\begin{array}{ccccc}
 \mathbf{H} = [1 & 0 & 0] & \mathbf{x}_k = \begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \end{bmatrix} & \mathbf{v}_k = \begin{bmatrix} v_k \\ 0 \\ 0 \end{bmatrix} & \text{y} & \mathbf{R}_k = E(\mathbf{v}_k \mathbf{v}_k^T) = \\
 \text{Matriz} & & & & & & \text{Matriz (escalar) discretizada} \\
 \text{de medida} & & \text{Vector} & & \text{Vector} & & \text{de la covarianza del ruido de medida} \\
 & & \text{de estados} & & \text{discretizado} & & \\
 & & & & \text{del ruido} & & \\
 & & & & \text{de medida} & &
 \end{array}$$

Las ecuaciones recursivas de Riccati sin ruido de proceso son

$$\begin{aligned}
 \mathbf{M}_k &= \Phi_k \mathbf{P}_{k-1} \Phi_k^T \\
 \mathbf{K}_k &= \mathbf{M}_k \mathbf{H}^T (\mathbf{H} \mathbf{M}_k \mathbf{H}^T + \mathbf{R}_k)^{-1} \\
 \mathbf{P}_k &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \mathbf{M}_k
 \end{aligned}$$

El listado que se da a continuación es una simulación que ilustra como se puede utilizar el el filtro de Kalman polinomial de segundo orden para procesar las medidas radar, y estimar la altura y la velocidad del objeto.

En el listado se puede ver como los estados del filtro se inicializan a cero

$$X_H = X_{DH} = X_{DDH} = 0$$

y como los elementos de la diagonal de la matriz de covarianza inicial son infinito

$$P(1,1) = P(2,2) = P(3,3) = 99999999999999$$

ya que no tenemos ninguna información a priori con respecto a la altura y velocidad iniciales del objeto.

La estimaciones de los estados iniciales presentan unos errores considerables ya que la altura, por ejemplo, son 400,000 ft, la velocidad son – 6000 ft/s , y la aceleración son siempre – 32 ft/s².

Listado (Filtro de Kalman polinomial de tres estados)

```

ORDER =3; ← dimensionado de las matrices
TS=.1; ← Tiempo de muestreo
A0=400000;
A1=-6000.;
A2=-16.1; } Coeficientes del modelo polinomial de mundo real
               $x = 400,000 - 6000t - \frac{gt^2}{2}$ 
XH=0;
XDH=0;
XDDH=0; } Establecimiento del estado inicial
SIGNOISE= 1000 ← Desviación típica del ruido de medida
PHI= ; ←
P=[99999999 0 0;0 99999999 0;0 0 99999999]; ← Matriz de covarianza inicial infinita
IDNP=eye(ORDER);
H= ; ←  $H = [1 \ 0 \ 0]$ 
HT=H';
R= ; ←  $R_k = SIGNOISE^2$ 
PHIT=PHI';
count=0; ← variable para controlar el indexado de los arrays

```

Matriz fundamental discretizada
o matriz de transición

$$\Phi_k = \begin{bmatrix} 1 & T_s & 0,5T_s^2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$$

⋮



for T=0:TS:30

```
PHIP=PHI*P;
PHIPPHIT=PHIP*PHIT;
M=PHIPPHIT+Q;
HM=HMAT*M;
HMHT=HM*HT;|
HMHTR=HMHT+RMAT;
HMHTRINV=inv(HMHTR)
MHT=M*HT;
GAIN=MHT*HMHTRINV;
KH=GAIN*HMAT;
IKH=IDNP-KH;
P=IKH*M;
```

XNOISE= SIGNOISE * randn

X= A0 + A1 * T + A2 * T * T

XD= A1 + 2 * A2 * T

XDD= 2 * A2

XS= X + XNOISE

RES= XS - XH - TS * XDH - 0,5 * TS * TS * XDDH ; $\text{Res}_k = x_k^* - x_{k-1} - T_s \hat{x}_{k-1} - 0.5 T_s^2 \hat{x}_{k-1}$

XH= XH + XDH * TS + 0,5 * TS * TS * XDDH + K(1,1) * RES ;

XDH= XDH + XDDH * TS + K(2,1) * RES ;

XDDH= XDDH + K(3,1) * RES ;

SP11= SQRT(P(1,1))

SP22= SQRT(P(2,2))

SP33= SQRT(P(3,3))

XHERR= X - XH

XDHERR= XD - XDH

XDDHERR= XDD - XDDH

SP11P=- SQRT(P(1,1))

SP22P=- SQRT(P(2,2))

SP33P=- SQRT(P(3,3))

$$\begin{bmatrix} \hat{x}_k \\ \hat{\dot{x}}_k \\ \hat{\ddot{x}}_k \end{bmatrix} = \begin{bmatrix} 1 & T_s & 0,5T_s^2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{k-1} \\ \hat{\dot{x}}_{k-1} \\ \hat{\ddot{x}}_{k-1} \end{bmatrix} + \begin{bmatrix} K_{1k} \\ K_{2k} \\ K_{3k} \end{bmatrix} [x_k^* - [1 \ 0 \ 0] \begin{bmatrix} 1 & T_s & 0,5T_s^2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{k-1} \\ \hat{\dot{x}}_{k-1} \\ \hat{\ddot{x}}_{k-1} \end{bmatrix}]$$

Ecuaciones de Riccati
para obtener
la ganancia Kalman y
la covarianza del error
de estimación

; ← muestra de ruido

Cálculo de los estados de mundo real
con el modelo polinomial de segundo orden

no hace falta integrar
numéricamente
 $\ddot{x} = -g$

Medida ruidosa de la altura con el radar

Ecuaciones de filtrado para obtener los estados estimados

límites teóricos superior es de los errores
a partir de la matriz de covarianza
del error de estimación

Cálculo de los errores
de los estados estimados

límites teóricos inferiores
del error
de estimación

$$\hat{x}_k = \hat{x}_{k-1} + T_s \hat{\dot{x}}_{k-1} + 0.5 T_s^2 \hat{\ddot{x}}_{k-1} + K_{1k} \text{Res}_k$$

$$\hat{\dot{x}}_k = \hat{\dot{x}}_{k-1} + T_s \hat{\ddot{x}}_{k-1} + K_{2k} \text{Res}_k$$

$$\hat{\ddot{x}}_k = \hat{\ddot{x}}_{k-1} + K_{3k} \text{Res}_k$$

count=count+1;
ArrayT(count)=T;
ArrayX(count)=X;
ArrayXH(count)=XH;
ArrayXD(count)=XD;
ArrayXDH(count)=XDH;
ArrayXDD(count)=XDD;
ArrayXDDH(count)=XDDH;
ArrayXHERR(count)=XHERR;
ArraySP11(count)=SP11;
ArraySP11P(count)=SP11P;
ArrayXDHERR(count)=XDHERR;
ArraySP22(count)=SP22;
ArraySP22P(count)=SP22P;
ArrayXDDHERR(count)=XDDHERR;
ArraySP33(count)=SP33;
ArraySP33P(count)=SP33P;

relleno de los arrays para las funciones "plot"

→ end

!

```
figure plot(ArrayT, ArrayX, '—');  
hold on; plot (ArrayT, ArrayXH); hold off; grid on;  
xlabel ('Time (Sec)') ;  
ylabel ('Altitude (Ft)');  
axis([0 30 0 400000]);
```

```
figure plot(ArrayT, ArrayXD, '—');  
hold on; plot (ArrayT, ArrayXDH); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Velocity (Ft/Sec)');  
axis([0 30 -10000 0])
```

```
figure plot(ArrayT, ArrayXDD, '—');  
hold on; plot (ArrayT, ArrayXDDH); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Acceleration (Ft/Sec^2)');  
axis([0 30 -100 100]);
```

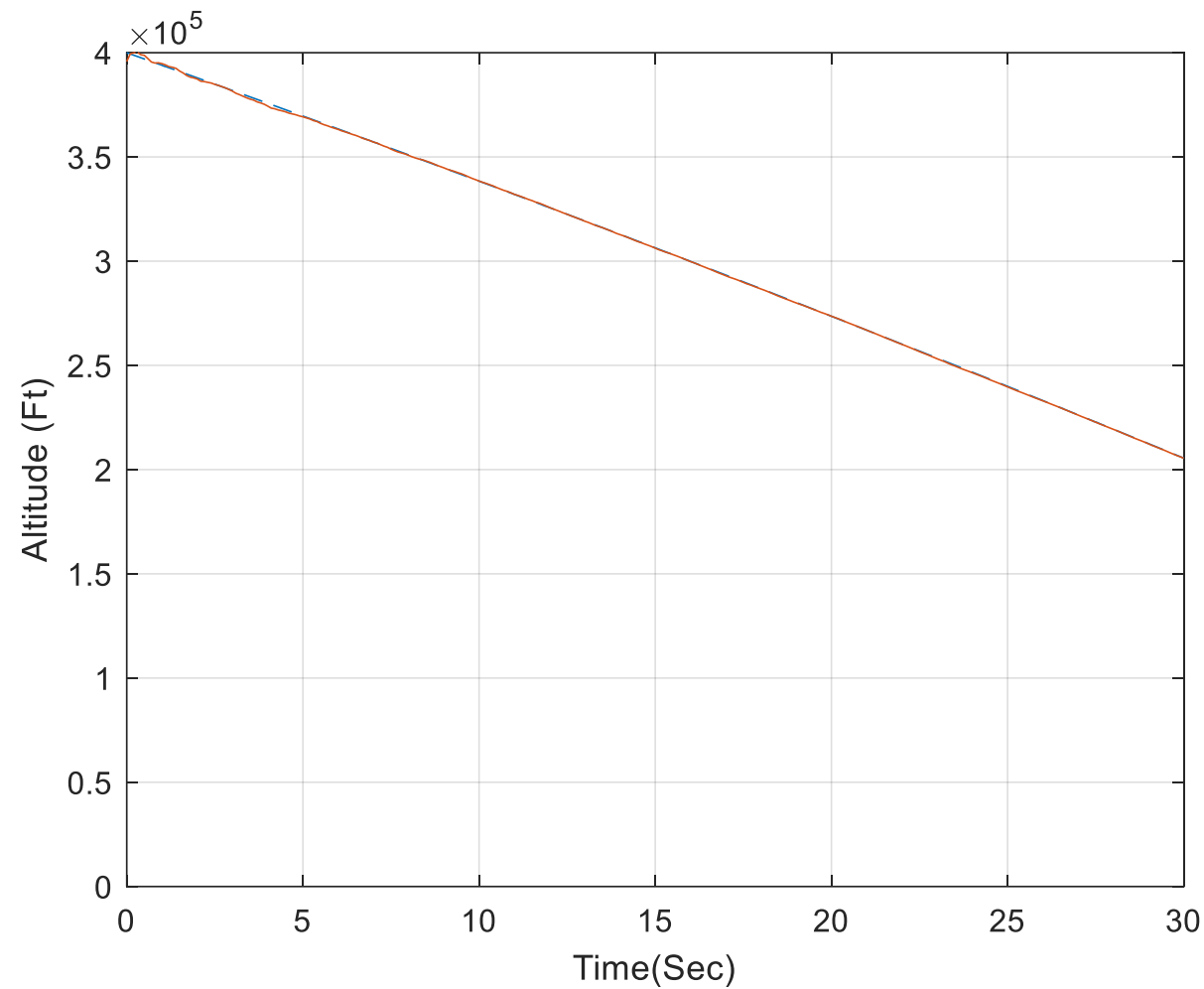
```
figure plot(ArrayT, ArrayXHERR);  
hold on; plot (ArrayT, ArraySP11);  
plot(ArrayT, ArraySP11P); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Error in Estimate of Altitude (Ft)');  
axis([0 30 -1500 1500])
```

```
figure plot(ArrayT, ArrayXDHERR);  
hold on; plot (ArrayT, ArraySP22);  
plot(ArrayT, ArraySP22P); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Error in Estimate of Velocity (Ft/Sec)');  
axis([0 30 -500 500])
```

```
clc
```

