

**Universidad de Granada**

Departamento de Estadística e Investigación Operativa



**Estimación de señales en  
sistemas estocásticos no lineales**

**Silvia M<sup>a</sup> Valenzuela Ruiz**

Doctorado en Estadística e Investigación Operativa

Granada 2010

Editor: Editorial de la Universidad de Granada  
Autor: Silvia M<sup>a</sup> Valenzuela Ruiz  
D.L.: GR 1380-2011  
ISBN: 978-84-694-0929-9





TESIS DOCTORAL

ESTIMACIÓN DE SEÑALES EN SISTEMAS ESTOCÁSTICOS NO  
LINEALES

Memoria realizada para optar al grado de doctor por Silvia M<sup>a</sup> Valenzuela Ruiz bajo la dirección de Dra. D<sup>a</sup> Aurora Hermoso Carazo y Dra. D<sup>a</sup> Josefa Linares Pérez, profesoras del Departamento de Estadística e Investigación Operativa de la Universidad de Granada.

Diciembre de 2010

**Doctorando**

Silvia M<sup>a</sup> Valenzuela Ruiz

**V<sup>o</sup> B<sup>o</sup> de las directoras de Tesis**

Dra. D<sup>a</sup> Aurora Hermoso Carazo

Dra. D<sup>a</sup> Josefa Linares Pérez



# Índice general

<b>Introducción general</b>	<b>III</b>
<b>1. Estimación en sistemas no lineales</b>	<b>1</b>
1.1. Introducción . . . . .	1
1.2. Filtro de Kalman extendido . . . . .	3
1.2.1. Sistemas no lineales con ruidos aditivos . . . . .	3
1.2.2. Sistemas no lineales con ruidos no aditivos . . . . .	7
1.3. Filtro de Kalman extendido iterado . . . . .	9
1.4. Filtro de Kalman extendido de segundo orden . . . . .	12
1.5. Filtro de Kalman ensemble . . . . .	15
1.6. Ejemplo numérico . . . . .	18
Apéndice: Filtro de Kalman . . . . .	25
<b>2. Estimación cuadrática en sistemas no lineales con certidumbre</b>	<b>27</b>
2.1. Introducción . . . . .	27
2.2. Estimación cuadrática y proyecciones ortogonales . . . . .	29
2.3. Descripción del sistema . . . . .	32
2.4. Filtro extendido cuadrático . . . . .	34
2.4.1. Linealización del sistema . . . . .	34

2.4.2. Sistema aumentado. Propiedades . . . . .	36
2.4.3. Algoritmo de filtrado en el sistema aumentado . . . . .	42
2.5. Filtro extendido cuadrático iterado . . . . .	43
2.6. Filtro extendido cuadrático de segundo orden . . . . .	45
2.7. Ejemplo numérico . . . . .	47
<b>3. Estimación cuadrática en sistemas no lineales con observaciones inciertas</b>	<b>53</b>
3.1. Introducción . . . . .	53
3.2. Descripción del sistema . . . . .	56
3.3. Sistema aumentado . . . . .	59
3.3.1. Linealización del sistema . . . . .	60
3.3.2. Propiedades del sistema aumentado . . . . .	61
3.4. Filtro extendido cuadrático . . . . .	67
3.4.1. Filtro lineal en sistemas con ruido de observación no blanco . . . . .	68
3.4.2. Algoritmo de filtrado lineal en el sistema aumentado . . . . .	76
3.5. Filtro ensemble . . . . .	77
3.6. Filtro ensemble cuadrático . . . . .	82
3.7. Ejemplo numérico . . . . .	86
3.7.1. Actuación del filtro extendido cuadrático . . . . .	88
3.7.2. Actuación de los filtros ensemble . . . . .	91
<b>Conclusiones y Futuras Líneas de Investigación</b>	<b>99</b>
<b>Bibliografía</b>	<b>105</b>
<b>Códigos fuente de MATLAB</b>	<b>113</b>

# Introducción general

La Teoría de Sistemas Dinámicos tiene como objetivo el estudio de los cambios que experimentan los sistemas físicos en el transcurso del tiempo, debido a la influencia de factores externos e internos. Si dichos cambios son de naturaleza aleatoria, es decir, no se rigen por leyes exactas, estos sistemas se denominan Sistemas Estocásticos.

Para el análisis de un sistema dinámico estocástico es necesario conocer el modelo matemático que rige su evolución, así como las condiciones iniciales que posee al comienzo del estudio. Para la modelización, el estado del sistema en cada instante de tiempo se describe mediante una o más variables aleatorias que representan las características de interés de dicho sistema, que constituyen el denominado *vector estado* o *señal*.

Una importante parcela en el estudio y análisis de un sistema dinámico estocástico, en la que centramos nuestro objetivo en este trabajo, es la predicción de los valores de la señal a lo largo del tiempo a partir de la observación del sistema. El problema que generalmente se presenta es que la señal no será directamente observable, sino que sólo se dispondrá de observaciones de alguna manera relacionadas con él que, además, estarán usualmente perturbadas por ruido. Así, una vez determinado el modelo que rige la evolución del sistema a lo largo del tiempo, y el modelo que describe las observaciones ruidosas que han de usarse, pueden utilizarse diversas técnicas de estimación, esto es, diversos criterios de optimalidad, para obtener aproximaciones de la señal basadas en dichas observaciones.



Aunque el problema de estimación es un tópico bien antiguo, ya que el hombre siempre ha estado interesado en predecir los fenómenos de la naturaleza, el inicio de la teoría de estimación puede situarse a principios del siglo XVII, con los intentos de Galileo Galilei de minimizar varias funciones de error para estimar la posición de los satélites del planeta Jupiter; sin embargo, es Newton quien, por los enunciados de las leyes de la mecánica y de la gravitación universal (1687), se considera el precursor formal de la estimación en modelos dinámicos.

En lo que atañe al enfoque de nuestro trabajo, cabe destacar la importancia que supuso en la teoría de estimación el *método de mínimos cuadrados*, desarrollado por Gauss en 1795, mediante el cual se pudo determinar con precisión la órbita del asteroide Ceres después de su ocultación tras el sol. Posteriormente, en 1821, Gauss desarrolló también una formulación recursiva de este método de estimación, la cual permite actualizar sucesivamente un estimador con nuevas observaciones, sin necesidad de repetir los cálculos desde su inicio. Este gran avance convirtió la estimación por mínimos cuadrados en la base para la resolución de un gran número de problemas de estimación.

Fue en la primera mitad del siglo XX cuando se realizaron las primeras aplicaciones del método de mínimos cuadrados a la modelización dinámica. Entre las primeras aportaciones importantes, destacamos las de Kolmogorov (1939), quien proporcionó un tratamiento completo del problema de predicción en modelos estacionarios en tiempo discreto, cuya metodología fue extendida por Krein al caso continuo en 1945. De forma independiente, en 1942, Wiener había obtenido una fórmula explícita basada en cálculos variacionales para el problema de predicción lineal en tiempo continuo.

A partir de los años cincuenta del siglo XX, los estudios se dirigen hacia la debilitación de las restricciones impuestas sobre los procesos objeto de la estimación en los

trabajos previamente desarrollados. Sin embargo, todos los resultados teóricos obtenidos en esta época presentaban graves inconvenientes a la hora de su aplicación práctica, dado que requerían la resolución de complicadas ecuaciones, así como la repetición de todos los cálculos cada vez que se ampliaba el intervalo de observación. Estos inconvenientes supusieron una gran limitación en el desarrollo de las técnicas de estimación y en su aplicación a problemas reales, incluyendo los impulsados por la revolución espacial en los años cincuenta, referidos a la determinación de la posición y velocidad de satélites.

Para resolver estos problemas, Kalman [34], en 1960, desarrolló un algoritmo de estimación recursivo basado en una nueva formulación del problema; con la introducción del *modelo de espacio de estados*, una ecuación lineal para representar la evolución de una señal dinámica, varió la formulación convencional de Wiener y, haciendo uso de una técnica basada en proyecciones ortogonales, obtuvo una expresión recursiva para la obtención de los estimadores. Dicho algoritmo, conocido como *filtro de Kalman*, resuelve, bajo el criterio de mínimos cuadrados, el problema de estimación de una señal lineal discreta a partir de observaciones ruidosas, también lineales, bajo hipótesis de gaussianidad e independencia mutua de los ruidos y de la señal en el instante inicial.

Posteriormente, aparecen numerosos estudios sobre aspectos teóricos y prácticos del filtro de Kalman. Entre ellos, algunos proponen distintas formas de obtener las ecuaciones del filtro; citemos, por ejemplo, Kailath [31], Meditch [39], Anderson y Moore [1], Caines [5], Brammer y Siffing [3], Catlin [8], Korbicz y Bidyuk [35] y Grewal y Andrews [20], [21]. Otros generalizan el filtro de Kalman obteniendo algoritmos para el problema de suavizamiento como Meditch [38] y Kailath-Frost [33] y, además, se han establecido conexiones con otros tipos de estimación (bayesiana, máxima verosimilitud, etc.), como puede verse en Jazwinski [30].

Un recorrido histórico completo y bien documentado sobre el desarrollo de la teoría de estimación y aplicaciones en sistemas dinámicos puede encontrarse en Kailath [32] o en Sorenson [52].

Como hemos indicado, el filtro de Kalman proporciona el estimador de menor error cuadrático medio en sistemas lineales bajo hipótesis de gaussianidad e independencia de los ruidos y de la señal en el instante inicial; sin embargo, cuando estas hipótesis no son satisfechas, dicho filtro sólo proporciona el estimador óptimo en la clase de funciones lineales de las observaciones.

En 1995, De Santis, Germani y Raimondi [49] trataron el problema de estimación en sistemas lineales no gaussianos, deduciendo un algoritmo recursivo para la obtención del estimador óptimo en la clase de transformaciones polinomiales de segundo grado de las observaciones, que proporciona estimadores con menor error cuadrático medio que el filtro de Kalman. Esta técnica de estimación cuadrática fue posteriormente generalizada por Carravetta *et al.* [6], [7] considerando estimadores polinomiales de orden arbitrario.

En la teoría clásica de estimación en sistemas estocásticos lineales, cuyo desarrollo general hemos comentado hasta el momento, se supone que la señal que se desea estimar está siempre presente en las observaciones disponibles; sin embargo, existen situaciones prácticas en las cuales, debido a fallos en la transmisión de la señal o a posibles interrupciones aleatorias en el mecanismo de medidas, las observaciones pueden consistir únicamente en ruido. En estas situaciones, la presencia o no de señal en las observaciones usadas para la estimación debe modelizarse de forma aleatoria, dando lugar a los denominados *sistemas con observaciones inciertas*, que incluyen en la ecuación de observación un ruido escalar multiplicativo para modelizar dicha incertidumbre.

En 1969, Nahi [42] trató por primera vez el problema de estimación en sistemas linea-

---

les con observaciones inciertas. Modelizando la incertidumbre sobre la presencia de señal en las observaciones mediante un proceso estocástico definido por variables de Bernoulli independientes, y considerando ruidos aditivos blancos, independientes e independientes de la señal en el instante inicial, Nahi obtuvo un algoritmo recursivo de filtrado lineal de menor error cuadrático medio con una estructura similar al filtro de Kalman, cuya estabilidad fue posteriormente analizada por Tugnait [54]. Más tarde, Monzingo [40] generalizó el algoritmo de Nahi para resolver el problema de suavizamiento lineal en este mismo tipo de sistemas.

Debemos indicar que, aun bajo hipótesis de gaussianidad sobre los ruidos aditivos y la señal inicial, los sistemas con observaciones inciertas son siempre no gaussianos y, por tanto, el filtro de Nahi sólo proporciona el estimador óptimo en la clase de estimadores lineales. Este hecho condujo a Jaffer y Gupta [28], en 1971, a desarrollar un algoritmo recursivo para la obtención del estimador óptimo (en el sentido de menor error cuadrático medio) en sistemas con observaciones inciertas; sin embargo, aunque teóricamente resolvieron el problema, el algoritmo obtenido no era computacionalmente eficiente, ya que requería un crecimiento exponencial de memoria para su aplicación práctica; de ahí que, a raíz de este trabajo, el estudio del problema de estimación en sistemas con observaciones inciertas se centrara en la búsqueda de estimadores subóptimos que mejorasen al lineal propuesto por Nahi, desarrollándose una extensa literatura sobre el tema.

En ese mismo año, 1971, Rajasekaran *et al.* [48] obtuvieron estimadores lineales de menor error cuadrático medio recursivos para el caso de ruido multiplicativo blanco. La estabilidad de estos estimadores fue estudiada posteriormente por Tugnait [55]. Hadidi y Schwartz [22], estudiaron el problema de estimación eliminando la hipótesis de independencia del ruido multiplicativo y probaron además que, en general, el filtro lineal de

mínimos cuadrados no tiene estructura recursiva, estableciendo una condición necesaria y suficiente para que dicho estimador fuera recursivo. Posteriormente, Wang [56] obtuvo estos mismos resultados utilizando un tratamiento distinto.

En 1981, Monzingo [41], generalizó el estudio realizado por Hadidi y Schwartz analizando el problema de suavizamiento para el tipo de sistemas considerado en [22]. Chow y Birkemeier [9], [10], consideraron un modelo con ruido multiplicativo descrito por una ecuación dinámica y dedujeron una estructura recursiva para el estimador distinta a la de Kalman.

Por otro lado, Hermoso y Linares trataron los problemas de filtrado y suavizamiento en el caso de ruidos aditivos correlados, [23] y [24], respectivamente. En 1997, García-Ligero *et al.* [19] dedujeron un algoritmo de estimación polinomial de segundo grado considerando que la incertidumbre en las observaciones estaba modelizada por variables aleatorias de Bernoulli independientes. Este estudio fue generalizado por Caballero *et al.* en [4] utilizando estimadores polinomiales de orden arbitrario. Nakamori *et al.* abordaron el problema de estimación polinomial usando como información la función de covarianzas del proceso implicado [44] y [45], ya que en muchas situaciones prácticas, el modelo de espacio de estados puede no estar disponible.

Todos los trabajos mencionados hasta el momento tratan el problema de estimación de señales en sistemas estocásticos lineales; sin embargo, una gran parte de los sistemas reales suelen presentar algún tipo de no linealidad, lo que hace que el problema de estimación óptima para este tipo de sistemas sea de difícil resolución; en consecuencia, un gran número de métodos subóptimos han sido desarrollados para abordar el problema de estimación en sistemas no lineales.

Uno de los primeros métodos de estimación en sistemas no lineales, y más extendido,

---

---

es el desarrollado en 1970 por Schmidt [50] con el fin de mejorar los simuladores de vuelo de la NASA, conocido como *filtro de Kalman Extendido*. La metodología de este algoritmo de filtrado se basa en la linealización de las funciones no lineales que describen el sistema, aproximándolas por los términos de primer orden de sus desarrollos en serie de Taylor en los estimadores previamente obtenidos en cada etapa. Una vez obtenida la aproximación del sistema original, se aplica el filtro de Kalman al sistema linealizado.

Aunque el filtro de Kalman extendido ha sido aplicado con notable éxito en muchas situaciones, puede presentar graves inconvenientes cuando las funciones que describen el sistema son severamente no lineales. Por este motivo, se han desarrollado distintas versiones que intentan mejorar las estimaciones proporcionadas por él. Entre ellas se encuentran el *filtro de Kalman extendido iterado*, basado en sucesivos desarrollos en serie de Taylor de las mencionadas funciones, y el *filtro de Kalman extendido de segundo orden*, que tiene en cuenta en la linealización de las funciones los términos de segundo orden de sus desarrollos en serie de Taylor (véase, por ejemplo, Simon [51] y Tanizaki [53]). Estas aproximaciones, generalmente, proporcionan mejores estimaciones que el filtro de Kalman extendido; sin embargo, desde el punto de vista computacional, su aplicación práctica puede ser altamente compleja.

Como alternativa al filtro de Kalman extendido, con el fin de resolver los problemas que éste presenta cuando las funciones que describen el sistema de interés son severamente no lineales, en 1994 Evensen [13] introdujo otro método de estimación de señales modelizadas por ecuaciones no lineales, que conduce al denominado *filtro de Kalman ensemble*. Este filtro pertenece a una categoría conocida como *filtros partículas*, y la idea es basar la estimación en simulaciones de valores de la señal en el instante inicial y de los ruidos que intervienen en el sistema. El filtro de Kalman ensemble ha demostrado gran eficacia

en diversas aplicaciones prácticas y, además, el hecho de tener una simple formulación conceptual y una fácil implementación explica la gran popularidad que ha adquirido en los últimos años. Una buena referencia para la introducción a este filtro puede ser [16].