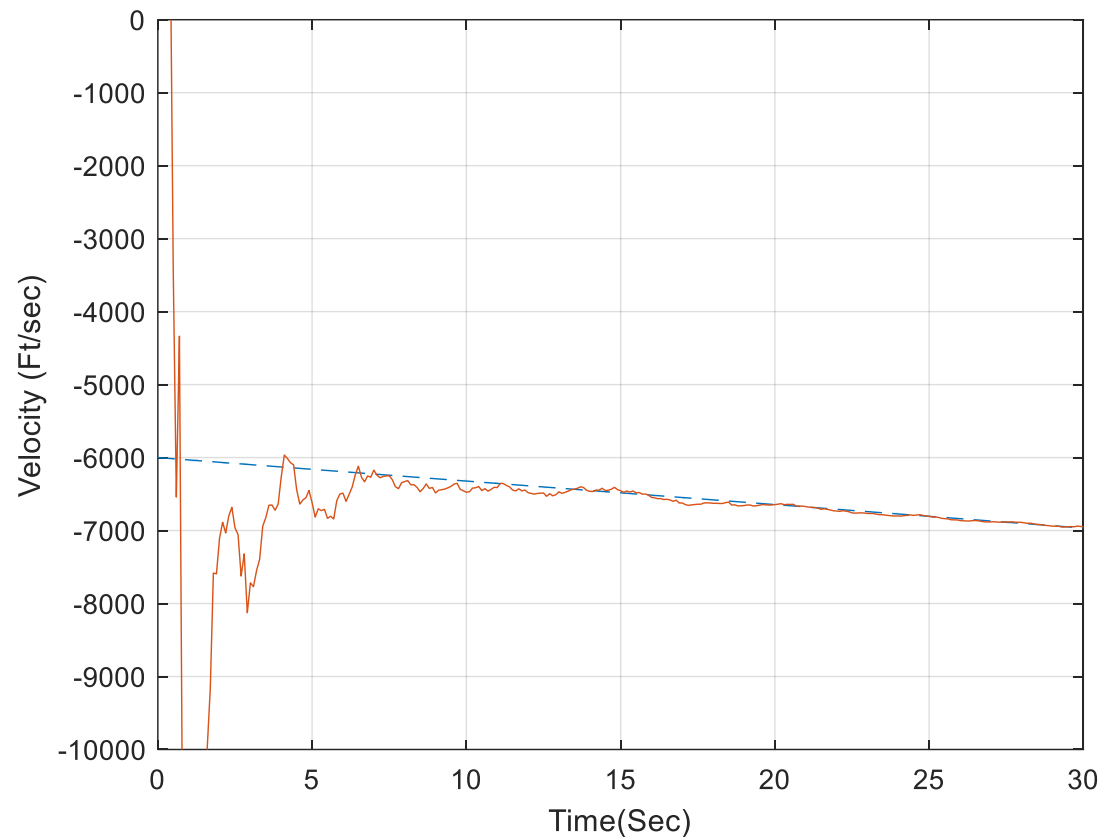


```
figure plot(ArrayT, ArrayX,'—');
hold on; plot (ArrayT, ArrayXH); hold off; grid on;
xlabel ('Time (Sec)') ;
ylabel ('Altitude (Ft)');
axis([0 30 0 400000]);
```



La figura siguiente presenta la altura del objeto junto a la altura estimada por el filtro, obteniendose una concordancia cercana a la perfección.

```
figure plot(ArrayT, ArrayXD,'—');
hold on; plot (ArrayT, ArrayXDH); hold off; grid on;
xlabel ('Time (Sec)');
ylabel ('Velocity (Ft/Sec)');
axis([0 30 -10000 0])
```

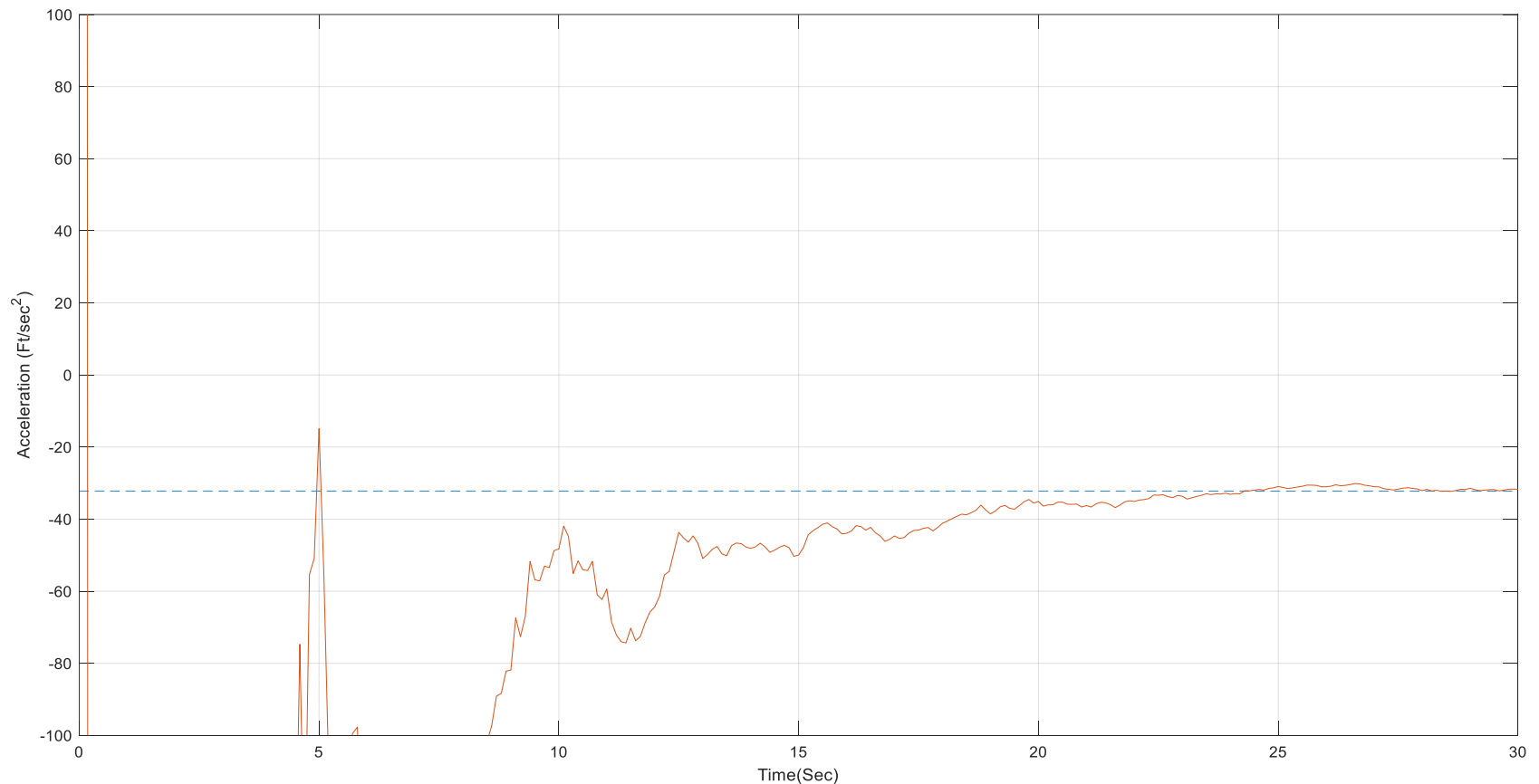


La figura representa la velocidad real del objeto junto a la velocidad estimada por el filtro.

Hasta que no pasan unos 10 s no se obtienen estimaciones altamente precisas de la velocidad.