Las aportaciones sobre filtrado ensemble posteriores al primer trabajo publicado por Evensen han sido numerosas. Ese mismo año, él mismo completó su aportación mostrando la eficacia del filtro de Kalman ensemble sobre el filtro de Kalman extendido mediante la aplicación a un modelo oceanográfico no lineal, [14]. Algunas publicaciones han sido dirigidas a la generalización de este filtro al problema de suavizamiento. La primera formulación que resuelve este problema fue dada por van Leeuwen y Evensen [17]; no obstante, posteriormente, volvieron a revisar dicha formulación y desarrollaron un nuevo algoritmo con mejores propiedades [18]. Otros estudios discuten el potencial del uso del filtro de Kalman ensemble en la estimación de parámetros, por ejemplo el publicado por Evensen en 2003 [15].

Como hemos comentado anteriormente, es de manifiesta importancia el contemplar el problema de estimación en sistemas donde existe la posibilidad de que las observaciones consistan exclusivamente en ruido. Este hecho ha sido extensamente estudiado en el caso lineal; sin embargo, las aportaciones para el caso de sistemas no lineales con observaciones inciertas han sido significativamente inferiores. Entre ellas podemos destacar los trabajos de Nanakara y Yaz [43], Hermoso y Linares [25] y Nakamori *et al.* [47], Hermoso *et al.* [27] que, usando diferentes aproximaciones, presentan distintos algoritmos recursivos de estimación a partir de observaciones inciertas en sistema no lineales.

En términos generales, el objetivo de este trabajo consiste en el estudio del problema de estimación de menor error cuadrático medio de una señal no lineal discreta a partir de observaciones ruidosas, también no lineales, relacionadas con ella, suponiendo

---

conocido el modelo que genera la señal. Después de presentar de manera general las distintas aproximaciones que utilizaremos en nuestro estudio, abordaremos el problema de estimación polinomial de segundo grado en sistemas con certidumbre en las observaciones, para, posteriormente, generalizar los resultados a sistemas con observaciones inciertas.

El desarrollo del trabajo se ha estructurado en tres capítulos, de los que presentamos seguidamente una breve descripción.

- El objetivo del **Capítulo 1** es presentar las bases sobre las técnicas de estimación no lineales que se desarrollan posteriormente. Concretamente, se presenta la metodología del filtro de Kalman extendido, así como sus extensiones anteriormente comentadas, el filtro de Kalman extendido iterado, y el filtro de Kalman extendido de segundo orden. La idea general de estos algoritmos es la de linealizar el sistema y, una vez obtenida la aproximación lineal, aplicar el filtro de Kalman. También se incluye en este capítulo el desarrollo del filtro de Kalman ensemble. Mediante un ejemplo numérico se ilustran los resultados obtenidos con los distintos algoritmos.
- En el **Capítulo 2** se aborda el problema de estimación cuadrática en sistemas no lineales con certidumbre en las observaciones. El objetivo de este capítulo es desarrollar un algoritmo de filtrado, el *filtro extendido cuadrático*, que proporcione estimaciones con menor error cuadrático medio que el filtro de Kalman extendido. Para ello, después de realizar la aproximación lineal del sistema, usamos la técnica propuesta por [49] para construir un nuevo sistema lineal, aumentando los vectores señal y observación con sus potencias de segundo orden. Así, el estimador cuadrático de menor error cuadrático medio de la señal original se obtiene a partir del estimador lineal de la señal del nuevo sistema. Con el fin de mejorar las estimaciones proporcionadas por este algoritmo, se desarrollan también dos versiones del mismo,



siguiendo la metodología del filtro de Kalman extendido iterado y del filtro de Kalman extendido de segundo orden. Por último, se presenta un ejemplo numérico que ilustra los resultados obtenidos en este capítulo.

- En el **Capítulo 3**, aportamos nuevos resultados al problema de estimación en sistemas no lineales con observaciones inciertas. En primer lugar, desarrollamos un algoritmo de filtro extendido cuadrático, utilizando la misma metodología del capítulo anterior. En este caso nos encontramos con que el sistema aumentado no satisface las condiciones necesarias para aplicar el filtro convencional de Nahi, lo que nos ha obligado a desarrollar un nuevo algoritmo recursivo de estimación lineal para dicho sistema. Posteriormente, se presentan dos extensiones del filtro de Kalman ensemble para el caso de observaciones inciertas, el *filtro ensemble* y el *filtro ensemble cuadrático*, en los cuales, además de simular la señal inicial y los ruidos aditivos, se simula el ruido multiplicativo que describe la incertidumbre en las observaciones. El primero de ellos constituye una extensión directa del filtro de Kalman ensemble para sistemas con certidumbre en las observaciones, mientras que el segundo basa la estimación en las observaciones disponibles y sus potencias de segundo grado. Finalmente, como en los capítulos anteriores, se presenta un ejemplo numérico para mostrar el comportamiento de los estimadores propuestos.
- En un **Anexo** se presentan los programas de Matlab correspondientes a los distintos algoritmos propuestos en este trabajo. Dichos programas, en cada iteración, simulan la señal que se desea estimar, así como la observación correspondiente, y proporcionan, tanto las estimaciones como los errores cuadráticos medios asociados a las mismas.

# Capítulo 1

## Estimación en sistemas no lineales

### 1.1. Introducción

El problema de estimación de señales modelizadas por una dinámica no lineal a partir de observaciones ruidosas de funciones, también no lineales, de las mismas ha sido extensamente investigado debido a que una gran cantidad de sistemas reales suelen presentar algún tipo de no linealidad, como sucede, por ejemplo, en diversos ámbitos de aplicación como la Ingeniería, Biología y Economía. La gran dificultad que supone la obtención de estimadores óptimos en este tipo de sistemas ha motivado el desarrollo de un gran número de métodos subóptimos para abordar el problema de estimación.

Una de las primeras y más utilizadas aproximaciones para abordar el problema de estimación no lineal es el *filtro de Kalman extendido* (véase Schmidt [50], Simon [51] y Tanizaki [53], entre otros), basado en la linealización de las funciones que describen el sistema objeto de estudio; considerando únicamente los términos de primer orden del desarrollo en serie de Taylor de tales funciones, se obtiene una aproximación lineal del sistema no lineal original, en la que se aplica el tradicional filtro de Kalman para obtener la estimación de la señal.

Sin embargo, a pesar de su indudable atractivo, debido a su simplicidad y su buen

comportamiento en numerosas situaciones prácticas, el filtro de Kalman extendido puede presentar serias complicaciones cuando las funciones que describen el sistema de interés son severamente no lineales. Por este motivo, se han desarrollado distintas versiones de este filtro con objeto de mejorar las estimaciones proporcionadas por el mismo. Entre ellas se encuentran el *filtro de Kalman extendido iterado*, en el que se realizan sucesivos desarrollos en serie de Taylor de las funciones no lineales que describen las observaciones usadas para la estimación, y el *filtro de Kalman extendido de segundo orden*, con el que se intenta disminuir el error de linealización teniendo en cuenta en las aproximaciones lineales de las funciones que definen el sistema los términos de segundo orden de su desarrollo (véase, por ejemplo, Simon [51] y Tanizaki [53]).

Estas aproximaciones pueden mejorar significativamente las estimaciones obtenidas mediante el filtro de Kalman extendido, aunque en ocasiones su aplicación práctica puede ser altamente compleja desde el punto de vista computacional.

En 1994, Evensen [13] introdujo un nuevo método de estimación de señales en sistemas no lineales, conocido como *filtro de Kalman ensemble*. En principio, este filtro fue considerado únicamente en aplicaciones específicas, como en Oceanografía y Meteorología (Bertino *et al.* [2] y Zheng *et al.* [59], entre otros); sin embargo, su popularidad ha ido creciendo debido a su simple formulación conceptual y a su fácil implementación (véase Evensen [15], [16] y Li y Xiu [36]).

El filtro de Kalman ensemble pertenece a una categoría de filtros conocidos como *filtros partícula*, los cuales utilizan técnicas de estimación basadas en la simulación de valores de la señal inicial y de los procesos ruido que intervienen en el sistema.

El objetivo de este capítulo es presentar las técnicas de estimación previamente citadas en sistemas no lineales estándar, como base para su posterior extensión en los sucesivos

capítulos. El capítulo se estructura de la siguiente forma: en la Sección 1.2 se desarrolla el filtro de Kalman extendido en sistemas no lineales con ruidos aditivos, así como su generalización al caso de ruidos no aditivos. En las secciones 1.3 y 1.4 se describe la metodología de los filtros de Kalman Extendido iterado y de segundo orden, respectivamente, y, por último, en la Sección 1.5, se presenta la formulación del filtro de Kalman ensemble. Finalmente, en la Sección 1.6, se realiza un ejemplo de simulación con la finalidad de ilustrar el comportamiento de los estimadores obtenidos con cada uno de los métodos de estimación presentados en este capítulo, y efectuar comparaciones entre los mismos.

## 1.2. Filtro de Kalman extendido

El filtro de Kalman extendido es un algoritmo recursivo para el problema de estimación de una señal a partir de un conjunto de observaciones ruidosas relacionadas con ella, aplicable al caso en que la ecuación que describe la dinámica de la señal o la ecuación que establece la relación entre las observaciones y la señal es no lineal. La metodología que subyace en esta aproximación al problema de estimación es la linealización de las ecuaciones del sistema original, para la posterior aplicación del filtro de Kalman al sistema linealizado.

En esta sección se describe la metodología del filtro de Kalman extendido para sistemas afectados por ruidos aditivos (Sección 1.2.1) y para el caso general de ruidos no aditivos (Sección 1.2.2).

### 1.2.1. Sistemas no lineales con ruidos aditivos

Sean  $\{x_k; k \geq 0\}$  e  $\{y_k; k \geq 1\}$  los procesos estocásticos que describen la señal a estimar y las observaciones de la misma que han de usarse para la estimación, respectivamente, ambos definidos sobre un espacio de probabilidad  $(\Omega, A, P)$ . Se supone que la

evolución de la señal y la relación entre las observaciones y la señal están descritas por el siguiente sistema de ecuaciones dinámicas:

$$\begin{aligned}x_k &= f_{k-1}(x_{k-1}, u_{k-1}) + w_{k-1}, & k \geq 1 \\y_k &= h_k(x_k) + v_k, & k \geq 1.\end{aligned}\tag{1.1}$$

donde  $f_k : \mathbb{R}^{n+p} \rightarrow \mathbb{R}^n$ ,  $k \geq 0$  y  $h_k : \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $k \geq 1$ , son funciones no lineales, continuas y diferenciables.

Sobre los procesos que intervienen en este sistema se suponen las siguientes hipótesis:

- i) La señal en el instante inicial,  $x_0$ , es un vector aleatorio  $n$ -dimensional, con media cero y matriz de covarianzas conocida,  $P_0$ .
- ii)  $\{u_k; k \geq 0\}$  es un proceso de control, no aleatorio,  $p$ -dimensional.
- iii)  $\{w_k; k \geq 0\}$  es un ruido blanco, centrado, con función de covarianzas  $E[w_k w_k^T] = Q_k$ .
- iv)  $\{v_k; k \geq 1\}$  es un ruido blanco, centrado, con función de covarianzas  $E[v_k v_k^T] = R_k$ .
- v) La señal inicial,  $x_0$ , y los ruidos  $\{w_k; k \geq 0\}$  y  $\{v_k; k \geq 1\}$  son mutuamente independientes.

Como ya se ha indicado, la idea básica del filtro de Kalman extendido es la aproximación del sistema bajo estudio mediante un sistema lineal al que se aplican las ecuaciones del filtro de Kalman. La linealización de las ecuaciones del sistema en cada instante de tiempo se realiza aproximando las funciones correspondientes mediante el primer término de su desarrollo en serie de Taylor alrededor de los estimadores de la señal obtenidos en la etapa anterior del algoritmo, como se muestra a continuación.

### Linealización de la ecuación de la señal

Partiendo del estimador de menor error cuadrático medio de la señal en el instante inicial,  $\hat{x}_{0/0} = 0$ , y notando  $\hat{x}_{k-1/k-1}$  al filtro de la señal en el instante  $k - 1$ , obtenido en la etapa anterior del algoritmo a partir de las observaciones  $y_1, \dots, y_{k-1}$ , en el instante  $k$  se linealiza la función  $f_{k-1}$  aproximándola por el primer término de su desarrollo en serie de Taylor alrededor de  $\hat{x}_{k-1/k-1}$ :

$$f_{k-1}(x_{k-1}, u_{k-1}) \approx f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}) + \left. \frac{\partial f_{k-1}(x, u_{k-1})}{\partial x} \right|_{x=\hat{x}_{k-1/k-1}} (x_{k-1} - \hat{x}_{k-1/k-1}), \quad k \geq 1.$$

De esta forma, la ecuación de la señal se aproxima por la siguiente ecuación lineal:

$$x_k = F_{k-1}x_{k-1} + \tilde{u}_{k-1} + w_{k-1}, \quad k \geq 1,$$

donde  $F_{k-1}$  y  $\tilde{u}_{k-1}$  están definidas como

$$\begin{aligned} F_{k-1} &= \left. \frac{\partial f_{k-1}(x, u_{k-1})}{\partial x} \right|_{x=\hat{x}_{k-1/k-1}} \\ \tilde{u}_{k-1} &= f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}) - F_{k-1}\hat{x}_{k-1/k-1}. \end{aligned} \quad (1.2)$$

### Linealización de la ecuación de las observaciones

Una vez linealizada la ecuación de la señal en el instante  $k$ , se obtiene el predictor de la señal en dicho instante,  $\hat{x}_{k/k-1}$ , a partir de las observaciones  $y_1, \dots, y_{k-1}$ , como en el filtro de Kalman, y, con un procedimiento similar al anterior, se aplica ahora el desarrollo en serie de Taylor a la función  $h_k$  alrededor de  $\hat{x}_{k/k-1}$ ,

$$h_k(x_k) \approx h_k(\hat{x}_{k/k-1}) + \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k-1}} (x_k - \hat{x}_{k/k-1}), \quad k \geq 1.$$

Así, si definimos

$$\begin{aligned} H_k &= \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k-1}} \\ z_k &= h_k(\hat{x}_{k/k-1}) - H_k\hat{x}_{k/k-1}, \end{aligned} \quad (1.3)$$

la ecuación de observación se aproxima por la siguiente:

$$y_k = H_k x_k + z_k + v_k, \quad k \geq 1.$$

De esta forma, el sistema no lineal queda aproximado por

$$\begin{aligned} x_k &= F_{k-1} x_{k-1} + \tilde{u}_{k-1} + w_{k-1}, \quad k \geq 1 \\ y_k &= H_k x_k + z_k + v_k, \quad k \geq 1, \end{aligned} \tag{1.4}$$

donde  $F_{k-1}$ ,  $\tilde{u}_{k-1}$ ,  $H_k$  y  $z_k$  están definidas en (1.2) y (1.3).

La estimación de la señal se realiza aplicando el filtro de Kalman al sistema (1.4) (véase el Apéndice al final de este capítulo), obteniendo el siguiente algoritmo:

- Se parte de las condiciones iniciales:

$$\begin{aligned} \hat{x}_{0/0} &= E[x_0] = 0 \\ P_{0/0} &= E[(x_0 - \hat{x}_{0/0})(x_0 - \hat{x}_{0/0})^T] = P_0. \end{aligned}$$

- En cada instante  $k \geq 1$ , partiendo del filtro obtenido en la etapa anterior del algoritmo,  $\hat{x}_{k-1/k-1}$ , y de la matriz de covarianzas del error de filtrado,  $P_{k-1/k-1}$ , se calcula el predictor,  $\hat{x}_{k/k-1}$ , y la correspondiente matriz de covarianzas del error,  $P_{k/k-1}$ , mediante las siguientes expresiones:

$$\begin{aligned} \hat{x}_{k/k-1} &= f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}), \quad k \geq 1 \\ P_{k/k-1} &= F_{k-1} P_{k-1/k-1} F_{k-1}^T + Q_{k-1}, \quad k \geq 1, \end{aligned} \tag{1.5}$$

donde  $F_{k-1}$  está dada por la expresión (1.2).