Las amplias excursiones en la estimación de la velocidad que hay al principio, se deben a que la estimación inicial es de 0 ft/s cuando en realidad la velocidad es de -6000 ft/s. Esa es la principal razón de que se tarde 10s en alcanzar estimaciones precisas.

```
figure plot(ArrayT, ArrayXDD,'—');
hold on; plot (ArrayT, ArrayXDDH); hold off; grid on;
xlabel ('Time (Sec)');
ylabel ('Acceleration (Ft/Sec^2)');
axis([0 30 -100 100]);
```

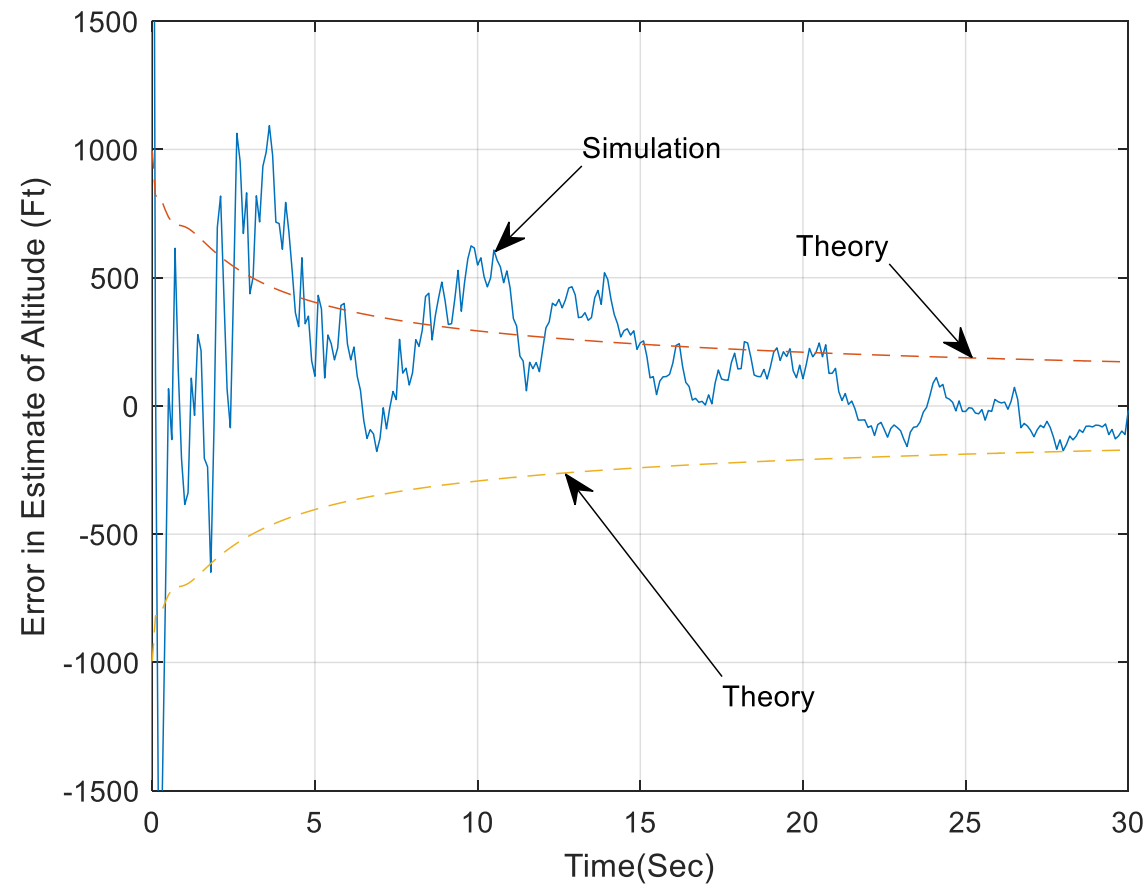


Hemos considerado la aceleración como un estado más del filtro a estimar cuando en realidad no habría hecho falta teniendo en cuenta su valor conocido de -32.2 ft/s^2 (caso ideal sin rozamiento). El filtro ha necesitado al menos 20s para alcanzar una estimación razonable de la misma, obteniéndose fatales estimaciones al principio debido a la discrepancia entre la estimación inicial y su valor real.

```

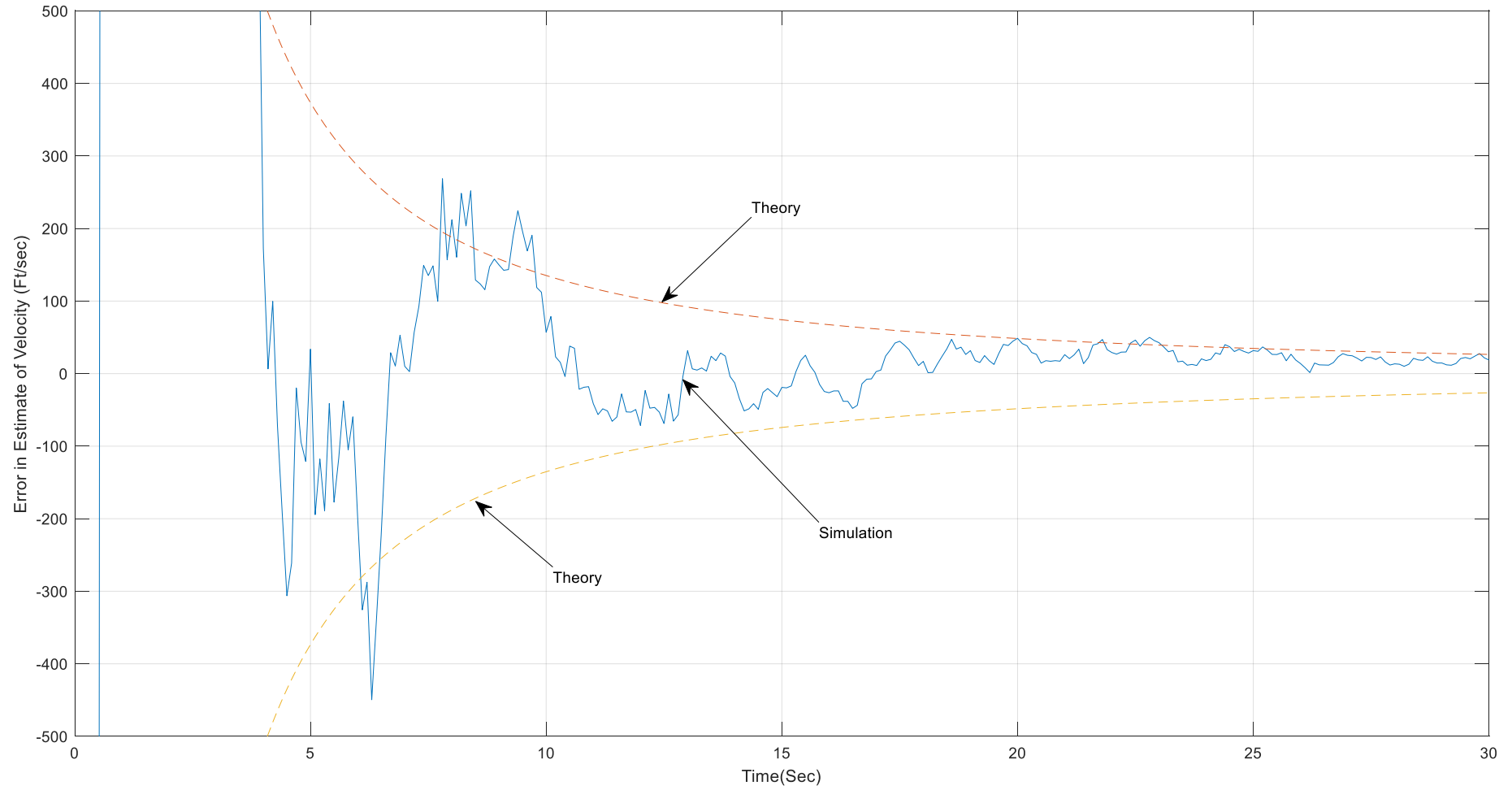
figure plot(ArrayT, ArrayXHERR);
hold on; plot (ArrayT, ArraySP11);
plot(ArrayT,ArraySP11P); hold off; grid on;
xlabel ('Time (Sec)');
ylabel ('Error in Estimate of Altitude (Ft)');
axis([0 30 -1500 1500])

```



El error cometido por el filtro polinomial de segundo orden estimando la altura parece estar dentro de los límites teóricos marcados por la matriz de covarianza del error (no menos del 68 % del tiempo).

```
figure plot(ArrayT, ArrayXDHERR);
hold on; plot (ArrayT, ArraySP22);
plot(ArrayT,ArraySP22P); hold off; grid on;
xlabel ('Time (Sec)');
ylabel ('Error in Estimate of Velocity (Ft/Sec)');
axis([0 30 -500 500])
```



El error cometido por el filtro polinomial de segundo orden estimando la velocidad parece estar dentro de los límites teóricos marcados por la matriz de covarianza del error (no menos del 68 % del tiempo).

Filtro de Kalman Polinomial de Primer Orden Siguiendo un Objeto en Caída

Veamos si también nos conviene un filtro más sencillo (2 estados) para estimar la altura y la velocidad de un objeto en caída libre.

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} \quad \text{Ecuación diferencial en espacio de estados} \quad \Rightarrow \quad \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

$\mathbf{x} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$ Vector de estados

Matriz de la dinámica del sistema

$$\mathbf{F} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$\Phi(t) = \mathcal{L}^{-1}[(s\mathbf{I} - \mathbf{F})^{-1}]$

Matriz fundamental

$$\Phi(t) = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}$$

discretización $t = T_s$

Matriz de transición

$$\Phi_k = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}$$

$$\Phi(t) = e^{\mathbf{F}t} = \mathbf{I} + \mathbf{F}t + \frac{(\mathbf{F}t)^2}{2!} + \dots + \frac{(\mathbf{F}t)^n}{n!} + \dots$$

$$\Phi(t) = e^{\mathbf{F}t} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}t + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}\frac{t^2}{2} = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}$$

$$z_k = Hx_k + v_k$$

Forma discretizada
de la ecuación de medida

Matriz
de medida

$$H = [1 \quad 0]$$

Vector de
estados

$$\hat{x}_k = \begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix}$$

Vector de
ruido de
medida

$$v_k = \begin{bmatrix} v_n \\ 0 \end{bmatrix}$$

Matriz (escalar) de
covarianza del ruido
de medida

$$R_k = E(v_k v_k^T) = \sigma_n^2$$

Listado (Filtro de Kalman polinomial de dos estados)

```

ORDER =2;  ← dimensionado de las matrices
TS=.1;  ← tiempo de muestreo
A0=400000; }
A1=-6000.; } Coeficientes del modelo polinomial
A2=-16.1; } de mundo real
               $x = 400,000 - 6000t - \frac{gt^2}{2}$ 
XH=0; }
XDH=0; } estimación inicial
SIGNOISE=1000.; ← desviación típica del ruido de medida
PHI=[ ]; ←  $\Phi_k = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}$  matriz fundamental discreta
P=[999999999 0;0 999999999 ]; ← matriz inicial de la covarianza del error para un desconocimiento total
IDNP=eye(ORDER);
H=[ ]; ← matriz de medida
HT=H';  $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$ 
R= ; ←  $R_k = SIGNOISE^2$ 
PHIT=PHI';
count=0; ← variable que controla el indexado de los arrays