- A continuación, se actualiza la estimación de la señal  $x_k$  incorporando la nueva información aportada por la observación  $y_k$ , obteniendo de este modo el filtro,  $\hat{x}_{k/k}$ , y la matriz de covarianzas del error de filtrado,  $P_{k/k}$ :

$$\begin{aligned} \hat{x}_{k/k} &= \hat{x}_{k/k-1} + K_k [y_k - h_k(\hat{x}_{k/k-1})], \quad k \geq 1 \\ P_{k/k} &= [I_n - K_k H_k] P_{k/k-1}, \quad k \geq 1, \end{aligned} \tag{1.6}$$

donde  $H_k$  está definida en (1.3),  $I_n$  denota la matriz identidad de dimensiones  $n \times n$  y  $K_k$ , la matriz de ganancia del filtro, es obtenida mediante la expresión

$$K_k = P_{k/k-1} H_k^T [H_k P_{k/k-1} H_k^T + R_k]^{-1}. \quad (1.7)$$

### 1.2.2. Sistemas no lineales con ruidos no aditivos

En esta sección presentamos las ecuaciones del filtro de Kalman extendido para sistemas no lineales en los que las perturbaciones ruidosas que afectan a la evolución de la señal y/o a las observaciones no son necesariamente aditivas. El sistema, en tal caso, puede describirse de forma genérica, como sigue:

$$\begin{aligned} x_k &= f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}), \quad k \geq 1 \\ y_k &= h_k(x_k, v_k), \quad k \geq 1, \end{aligned} \quad (1.8)$$

donde las hipótesis sobre los procesos que intervienen son las mismas que las correspondientes al sistema (1.1) considerado en la Sección 1.2.1.

Para aplicar la metodología del filtro de Kalman extendido a este tipo de sistemas, y conseguir un sistema linealizado, hay que tener en cuenta que, ahora, el desarrollo en serie de Taylor se aplica a funciones de dos variables; salvo esto, la forma de proceder es similar a la del caso anterior, como puede verse a continuación.

#### Linealización de la ecuación de la señal

Para obtener la aproximación lineal de la ecuación de la señal, se aplica el desarrollo en serie de Taylor de la función  $f_{k-1}$  en el punto  $(\hat{x}_{k-1/k-1}, \hat{w}_{k-1/k-1})$ , donde  $\hat{w}_{k-1/k-1}$  es el estimador de  $w_{k-1}$  basado en las observaciones  $y_1, \dots, y_{k-1}$ . Debido a las hipótesis de independencia impuesta sobre el sistema es claro que  $\hat{w}_{k-1/k-1} = 0$  y, por tanto, la señal



se linealiza mediante la siguiente expresión:

$$x_k \approx f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}, 0) + \left. \frac{\partial f_{k-1}(x, u_{k-1}, w)}{\partial x} \right|_{(x,w)=(\hat{x}_{k-1/k-1}, 0)} (x_{k-1} - \hat{x}_{k-1/k-1}) \\ + \left. \frac{\partial f_{k-1}(x, u_{k-1}, w)}{\partial w} \right|_{(x,w)=(\hat{x}_{k-1/k-1}, 0)} w_{k-1}, \quad k \geq 1.$$

Entonces, si se definen  $F_{k-1}$ ,  $\tilde{u}_{k-1}$ ,  $\tilde{w}_{k-1}$  y  $L_{k-1}$  como

$$F_{k-1} = \left. \frac{\partial f_{k-1}(x, u_{k-1}, w)}{\partial x} \right|_{(x,w)=(\hat{x}_{k-1/k-1}, 0)} \\ \tilde{u}_{k-1} = f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}, 0) - F_{k-1} \hat{x}_{k-1/k-1} \\ \tilde{w}_{k-1} = L_{k-1} w_{k-1} \\ L_{k-1} = \left. \frac{\partial f_{k-1}(x, u_{k-1}, w)}{\partial w} \right|_{(x,w)=(\hat{x}_{k-1/k-1}, 0)},$$

se obtiene la ecuación linealizada de la señal:

$$x_k = F_{k-1} x_{k-1} + \tilde{u}_{k-1} + \tilde{w}_{k-1}, \quad k \geq 1.$$

### Linealización de la ecuación de las observaciones

Con un procedimiento análogo al anterior, la aproximación lineal de la ecuación de observación se obtiene desarrollando en serie de Taylor la función  $h_k$  en el punto  $(\hat{x}_{k/k-1}, \hat{v}_{k/k-1})$ , donde  $\hat{v}_{k/k-1} = 0$  por el mismo razonamiento que en el caso anterior. Así,

$$y_k \approx h_k(\hat{x}_{k/k-1}, 0) + \left. \frac{\partial h_k(x, v)}{\partial x} \right|_{(x,v)=(\hat{x}_{k/k-1}, 0)} (x_k - \hat{x}_{k/k-1}) + \left. \frac{\partial h_k(x, v)}{\partial v} \right|_{(x,v)=(\hat{x}_{k/k-1}, 0)} v_k, \quad k \geq 1,$$

y definiendo  $H_k$ ,  $z_k$ ,  $\tilde{v}_k$  y  $M_k$  como

$$H_k = \left. \frac{\partial h_k(x, v)}{\partial x} \right|_{(x,v)=(\hat{x}_{k/k-1}, 0)} \\ z_k = h_k(\hat{x}_{k/k-1}, 0) - H_k \hat{x}_{k/k-1} \\ \tilde{v}_k = M_k v_k \\ M_k = \left. \frac{\partial h_k(x, v)}{\partial v} \right|_{(x,v)=(\hat{x}_{k/k-1}, 0)},$$

la ecuación de observación se aproxima por

$$y_k = H_k x_k + z_k + \tilde{v}_k, \quad k \geq 1.$$

El sistema resultante que se obtiene tras aplicar la aproximación lineal tiene, por tanto, la siguiente forma:

$$\begin{aligned} x_k &= F_{k-1} x_{k-1} + \tilde{u}_{k-1} + \tilde{w}_{k-1}, \quad k \geq 1 \\ y_k &= H_k x_k + z_k + \tilde{v}_k, \quad k \geq 1, \end{aligned} \tag{1.9}$$

y, de nuevo, la estimación de la señal se realiza aplicando el filtro de Kalman al sistema (1.9) obtenido tras la linealización. Así, partiendo de las condiciones iniciales,

$$\begin{aligned} \hat{x}_{0/0} &= E[x_0] = 0 \\ P_{0/0} &= E[(x_0 - \hat{x}_{0/0})(x_0 - \hat{x}_{0/0})^T] = P_0, \end{aligned}$$

el algoritmo es el siguiente:

$$\begin{aligned} \hat{x}_{k/k} &= \hat{x}_{k/k-1} + K_k [y_k - h_k(\hat{x}_{k/k-1}, 0)], \quad k \geq 1 \\ \hat{x}_{k/k-1} &= f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}, 0), \quad k \geq 1 \\ K_k &= P_{k/k-1} H_k^T [H_k P_{k/k-1} H_k^T + M_k R_k M_k^T]^{-1}, \quad k \geq 1 \\ P_{k/k} &= [I_n - K_k H_k] P_{k/k-1}, \quad k \geq 1 \\ P_{k/k-1} &= F_{k-1} P_{k-1/k-1} F_{k-1}^T + L_{k-1} Q_{k-1} L_{k-1}^T, \quad k \geq 1. \end{aligned}$$

### 1.3. Filtro de Kalman extendido iterado

El filtro de Kalman extendido iterado es un algoritmo de estimación de señales en sistemas no lineales desarrollado a fin de mejorar las estimaciones proporcionadas por el filtro de Kalman extendido (véase Simon [51] y Tanizaki [53]).

Como se ha mostrado en la Sección 1.2, en la etapa de actualización del filtro de Kalman extendido se utiliza el desarrollo en serie de Taylor de la función  $h_k$  alrededor

del predictor,  $\hat{x}_{k/k-1}$ , para obtener el filtro de la señal,  $\hat{x}_{k/k}$ ; el desarrollo se realiza en el predictor porque es la mejor estimación de la señal  $x_k$  que se tiene antes de disponer de la medida  $y_k$ . Sin embargo, una vez obtenida esta observación, y calculado el filtro, se dispone de una estimación mejorada de la señal que puede usarse para desarrollar nuevamente la función  $h_k$  en serie de Taylor alrededor de la misma, y volver a realizar los cálculos para conseguir una nueva estimación.

De forma más específica, considerando el sistema no lineal (1.1) y aplicando las ecuaciones (1.5) y (1.6), se obtiene el predictor, el filtro y las correspondientes matrices de covarianzas del error:

$$\begin{aligned}\hat{x}_{k/k-1} &= f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}), \quad k \geq 1 \\ \hat{x}_{k/k} &= \hat{x}_{k/k-1} + K_k[y_k - h_k(\hat{x}_{k/k-1})], \quad k \geq 1 \\ P_{k/k-1} &= F_{k-1}P_{k-1/k-1}F_{k-1}^T + Q_{k-1}, \quad k \geq 1 \\ P_{k/k} &= [I_n - K_k H_k]P_{k/k-1}, \quad k \geq 1,\end{aligned}$$

donde  $F_{k-1}$ ,  $H_k$  y  $K_k$  se encuentran definidas en (1.2), (1.3) y (1.7), respectivamente.

A continuación, se desarrolla la función  $h_k$  en serie de Taylor alrededor del filtro,  $\hat{x}_{k/k}$ :

$$h_k(x_k) \approx h_k(\hat{x}_{k/k}) + \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k}} (x_k - \hat{x}_{k/k}), \quad k \geq 1.$$

Entonces, definiendo  $H_k^0$  y  $z_k^0$  como

$$\begin{aligned}H_k^0 &= \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k}} \\ z_k^0 &= h_k(\hat{x}_{k/k}) - H_k^0 \hat{x}_{k/k},\end{aligned}$$

la ecuación de observación se aproxima por

$$y_k = H_k^0 x_k + z_k^0 + v_k, \quad k \geq 1.$$

Aplicando entonces el filtro de Kalman se obtienen las siguientes ecuaciones para el cálculo del nuevo estimador y la correspondiente matriz de covarianzas del error:

$$\begin{aligned}\hat{x}_{k/k}^1 &= \hat{x}_{k/k-1} + K_k^0 [y_k - h_k(\hat{x}_{k/k}) - H_k^0 (\hat{x}_{k/k-1} - \hat{x}_{k/k})], \quad k \geq 1 \\ P_{k/k}^1 &= [I_n - K_k^0 H_k^0] P_{k/k-1}, \quad k \geq 1,\end{aligned}$$

donde  $K_k^0$  se calcula mediante la siguiente expresión:

$$K_k^0 = P_{k/k-1} H_k^{0T} [H_k^0 P_{k/k-1} H_k^{0T} + R_k]^{-1}.$$

Este procedimiento puede aplicarse reiteradamente, obteniendo un algoritmo de filtrado partiendo de las condiciones iniciales definidas por el filtro de Kalman extendido,

$$\begin{aligned}\hat{x}_{k/k}^0 &= \hat{x}_{k/k}, \quad k \geq 1 \\ P_{k/k}^0 &= P_{k/k}, \quad k \geq 1,\end{aligned}$$

y realizando el siguiente bucle de iteraciones para el cálculo de los sucesivos filtros y matrices de covarianzas de los errores:

$$\begin{aligned}H_k^i &= \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k}^i} \\ K_k^i &= P_{k/k-1} H_k^{iT} [H_k^i P_{k/k-1} H_k^{iT} + R_k]^{-1} \\ \hat{x}_{k/k}^{i+1} &= \hat{x}_{k/k-1} + K_k^i [y_k - h_k(\hat{x}_{k/k}^i) - H_k^i (\hat{x}_{k/k-1} - \hat{x}_{k/k}^i)] \\ P_{k/k}^{i+1} &= [I - K_k^i H_k^i] P_{k/k-1},\end{aligned}$$

donde  $i = 0, \dots, N$ , siendo  $N$  el número de iteraciones.

En cada instante  $k$ , el filtro de la señal  $x_k$  está dado por  $\hat{x}_{k/k}^{N+1}$ , y la matriz de covarianzas del error de filtrado por  $P_{k/k}^{N+1}$ .

Debemos indicar que si el sistema bajo estudio es del tipo (1.8) el procedimiento a seguir para la estimación de la señal mediante el filtro de Kalman extendido iterado es similar; simplemente hay que tener en cuenta que el desarrollo en serie de Taylor se aplica a funciones de dos variables.

## 1.4. Filtro de Kalman extendido de segundo orden

El filtro de Kalman extendido de segundo orden es una versión del filtro de Kalman extendido, con el que se intenta reducir el error que se comete en la aproximación lineal del sistema de interés. Ambos algoritmos utilizan la misma metodología pero, con objeto de reducir el error cometido al aproximar el sistema, el algoritmo de estimación que se presenta en esta sección incluye los términos de segundo orden de los desarrollos en serie de Taylor de las funciones que intervienen en el mismo (véase Simon [51] y Wishner *et al.* [58]); de ahí el nombre de filtro de Kalman extendido de segundo orden.

Para describir de forma concreta la metodología a seguir se considera el sistema de ecuaciones no lineal (1.1) (para los sistemas del tipo (1.8) el procedimiento a seguir es similar).

### Linealización de la ecuación de la señal

Como ya se ha indicado, la linealización de la ecuación de la señal se realiza considerando el desarrollo en serie de Taylor de la función  $f_{k-1}$  en el filtro obtenido en la etapa  $k-1$ ,  $\hat{x}_{k-1/k-1}$ , e incluyendo en la aproximación los términos de segundo orden de dicho desarrollo; así, para  $k \geq 1$ :

$$f_{k-1}(x_{k-1}, u_{k-1}) \approx f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}) + \frac{\partial f_{k-1}(x, u_{k-1})}{\partial x} \Big|_{x=\hat{x}_{k-1/k-1}} (x_{k-1} - \hat{x}_{k-1/k-1}) \\ + \frac{1}{2} \sum_{i=1}^n \phi_i(x_{k-1} - \hat{x}_{k-1/k-1})^T \frac{\partial^2 f_{k-1}^i(x, u_{k-1})}{\partial x^2} \Big|_{x=\hat{x}_{k-1/k-1}} (x_{k-1} - \hat{x}_{k-1/k-1}),$$

donde  $f_{k-1}^i$  es la componente  $i$ -ésima de  $f_{k-1}$  y  $\phi_i$  es el vector de dimensión  $n \times 1$  cuyos elementos son cero, excepto el de la posición  $i$ , que es uno.

Notemos que en el término de segundo orden aparece una forma cuadrática que hace

que esta aproximación no sea lineal; sin embargo, usando la siguiente propiedad:

$$\begin{aligned} & (x_{k-1} - \hat{x}_{k-1/k-1})^T \frac{\partial^2 f_{k-1}^i(x, u_{k-1})}{\partial x^2} \Big|_{x=\hat{x}_{k-1/k-1}} (x_{k-1} - \hat{x}_{k-1/k-1}) \\ &= Tr \left[ \frac{\partial^2 f_{k-1}^i(x, u_{k-1})}{\partial^2 x} \Big|_{x=\hat{x}_{k-1/k-1}} (x_{k-1} - \hat{x}_{k-1/k-1})(x_{k-1} - \hat{x}_{k-1/k-1})^T \right], \end{aligned}$$

y aproximando el factor aleatorio  $(x_{k-1} - \hat{x}_{k-1/k-1})(x_{k-1} - \hat{x}_{k-1/k-1})^T$  por su valor medio, que es precisamente la covarianza del error del filtro,  $P_{k-1/k-1}$ , la función  $f_{k-1}(x_{k-1}, u_{k-1})$  se aproxima por

$$\begin{aligned} f_{k-1}(x_{k-1}, u_{k-1}) &\approx F_{k-1}x_{k-1} + f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}) - F_{k-1}\hat{x}_{k-1/k-1} \\ &\quad + \frac{1}{2} \sum_{i=1}^n \phi_i Tr \left[ \frac{\partial^2 f_{k-1}^i(x, u_{k-1})}{\partial x^2} \Big|_{x=\hat{x}_{k-1/k-1}} P_{k-1/k-1} \right], \quad k \geq 1, \end{aligned}$$

donde  $F_{k-1}$  y  $\tilde{u}_{k-1}$  se definen como:

$$\begin{aligned} F_{k-1} &= \frac{\partial f_{k-1}(x, u_{k-1})}{\partial x} \Big|_{x=\hat{x}_{k-1/k-1}} \\ \tilde{u}_{k-1} &= f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}) - F_{k-1}\hat{x}_{k-1/k-1} \\ &\quad + \frac{1}{2} \sum_{i=1}^n \phi_i Tr \left[ \frac{\partial^2 f_{k-1}^i(x, u_{k-1})}{\partial x^2} \Big|_{x=\hat{x}_{k-1/k-1}} P_{k-1/k-1} \right]. \end{aligned} \tag{1.10}$$