```

for T=0:TS:30

```

PHIP=PHI*P;
PHIPPHIT=PHIP*PHIT;
M=PHIPPHIT+Q;
HM=HMAT*M;
HMHT=HM*HT;
HMHTR=HMHT+RMAT;
HMHTRINV=inv(HMHTR);
MHT=M*HT;
GAIN=MHT*HMHTRINV;
KH=GAIN*HMAT;
IKH=IDNP-KH;
P=IKH*M;

```

ecuaciones de Riccati para el cálculo de la ganancia Kalman
y la matriz de covarianza del error de estimación

XNOISE= *SIGNOISE* * *randn*

; ← muestra de ruido

X= $A_0 + A_1 * T + A_2 * T^2$

; } cálculo de los estados de mundo real

XD= $A_1 + 2 * A_2 * T$

;

XS= X+XNOISE

;

← medida ruidosa

RES= XS-XH-TS*XDH

;

← residual

XH=XH+XDH*TS+K(1,1)*RES

;

} ecuaciones de filtrado

XDH= $XDH + K(2,1) * RES$;

;

SP11= $SQRT(P(1,1))$;

;

} límites teóricos superiores del error

SP22= $SQRT(P(2,2))$;

;

XHERR= X-XH;

;

} errores de las estimaciones

XDHERR= XD-XDH;

SP11P=- $SQRT(P(1,1))$;

;

SP22P=- $SQRT(P(2,2))$;

;

} límites teóricos inferiores del error

count=count+1;
ArrayT(count)=T;
ArrayX(count)=X;
ArrayXH(count)=XH;
ArrayXD(count)=XD;
ArrayXDH(count)=XDH;
ArrayXHERR(count)=XHERR;
ArraySP11(count)=SP11;
ArraySP11P(count)=SP11P;
ArrayXDHERR(count)=XDHERR;
ArraySP22(count)=SP22;
ArraySP22P(count)=SP22P;

}
relleno de arrays para funciones "plot"

→ end

|

```
figure plot(ArrayT, ArrayX,'- -');  
hold on; plot (ArrayT, ArrayXH); hold off; grid on;  
xlabel ('Time (Sec)') ;  
ylabel ('Altitude (Ft)');  
axis([0 30 0 400000]);
```

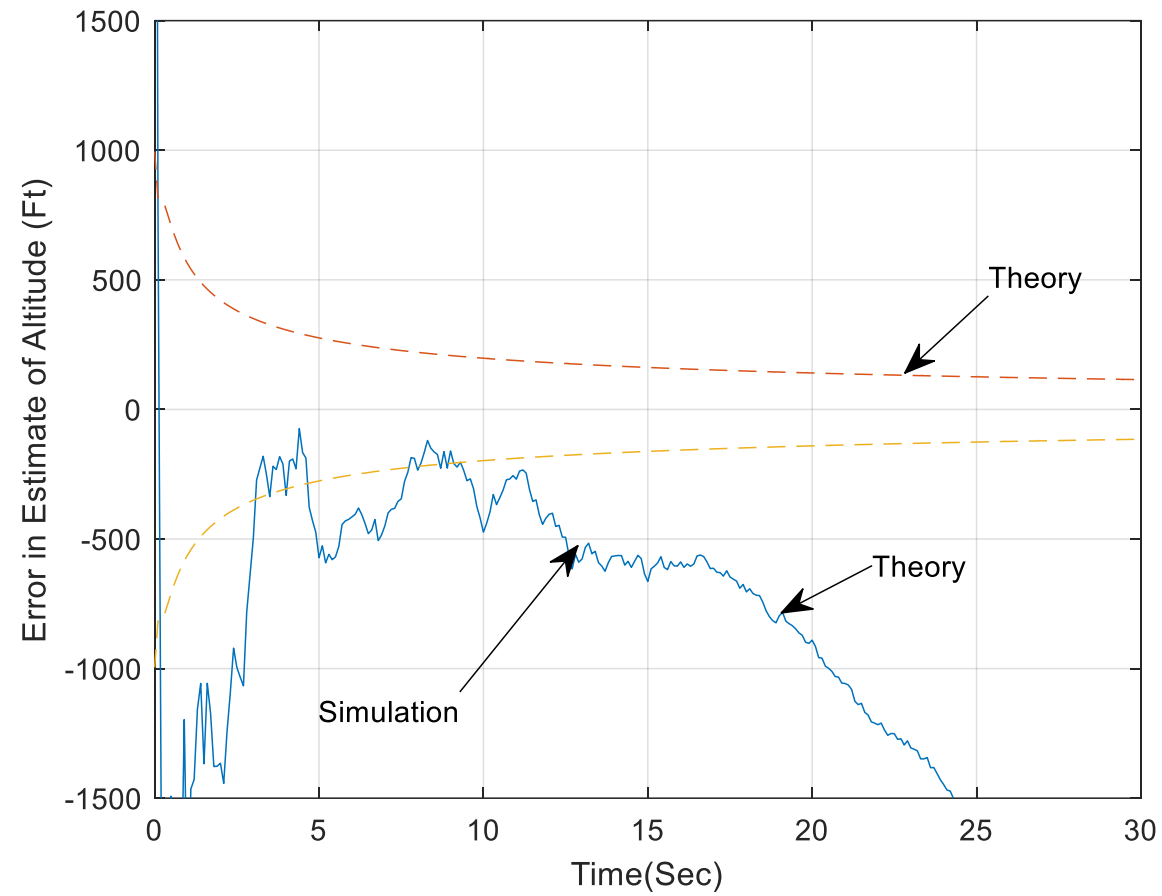
```
figure plot(ArrayT, ArrayXD,'- -');  
hold on; plot (ArrayT, ArrayXDH); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Velocity (Ft/Sec)');  
axis([0 30 -10000 0])
```

```
figure plot(ArrayT, ArrayXHERR);  
hold on; plot (ArrayT, ArraySP11, '- -');  
plot(ArrayT,ArraySP11P,'- -'); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Error in Estimate of Altitude (Ft)');  
axis([0 30 -1500 1500])
```

```
figure plot(ArrayT, ArrayXDHERR);  
hold on; plot (ArrayT, ArraySP22, '- -');  
plot(ArrayT,ArraySP22P, '- -'); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Error in Estimate of Velocity (Ft/Sec)');  
axis([0 30 -500 500])
```

```
clc
```

```
figure plot(ArrayT, ArrayXHERR);
hold on; plot (ArrayT, ArraySP11);
plot(ArrayT,ArraySP11P); hold off; grid on;
xlabel ('Time (Sec)');
ylabel ('Error in Estimate of Altitude (Ft)');
axis([0 30 -1500 1500])
```



La estimación de la posición (altura) diverge de la realidad.

La razón de esta divergencia se debe a que el modelo dinámico está ignorando la presencia de la gravedad, esto es, una aceleración.

$$\dot{x} = Fx \quad \Rightarrow \quad \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \quad \Rightarrow \quad \begin{array}{l} \dot{x} = \dot{x} \\ \ddot{x} = 0 \end{array} \quad \leftarrow \text{No es cierto !!}$$

La imperfección de nuestro modelo se fundamenta en haber ignorado la presencia de una aceleración.

Podremos completar nuestro fallido modelo con “ruido de proceso” sobre la aceleración, dado que en este ejemplo se supone desconocida, pero en cualquier caso no se debe de ignorar.

Filtro de Kalman Polinomial de Primer Orden (2 estados) con Ruido de Proceso en la Aceleración Siguiendo un Objeto en Caída

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{w} \Rightarrow \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ u_s \end{bmatrix} \Rightarrow \begin{matrix} \dot{x} = \dot{x} \\ \ddot{x} = u_s \end{matrix} \leftarrow \begin{matrix} \text{Ruido de proceso} \\ \text{en la aceleración} \end{matrix}$$

Vector de ruido de proceso $\mathbf{w} = \begin{bmatrix} 0 \\ u_s \end{bmatrix}$

Matriz continua de la covarianza del ruido de proceso

$$\mathbf{Q} = E[\mathbf{w}\mathbf{w}^T] = \Phi_s \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Densidad espectral del ruido de proceso $[(\text{m/s}^2)^2 / \text{Hz}]$

Discretización de la matriz de covarianza del ruido de proceso

Matriz continua de la
covarianza del ruido
de proceso

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\Phi^T(\tau) = \begin{bmatrix} 1 & 0 \\ t & 1 \end{bmatrix} \quad \text{Matriz fundamental transpuesta}$$

$$\mathbf{Q}_k = \int_0^{T_s} \Phi(\tau) \mathbf{Q} \Phi^T(\tau) d\tau = \int_0^{T_s} \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t & 1 \end{bmatrix} d\tau = \Phi_s \begin{bmatrix} \frac{T_s^2}{3} & \frac{T_s^2}{2} \\ \frac{T_s^2}{2} & T_s \end{bmatrix}$$

Matriz fundamental

$$\Phi(\tau) = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}$$

Matriz de transición

$$\Phi_k = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}$$

Matriz discreta de la
covarianza del ruido
de proceso

$$\mathbf{Q}_k = \Phi_s \begin{bmatrix} \frac{T_s^2}{3} & \frac{T_s^2}{2} \\ \frac{T_s^2}{2} & T_s \end{bmatrix}$$