En definitiva, la ecuación de la señal queda aproximada por la siguiente:

$$x_k = F_{k-1}x_{k-1} + \tilde{u}_{k-1} + w_{k-1}, \quad k \geq 1.$$

## Linealización de la ecuación de las observaciones

Con un procedimiento similar al anterior, esto es, aplicando el desarrollo en serie de Taylor de la función  $h_k$  alrededor del predictor,  $\hat{x}_{k/k-1}$ , incluyendo en la aproximación los términos de segundo orden de este desarrollo, y aproximando  $(x_k - \hat{x}_{k/k-1})(x_k - \hat{x}_{k/k-1})^T$  por su valor medio, la matriz de covarianzas del error de predicción, tenemos:

$$h_k(x_k) \approx H_k x_k + h_k(\hat{x}_{k/k-1}) - H_k \hat{x}_{k/k-1} + \frac{1}{2} \sum_{i=1}^m \phi_i Tr \left[ \frac{\partial^2 h_k^i(x)}{\partial x^2} \Big|_{x=\hat{x}_{k/k-1}} P_{k/k-1} \right] \quad k \geq 1,$$

donde  $h_k^i$  es la componente  $i$ -ésima de  $h_k$  y

$$H_k = \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k-1}}. \quad (1.11)$$

Entonces, definiendo  $z_k$  como

$$z_k = h_k(\hat{x}_{k/k-1}) - H_k \hat{x}_{k/k-1} + \frac{1}{2} \sum_{i=1}^m \phi_i Tr \left[ \left. \frac{\partial^2 h_k^i(x)}{\partial x^2} \right|_{x=\hat{x}_{k/k-1}} P_{k/k-1} \right], \quad (1.12)$$

la ecuación de las observaciones se aproxima por

$$y_k = H_k x_k + z_k + v_k, \quad k \geq 1.$$

Una vez linealizadas las ecuaciones, el sistema no lineal se aproxima por

$$\begin{aligned} x_k &= F_{k-1} x_{k-1} + \tilde{u}_{k-1} + w_{k-1}, \quad k \geq 1 \\ y_k &= H_k x_k + z_k + v_k, \quad k \geq 1, \end{aligned} \quad (1.13)$$

donde  $F_{k-1}$ ,  $\tilde{u}_{k-1}$ ,  $H_k$  y  $z_k$  están definidas en (1.10), (1.11) y (1.12). Entonces, aplicando el filtro de Kalman (Apéndice A) a este sistema lineal, se obtiene el siguiente algoritmo:

$$\begin{aligned} \hat{x}_{k/k} &= \hat{x}_{k/k-1} + K_k \left[ y_k - h_k(\hat{x}_{k/k-1}) - \frac{1}{2} \sum_{i=1}^m \phi_i Tr \left( \left. \frac{\partial^2 h_k^i(x)}{\partial x^2} \right|_{\hat{x}_{k/k-1}} P_{k/k-1} \right) \right], \quad k \geq 1 \\ \hat{x}_{k/k-1} &= f_{k-1}(\hat{x}_{k-1/k-1}, u_{k-1}) + \frac{1}{2} \sum_{i=1}^n \phi_i \left[ \left. \frac{\partial^2 f_{k-1}^i(x, u_{k-1})}{\partial x^2} \right|_{\hat{x}_{k-1/k-1}} P_{k-1/k-1} \right], \quad k \geq 1 \\ K_k &= P_{k/k-1} H_k^T [H_k P_{k/k-1} H_k^T + R_k]^{-1}, \quad k \geq 1 \\ P_{k/k} &= [I_n - K_k H_k] P_{k/k-1}, \quad k \geq 1 \\ P_{k/k-1} &= F_{k-1} P_{k-1/k-1} F_{k-1}^T + Q_{k-1}, \quad k \geq 1, \end{aligned}$$

cuyas condiciones iniciales son

$$\begin{aligned} \hat{x}_{0/0} &= E[x_0] = 0 \\ P_{0/0} &= E[(x_0 - \hat{x}_{0/0})(x_0 - \hat{x}_{0/0})^T] = P_0. \end{aligned}$$

## 1.5. Filtro de Kalman ensemble

Como ya se ha indicado, el filtro de Kalman ensemble pertenece a la categoría de los conocidos como *filtros partícula*, que utilizan técnicas simulación de los procesos que describen el sistema de interés.

Para describir la metodología de este filtro consideramos de nuevo el sistema no lineal con ruidos aditivos (1.1):

$$\begin{aligned}x_k &= f_{k-1}(x_{k-1}, u_{k-1}) + w_{k-1}, \quad k \geq 1 \\y_k &= h_k(x_k) + v_k, \quad k \geq 1,\end{aligned}$$

satisfaciendo las hipótesis establecidas en la Sección 1.2.1.

Al igual que los anteriores, en cada iteración, el algoritmo de filtrado Ensemble actúa en dos etapas, la de predicción y la de actualización, como se describe a continuación.

### Etapa de predicción

En el instante  $k = 1$ , se parte de un conjunto de puntos muestrales,  $\hat{x}_{0/0}^1, \dots, \hat{x}_{0/0}^q$ , simulados a partir de la distribución de la señal inicial,  $x_0$ . Para cada instante  $k > 1$ , se parte de un conjunto de  $q$  filtros,  $\hat{x}_{k-1/k-1}^1, \dots, \hat{x}_{k-1/k-1}^q$ , que se habrán obtenido en la iteración previa del algoritmo. Propagando cada uno de estos filtros a través de la ecuación de la señal, se obtiene el siguiente conjunto de predictores:

$$\hat{x}_{k/k-1}^i = f_{k-1}(\hat{x}_{k-1/k-1}^i, u_{k-1}) + w_{k-1}^i, \quad i = 1, \dots, q,$$

donde los valores  $w_{k-1}^i$  son simulados de la distribución del ruido  $w_{k-1}$ .

Una vez obtenido el conjunto de predictores se calcula el predictor de la señal,  $\hat{x}_{k/k-1}$ , y la correspondiente matriz de covarianzas del error,  $P_{k/k-1}$ , por la media y cuasicovarianza



de éstos; es decir:

$$\hat{x}_{k/k-1} = \frac{1}{q} \sum_{i=1}^q \hat{x}_{k/k-1}^i, \quad k \geq 1$$

$$P_{k/k-1} = \frac{1}{q-1} \sum_{i=1}^q (\hat{x}_{k/k-1}^i - \hat{x}_{k/k-1})(\hat{x}_{k/k-1}^i - \hat{x}_{k/k-1})^T, \quad k \geq 1.$$

### Etapa de actualización

El punto de inicio de esta etapa es el conjunto de predictores  $\hat{x}_{k/k-1}^1, \dots, \hat{x}_{k/k-1}^q$ , obtenido en la etapa anterior. Este conjunto de puntos se propaga a través de la ecuación de observación:

$$\hat{y}_{k/k-1}^i = h_k(\hat{x}_{k/k-1}^i) + v_k^i, \quad i = 1, \dots, q,$$

donde los valores  $v_k^i$  son muestreados de la distribución del ruido  $v_k$ .

A partir de los puntos transformados,  $\hat{y}_{k/k-1}^1, \dots, \hat{y}_{k/k-1}^q$ , se calcula el predictor de la observación,  $\hat{y}_{k/k-1}$ , y su correspondiente matriz de covarianzas del error,  $P_{k/k-1}^y$ , así como la matriz de covarianzas cruzadas del error del predictor de la señal y de la observación,  $P_{k/k-1}^{xy}$ , mediante la media y cuasicovarianzas correspondientes; es decir:

$$\hat{y}_{k/k-1} = \frac{1}{q} \sum_{i=1}^q \hat{y}_{k/k-1}^i, \quad k \geq 1$$

$$P_{k/k-1}^y = \frac{1}{q-1} \sum_{i=1}^q (\hat{y}_{k/k-1}^i - \hat{y}_{k/k-1})(\hat{y}_{k/k-1}^i - \hat{y}_{k/k-1})^T, \quad k \geq 1$$

$$P_{k/k-1}^{xy} = \frac{1}{q-1} \sum_{i=1}^q (\hat{x}_{k/k-1}^i - \hat{x}_{k/k-1})(\hat{y}_{k/k-1}^i - \hat{y}_{k/k-1})^T, \quad k \geq 1.$$

Una vez obtenidos estos valores, se calculan los filtros correspondientes mediante las ecuaciones del filtro de Kalman:

$$\hat{x}_{k/k}^i = \hat{x}_{k/k-1}^i + P_{k/k-1}^{xy} (P_{k/k-1}^y)^{-1} (y_k - \hat{y}_{k/k-1}^i), \quad i = 1, \dots, q.$$