Listado (Filtro de Kalman polinomial de dos estados con ruido de proceso)

```

ORDER = 2;  ← dimensionado de las matrices
PHIS= .;    ← densidad espectral del ruido de proceso
TS=.1;      ← tiempo de muestreo
A0= 400000; } coeficientes del
A1= -6000.; } modelo polinomial     $x = 400,000 - 6000t - \frac{gt^2}{2}$ 
A2=-16,1;   } de mundo real
XH=0;       }
XDH=0;      } estimación inicial
SIGNOISE=1000.; ← desviación típica del ruido de medida :  $\sigma_n$ 
PHI=[       ]; ←
P=[999999999 0;0 999999999 ]; ← matriz de covarianza inicial infinita
IDNP=eye(ORDER);
Q=zeros(ORDER);
H=[1 0];    ← matriz de medida
HT=H';
R= SIGNOISE^2; ←  $R_k = E(v_k v_k^T) =$ 
PHIT=PHI';
Q(1,1)=      ; }
Q(1,2)=      ; } matriz discreta de
Q(2,1)=      ; } covarianza del ruido
Q(2,2)=      ; } de proceso
count=0;     ← indexado del relleno
               de los arrays para las funciones "plot"

```

$$\Phi_k = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \quad \text{matriz de transición}$$

$$Q_k = \Phi_s \begin{bmatrix} \frac{T_s^2}{3} & \frac{T_s^2}{2} \\ \frac{T_s^2}{2} & T_s \end{bmatrix}$$

matriz (escalar) de covarianza del ruido de medida



for T=0:TS:30

```
PHIP=PHI*P;  
PHIPPHIT=PHIP*PHIT;  
M=PHIPPHIT+Q;  
HM=HMAT*M;  
HMHT=HM*HT;  
HMHTR=HMHT+RMAT;  
HMHTRINV=inv(HMHTR)  
MHT=M*HT;  
GAIN=MHT*HMHTRINV;  
KH=GAIN*HMAT;  
IKH=IDNP-KH;  
P=IKH*M;
```

ecuaciones de Riccati
para la obtención de la ganancia Kalman
y de la matriz de covarianza del error

XNOISE= *SIGNOISE* * randn; ← muestra de ruido

X= $A_0 + A_1 * T + A_2 * T^2$; ← estados de mundo real

XD= $A_1 + 2 * A_2 * T$;

XS= X+XNOISE; ← medida ruidosa de la altura = valor de altura de mundo real + muestra ruidosa

RES= XS-XH-TS*XDH; ← cálculo del residual de medida

XH= XH+XDH*TS+K(1,1)*RES; ← ecuaciones de filtrado

XDH= XDH + K(2,1) * RES;

SP11= SQRT(P(1,1));

SP22= SQRT(P(2,2));

XHERR= X-XH;

XDHERR= XD-XDH; ← errores de estimación y límites teóricos


SP11P=-SQRT(P(1,1));

SP22P=-SQRT(P(2,2));

⋮

```
count=count+1;  
ArrayT(count)=T;  
ArrayX(count)=X;  
ArrayXH(count)=XH;  
ArrayXD(count)=XD;  
ArrayXDH(count)=XDH;  
ArrayXHERR(count)=XHERR;  
ArraySP11(count)=SP11;  
ArraySP11P(count)=SP11P;  
ArrayXDHERR(count)=XDHERR;  
ArraySP22(count)=SP22;  
ArraySP22P(count)=SP22P;
```

} relleno de los arrays para las funciones "plot"

 end

⋮

⋮

```
figure plot(ArrayT, ArrayX, '—');  
hold on; plot (ArrayT, ArrayXH); hold off; grid on;  
xlabel ('Time (Sec)') ;  
ylabel ('Altitude (Ft)');  
axis([0 30 0 400000]);
```

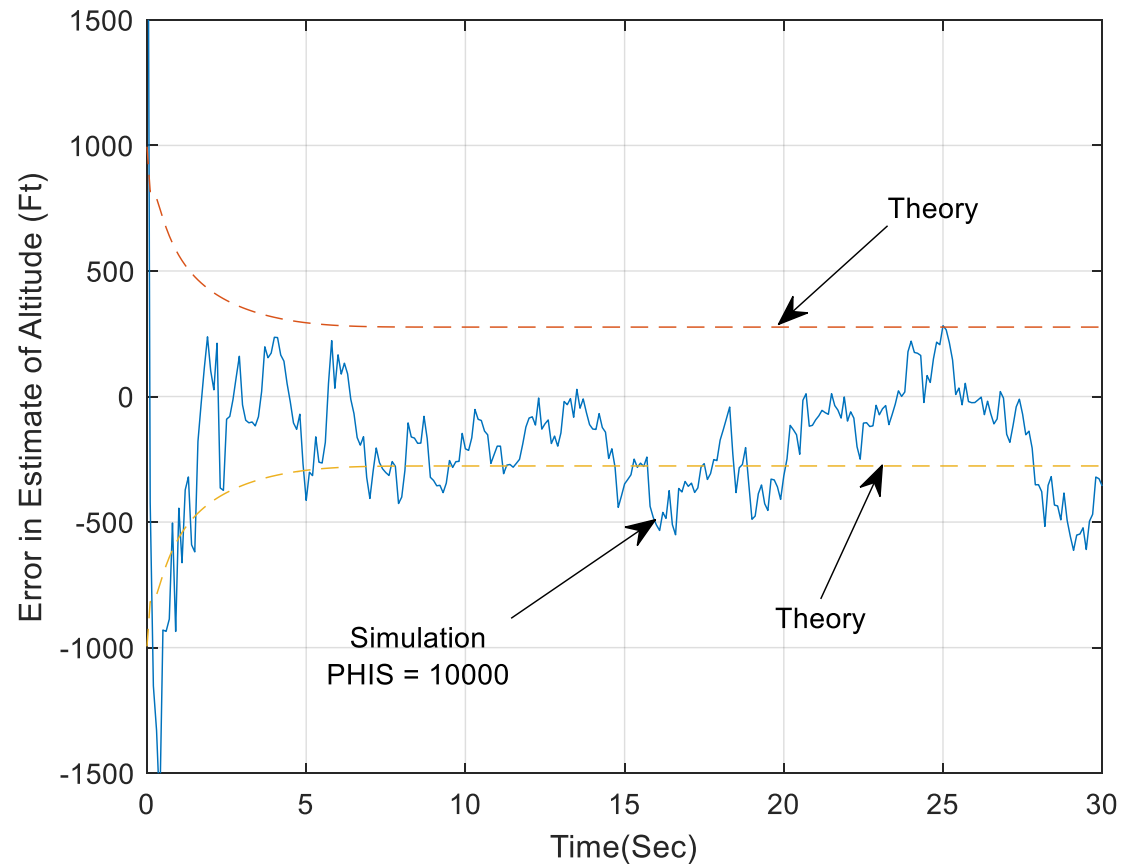
```
figure plot(ArrayT, ArrayXD, '—');  
hold on; plot (ArrayT, ArrayXDH); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Velocity (Ft/Sec)');  
axis([0 30 -10000 0])
```

```
figure plot(ArrayT, ArrayXHERR);  
hold on; plot (ArrayT, ArraySP11);  
plot(ArrayT, ArraySP11P); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Error in Estimate of Altitude (Ft)');  
axis([0 30 -1500 1500])
```

```
figure plot(ArrayT, ArrayXDHERR);  
hold on; plot (ArrayT, ArraySP22);  
plot(ArrayT, ArraySP22P); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Error in Estimate of Velocity (Ft/Sec)');  
axis([0 30 -500 500])
```

```
clc
```

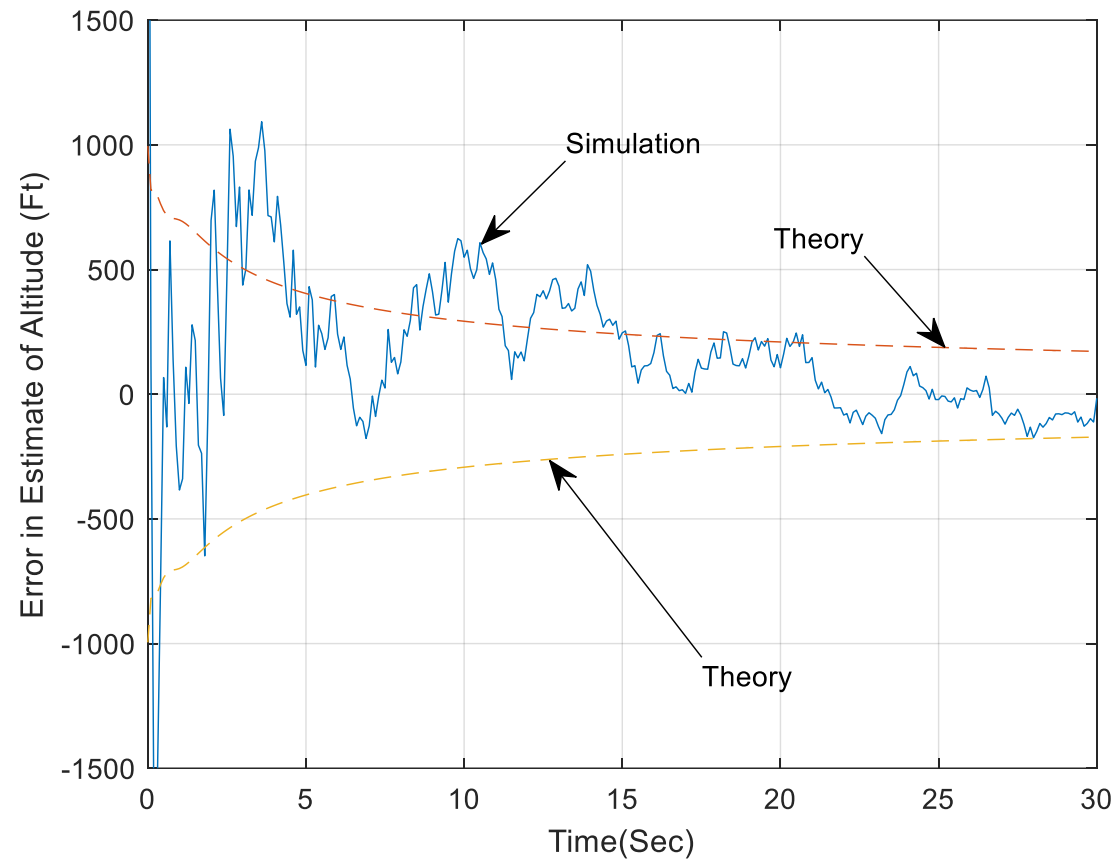
```
figure plot(ArrayT, ArrayXHERR);
hold on; plot (ArrayT, ArraySP11);
plot(ArrayT,ArraySP11P); hold off; grid on;
xlabel ('Time (Sec)');
ylabel ('Error in Estimate of Altitude (Ft)');
axis([0 30 -1500 1500])
```



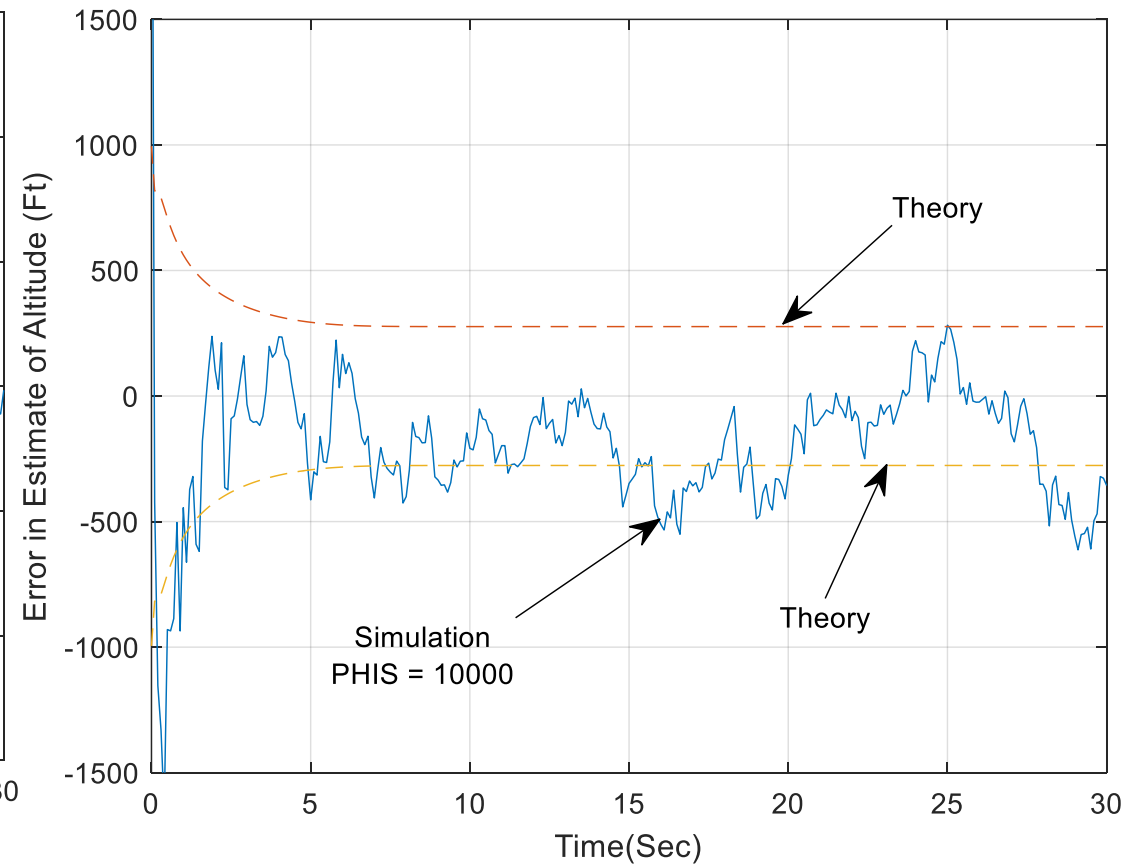
Se ha evitado la divergencia añadiendo ruido de proceso con densidad espectral de valor $\Phi_s = 10000 \text{ [(m/s}^2\text{)}^2 / \text{Hz}]$.

Aunque la inyección de ruido de proceso haya evitado la divergencia hay que apuntar que ello es a costa de incrementar el error de estimación.

Comparamos el error en la estimación de la altura del filtro polinomial de segundo orden con el del filtro polinomial de primer orden con ruido de proceso.

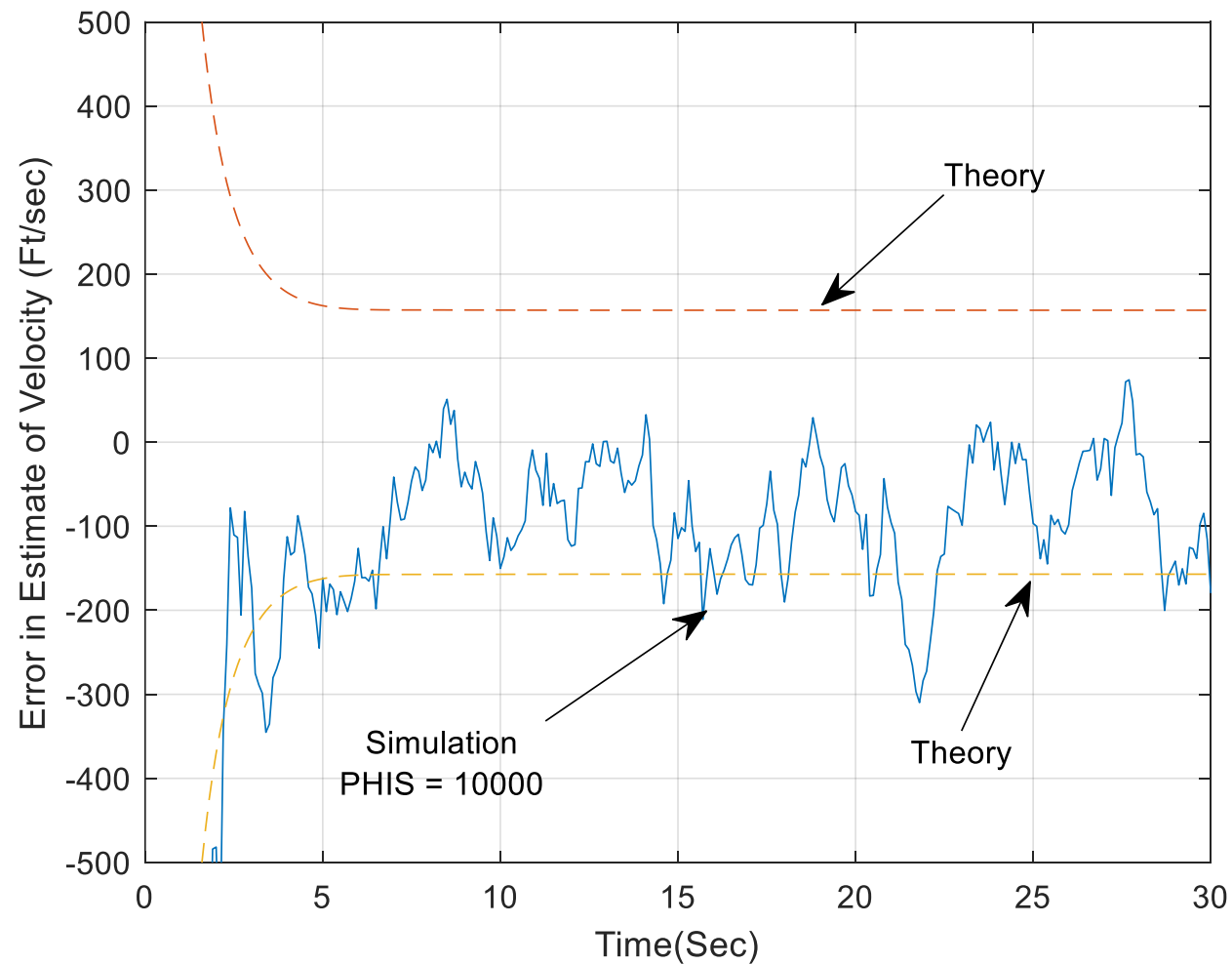


Error de $\pm 200\text{ ft}$



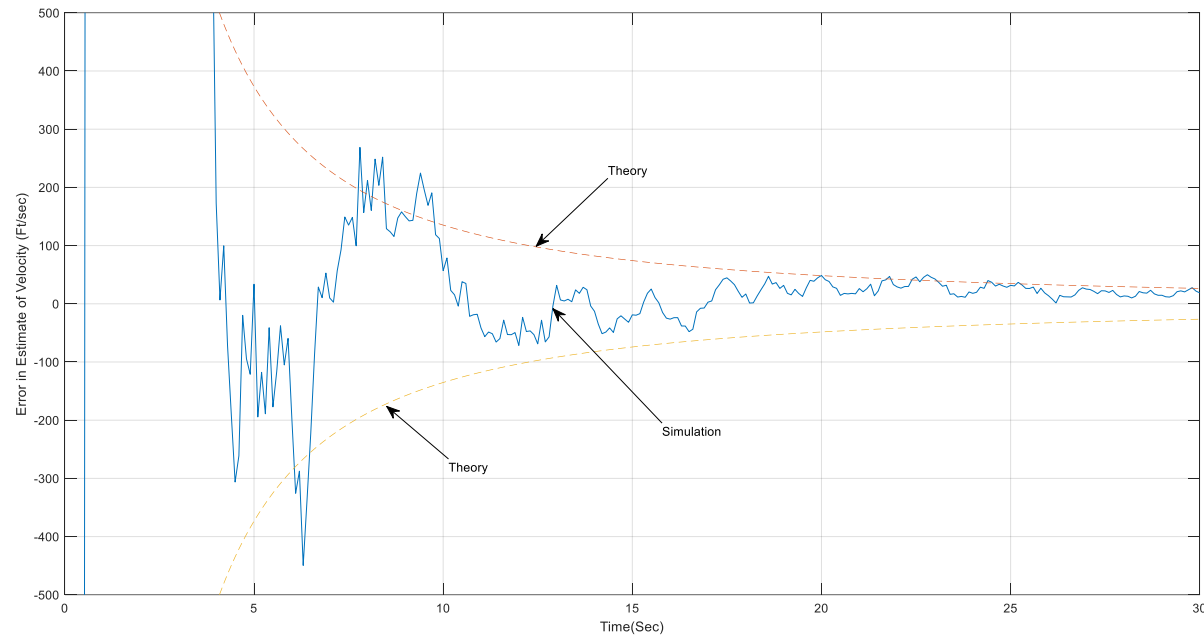
Error de $\pm 300\text{ ft}$

```
figure plot(ArrayT, ArrayXDHERR);
hold on; plot (ArrayT, ArraySP22);
plot(ArrayT,ArraySP22P); hold off; grid on;
xlabel ('Time (Sec)');
ylabel ('Error in Estimate of Velocity (Ft/Sec)');
axis([0 30 -500 500])
```

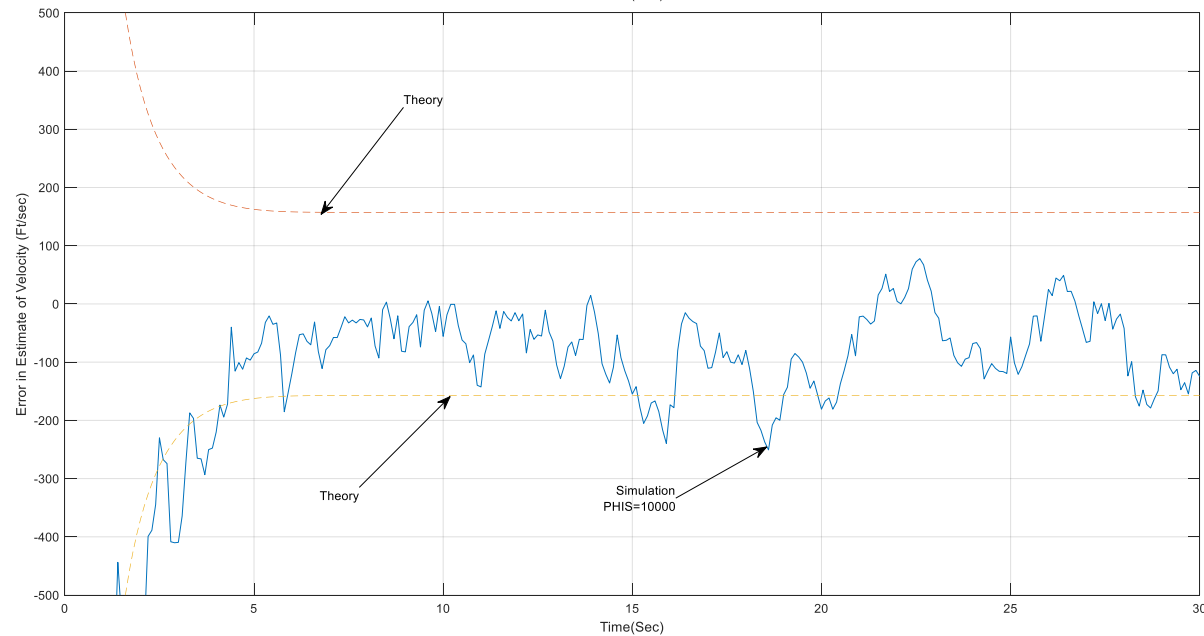


El error en la estimación de la velocidad está en torno a ± 150 ft/s cuando era del orden de ± 25 ft en el filtro polinomial de segundo orden.

Comparamos el error en la estimación de la velocidad del filtro polinomial de segundo orden con el del filtro polinomial de primer orden con ruido de proceso.



Error de $\pm 25 \text{ ft/s}$



Error de $\pm 150 \text{ ft/s}$

Filtro de Kalman Polinomial de Primer Orden con Información Determinista a Priori Siguiendo un Objeto en Caída

Haciendo uso de información a priori podemos reducir los errores de estimación con un filtro de primer orden y sin tener que utilizar el ruido de proceso

Como sabemos que sobre el cuerpo está actuando únicamente la gravedad, esta se podrá incorporar a la ecuación de espacio de estados como una información determinista.

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}u + \mathbf{w}$$

La ecuación de filtrado resultante es

$$\hat{\mathbf{x}}_k = \Phi_k \hat{\mathbf{x}}_{k-1} + \mathbf{G}_k u_{k-1} + \mathbf{K}_k (z_k - \mathbf{H} \Phi_k \hat{\mathbf{x}}_{k-1} - \mathbf{H} \mathbf{G}_k u_{k-1})$$

Si se asume que u_{k-1} es constante entre instantes de muestreo, \mathbf{G}_k se obtiene a partir de

$$\mathbf{G}_k = \int_0^{T_s} \Phi(\tau) \mathbf{G} \, d\tau$$

En nuestro problema particular la única ecuación diferencial que describe el mundo real es

$$\ddot{x} = -g$$

que en fomato de espacio de estados es

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}u$$

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} g$$

esto es

Matriz de la dinámica del sistema	$\mathbf{F} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$	\Rightarrow	$\Phi_k = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}$	Matriz fundamental discreta o matriz de transición
--------------------------------------	-------------------------------------------------------------	---------------	-----------------------------------------------------------	-------------------------------------------------------

de la ecuación en espacio de estados se tiene

$$\mathbf{G} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

y

$$u_{k-1} = g$$

La matriz discreta \mathbf{G}_k será

$$\mathbf{G}_k = \int_0^{T_s} \Phi(\tau) \mathbf{G} d\tau = \int_0^{T_s} \begin{bmatrix} 1 & \tau \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ -1 \end{bmatrix} d\tau = \begin{bmatrix} -\frac{T_s^2}{2} \\ -T_s \end{bmatrix}$$

Al ser la ecuación de filtrado

$$\hat{\mathbf{x}}_k = \Phi_k \hat{\mathbf{x}}_{k-1} + \mathbf{G}_k u_{k-1} + \mathbf{K}_k (z_k - \mathbf{H} \Phi_k \hat{\mathbf{x}}_{k-1} - \mathbf{H} \mathbf{G}_k u_{k-1})$$

tendremos

$$\begin{bmatrix} \hat{x}_k \\ \hat{\dot{x}}_k \end{bmatrix} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{k-1} \\ \hat{\dot{x}}_{k-1} \end{bmatrix} + \begin{bmatrix} -\frac{T_s^2}{2} \\ -T_s \end{bmatrix} g \\ + \begin{bmatrix} K_{1_k} \\ K_{2_k} \end{bmatrix} \left[x_k^* - \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{k-1} \\ \hat{\dot{x}}_{k-1} \end{bmatrix} - \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} -\frac{T_s^2}{2} \\ -T_s \end{bmatrix} g \right]$$

De esta ecuación matricial se tienen las siguientes ecuaciones escalares

Ecuaciones escalares

$$\hat{x}_k = \hat{x}_{k-1} + \hat{\dot{x}}_{k-1} T_s - 0.5gT_s^2 + K_{1k}(x_k^* - \hat{x}_{k-1} - \hat{\dot{x}}_{k-1}T_s + 0.5gT_s^2)$$

$$\hat{\dot{x}}_k = \hat{\dot{x}}_{k-1} - gT_s + K_{2k}(x_k^* - \hat{x}_{k-1} - \hat{\dot{x}}_{k-1}T_s + 0.5gT_s^2)$$

Si definimos el residual como

$$Res_k = x_k^* - \hat{x}_{k-1} - \hat{\dot{x}}_{k-1}T_s + 0.5gT_s^2$$

las ecuaciones de filtrado Kalman toman una forma más simplificada

$$\hat{x}_k = \hat{x}_{k-1} + \hat{\dot{x}}_{k-1} T_s - 0.5gT_s^2 + K_{1k}Res_k$$

$$\hat{\dot{x}}_k = \hat{\dot{x}}_{k-1} - gT_s + K_{2k}Res_k$$

La información a priori sobre la gravedad afecta únicamente a la estructura del filtro de Kalman.

La gravedad no influye sobre las ecuaciones de Riccati.

Listado

(Filtro de Kalman Polinomial de Primer Orden o de Dos estados con Información Determinista a Priori)

```

TS=.1; ← Tiempo de muestreo
PHIS= .; ← Sin ruido de proceso
A0=400000 .;
A1=- 6000.;
A2=- 16.1; } Coeficientes del modelo polinomial
              de mundo real
XH=0.;
XDH=0.; } Establecimiento de un estado inicial
SIGNOISE=1000.; ← Desviación típica del ruido de medida
ORDER= ; ← dimensionado de las matrices
T=0.; ← inicialización de la variable tiempo
PHI=[      ]; ←  $\Phi_k = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}$  Matriz de transición
P=[99999999 0;0 999999999]; ← Matriz de covarianza inicial infinita
IDNP=eye(ORDER);
Q=zeros(ORDER);
RMAT= SIGNOISE2; ← Matriz discreta (escalar) de la covarianza del ruido de medida  $R_k = E(v_k v_k^T) = SIGNOISE^2$ 
              } Matriz discreta de la
              covarianza del ruido
              de proceso  $Q_k = \Phi_s \begin{bmatrix} \frac{T_s^2}{3} & \frac{T_s^2}{2} \\ \frac{T_s^2}{2} & T_s \end{bmatrix}$ 
HMAT=[1 0]; ← matriz de medida
HT=HMAT';
PHIT=PHI';
count=0; ← Inicialización del
           indexado de los arrays
           que almacenan los
           resultados de la simulación
           ...
    
```

for T=0:TS:30

```

PHIP=PHI*P;
PHIPPHIT=PHIP*PHIT;
M=PHIPPHIT+Q;
HM=HMAT*M;
HMHT=HM*HT;
HMHTR=HMHT+RMAT;
HMHTRINV=inv(HMHTR)
MHT=M*HT;
GAIN=MHT*HMHTRINV;
KH=GAIN*HMAT;
IKH=IDNP-KH;
P=IKH*M;
XNOISE= SIGNOISE * randn;
X= A0 + A1 * T + A2 * T^2;
XD= A1 + 2 * A2 * T;
XS= X+XNOISE;
RES= XS-XH-TS*XDH+16.1*TS*TS;
XH=XH+XDH*TS-16.1*TS*TS+K(1,1)*RES;
XDH=XDH - 32.2 * TS + K(2,1) * RES;
SP11= SQRT(P(1,1));
SP22=SQRT(P(2,2));
XHERR= X-XH;
XDHERR= XD-XDH;
SP11P=- SQRT(P(1,1));
SP22P=-SQRT(P(2,2));

```

Ecuaciones de Riccati para el cálculo
de la ganancia Kalman y
la matriz de covarianza del error de estimación

muestra de ruido

Modelo polinomial de segundo orden (mundo real)

Medida ruidosa radar de la altura

Ecuaciones de filtrado Kalman (primer orden)
con información determinista a priori sobre la gravedad

límite teórico superior del error
a partir de la matriz de covarianza

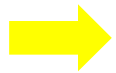
Error de la estimación

límite teórico inferior del error
a partir de la matriz de covarianza

$$\hat{x}_k = \hat{x}_{k-1} + \hat{x}_{k-1} T_s - 0.5 g T_s^2 + K_{1k} Res_k$$

$$\hat{\dot{x}}_k = \hat{\dot{x}}_{k-1} - g T_s + K_{2k} Res_k$$

count=count+1;
ArrayT(count)=T;
ArrayX(count)=X;
ArrayXH(count)=XH;
ArrayXD(count)=XD;
ArrayXDH(count)=XDH;
ArrayXHERR(count)=XHERR;
ArraySP11(count)=SP11;
ArraySP11P(count)=SP11P;
ArrayXDHERR(count)=XDHERR;
ArraySP22(count)=SP22;
ArraySP22P(count)=SP22P;



end

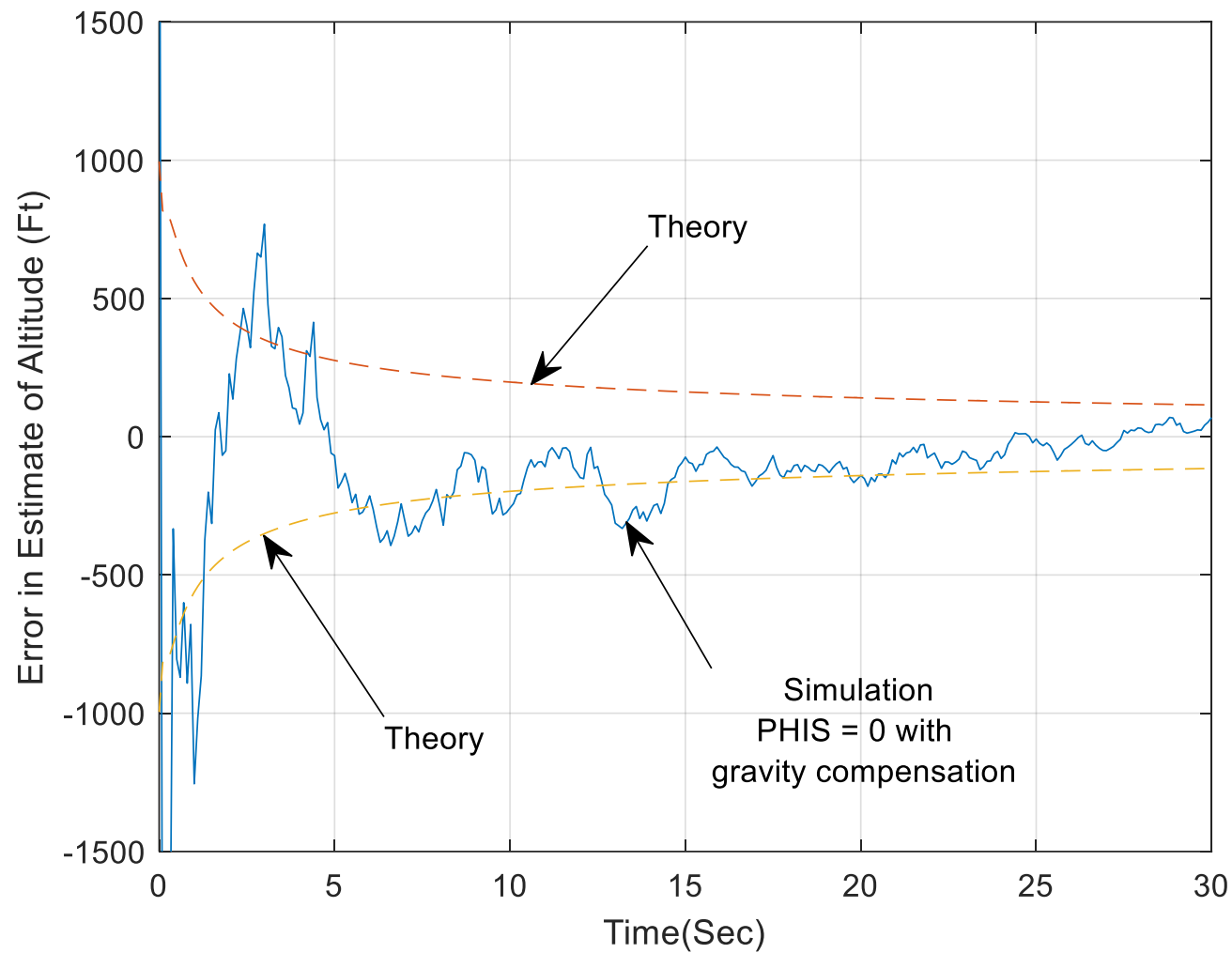
relleno de los arrays para las funciones "plot"

!

```
figure plot(ArrayT, ArrayXHERR);  
hold on; plot (ArrayT, ArraySP11);  
plot(ArrayT,ArraySP11P); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Error in Estimate of Altitude (Ft)');  
axis([0 30 -1500 1500])
```

```
figure plot(ArrayT, ArrayXDHERR);  
hold on; plot (ArrayT, ArraySP22);  
plot(ArrayT,ArraySP22P); hold off; grid on;  
xlabel ('Time (Sec)');  
ylabel ('Error in Estimate of Velocity (Ft/Sec)');  
axis([0 30 -500 500])
```

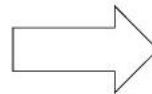
```
clc
```



Añadiendo información a priori sobre la gravedad al filtro polinomial de primer orden sin ruido de proceso, se reduce el error en altura respecto al filtro polinomial de segundo orden.

Filtro polinomial de segundo orden

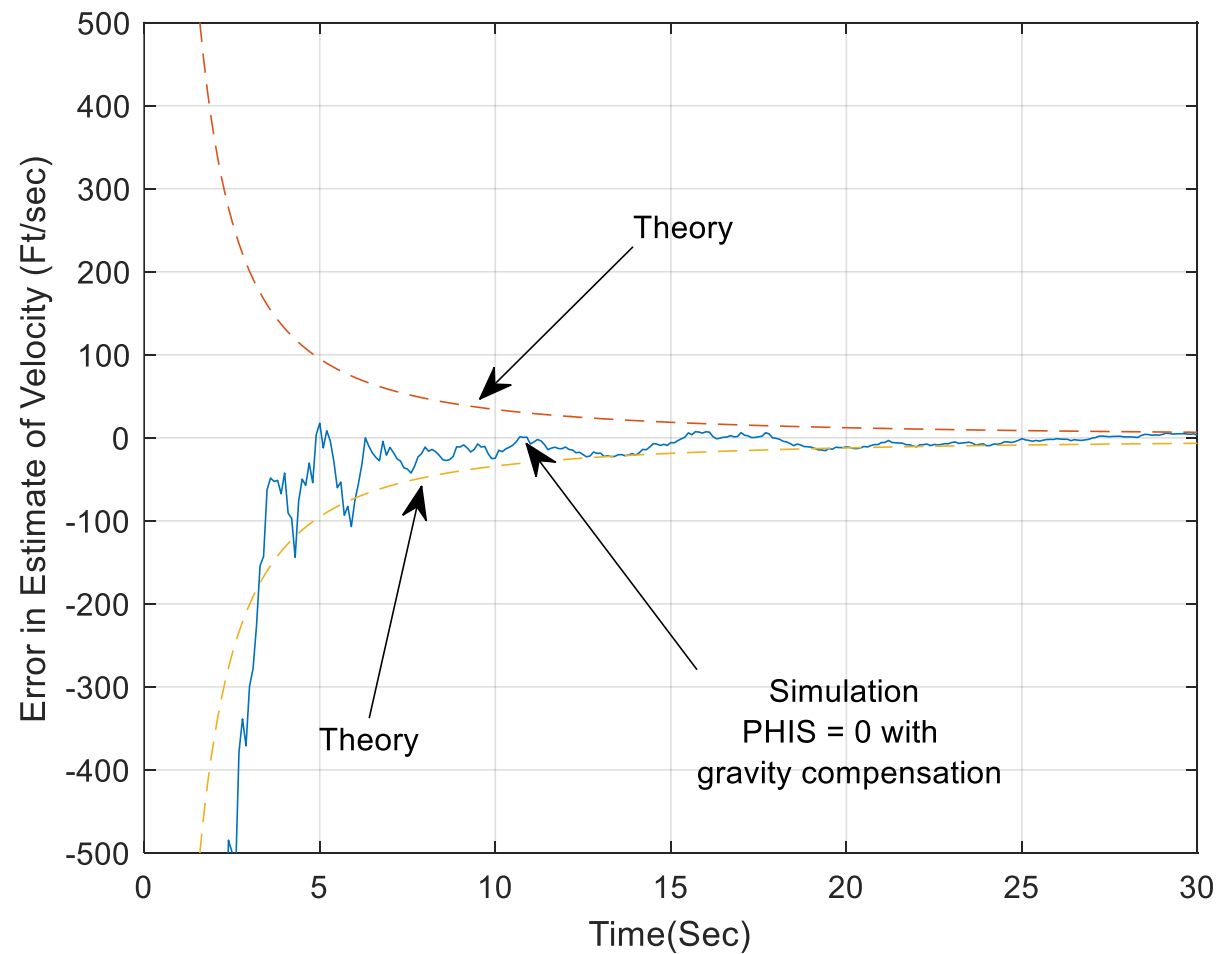
Error en altura ± 200 ft



Filtro polinomial de primer orden

con información a priori sobre la gravedad

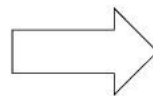
Error en altura ± 100 ft



Añadiendo información a priori sobre la gravedad al filtro polinomial de primer orden sin ruido de proceso, se reduce el error en velocidad respecto al filtro polinomial de segundo orden.

Filtro polinomial de segundo orden

Error en velocidad ± 25 ft/s



Filtro polinomial de primer orden
con información a priori sobre la gravedad

Error en velocidad ± 5 ft/s