Finalmente, el filtro de la señal,  $\hat{x}_{k/k}$ , y la matriz de covarianza del error de filtrado,  $P_{k/k}$ , se calculan como

$$\hat{x}_{k/k} = \frac{1}{q} \sum_{i=1}^q \hat{x}_{k/k}^i, \quad k \geq 1$$

$$P_{k/k} = \frac{1}{q-1} \sum_{i=1}^q (\hat{x}_{k/k}^i - \hat{x}_{k/k})(\hat{x}_{k/k}^i - \hat{x}_{k/k})^T, \quad k \geq 1.$$

### *Versión alternativa del filtro de Kalman ensemble*

Como alternativa al procedimiento descrito anteriormente en la etapa de actualización, el conjunto de predictores puede propagarse solamente a través de la función  $h_k$  y, en tal caso, las matrices de covarianzas  $P_{k/k-1}^y$  y  $P_{k/k-1}^{xy}$  se obtienen a partir de las correspondientes a los puntos transformados y de la matriz de covarianzas del ruido  $v_k$ :

$$\hat{z}_{k/k-1}^i = h_k(\hat{x}_{k/k-1}^i), \quad i = 1, \dots, q$$

$$\hat{z}_{k/k-1} = \frac{1}{q} \sum_{i=1}^q \hat{z}_{k/k-1}^i, \quad k \geq 1$$

$$P_{k/k-1}^y = \frac{1}{q-1} \sum_{i=1}^q (\hat{z}_{k/k-1}^i - \hat{z}_{k/k-1})(\hat{z}_{k/k-1}^i - \hat{z}_{k/k-1})^T + R_k, \quad k \geq 1$$

$$P_{k/k-1}^{xy} = \frac{1}{q-1} \sum_{i=1}^q (\hat{x}_{k/k-1}^i - \hat{x}_{k/k-1})(\hat{z}_{k/k-1}^i - \hat{z}_{k/k-1})^T, \quad k \geq 1.$$

Entonces, los filtros se calculan a partir de las siguientes ecuaciones:

$$\hat{x}_{k/k}^i = \hat{x}_{k/k-1}^i + P_{k/k-1}^{xy} (P_{k/k-1}^y)^{-1} [y_k - \hat{z}_{k/k-1}^i - v_k^i], \quad i = 1, \dots, q,$$

donde los valores  $v_k^i$  son muestreados de la distribución del ruido  $v_k$ .

## 1.6. Ejemplo numérico

Con la finalidad de ilustrar los resultados expuestos en este capítulo se han implementado en Matlab los distintos algoritmos, aplicándose a un modelo no lineal correspondiente a una señal escalar,  $\{x_k; k \geq 0\}$ , que evoluciona de acuerdo a la siguiente dinámica:

$$x_k = \frac{1}{x_{k-1}^2 + 3} + w_{k-1}, \quad k \geq 1,$$

donde

- i)* La señal en el instante inicial,  $x_0$ , es una variable aleatoria gaussiana, con media cero y varianza  $P_0 = 1$ .
- ii)* El proceso  $\{w_k; k \geq 0\}$  es un ruido blanco gaussiano, centrado, con varianzas  $Q_k = 1, k \geq 0$ .

Las observaciones usadas para la estimación de la señal están descritas por la siguiente ecuación:

$$y_k = x_k^2 + \exp(x_k) + v_k, \quad k \geq 1, \quad (1.14)$$

donde el proceso  $\{v_k; k \geq 1\}$  es un ruido blanco gaussiano, centrado, con varianzas  $R_k = 1, k \geq 1$ .

Para cada uno de los algoritmos, se ha realizado un programa que simula la señal que se desea estimar, así como las observaciones correspondientes, y proporciona las estimaciones de filtrado. En todos los casos, se han considerado 1000 simulaciones y 50 iteraciones del algoritmo para cada simulación.

Para medir la bondad de las estimaciones proporcionadas por cada algoritmo se utiliza la media de las desviaciones cuadráticas de los valores simulados de la señal y sus

estimaciones; es decir, el error cuadrático medio,

$$ECM_k = \frac{1}{1000} \sum_{s=1}^{1000} (x_k^s - \hat{x}_{k/k}^s)^2, \quad k = 1, \dots, 50,$$

donde, para cada instante  $k$ ,  $x_k^s$  denota el valor de la señal generado en la simulación  $s$  y  $\hat{x}_{k/k}^s$  denota el filtro calculado con el correspondiente algoritmo en dicha simulación.

Como ya se ha indicado, el filtro de Kalman extendido iterado y el filtro de Kalman extendido de segundo orden son modificaciones del filtro de Kalman extendido que intentan mejorar las estimaciones proporcionadas por éste. A continuación, verificamos este hecho comparando los errores cuadráticos medios de las estimaciones obtenidas con cada uno de estos algoritmos.

El filtro de Kalman extendido iterado se basa en sucesivos desarrollos en serie de Taylor de la función  $h_k$ . En el ejemplo que nos ocupa, considerando 20 iteraciones de este proceso en cada instante de tiempo  $k$ , observamos una evidente mejora sobre el filtro de Kalman extendido, como se muestra en la Figura 1.1.

En la Figura 1.2, para las mismas simulaciones de la señal y las observaciones consideradas en la Figura 1.1, se presentan los errores cuadráticos medios del filtro de Kalman extendido y del filtro de Kalman extendido de segundo orden, observándose, de forma global, la mejora en el comportamiento del segundo frente al primero.

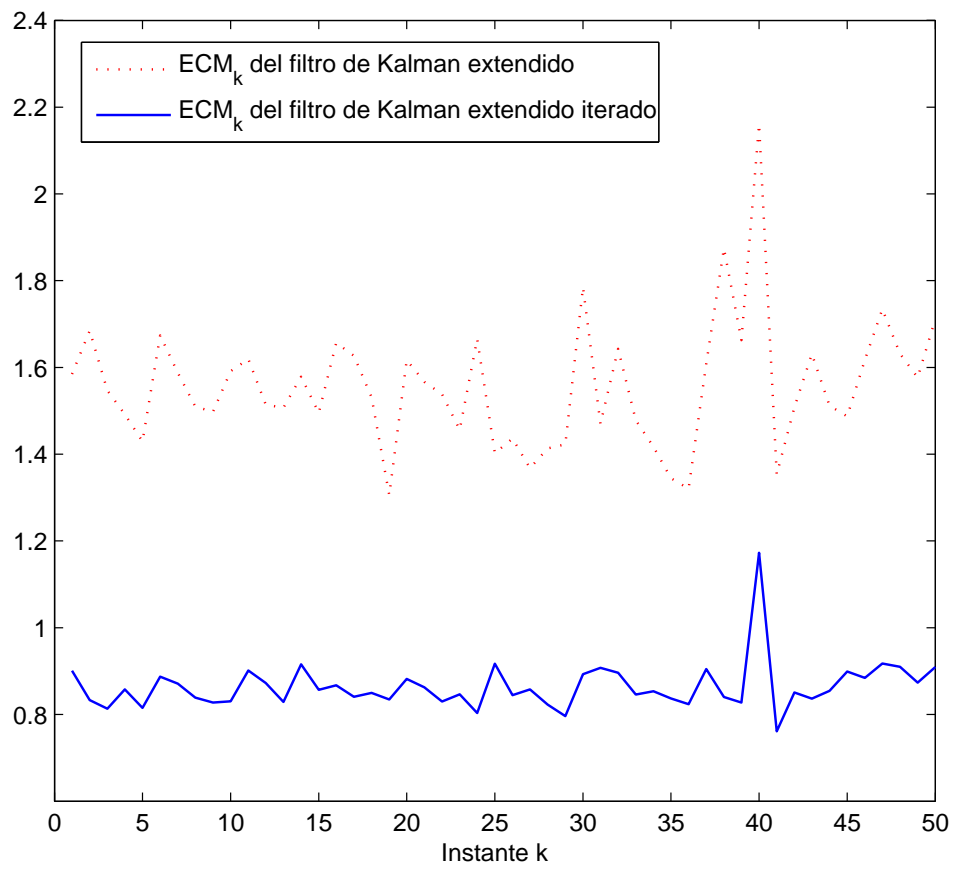


Figura 1.1: Error cuadrático medio del filtro de Kalman extendido y del filtro de Kalman extendido iterado.

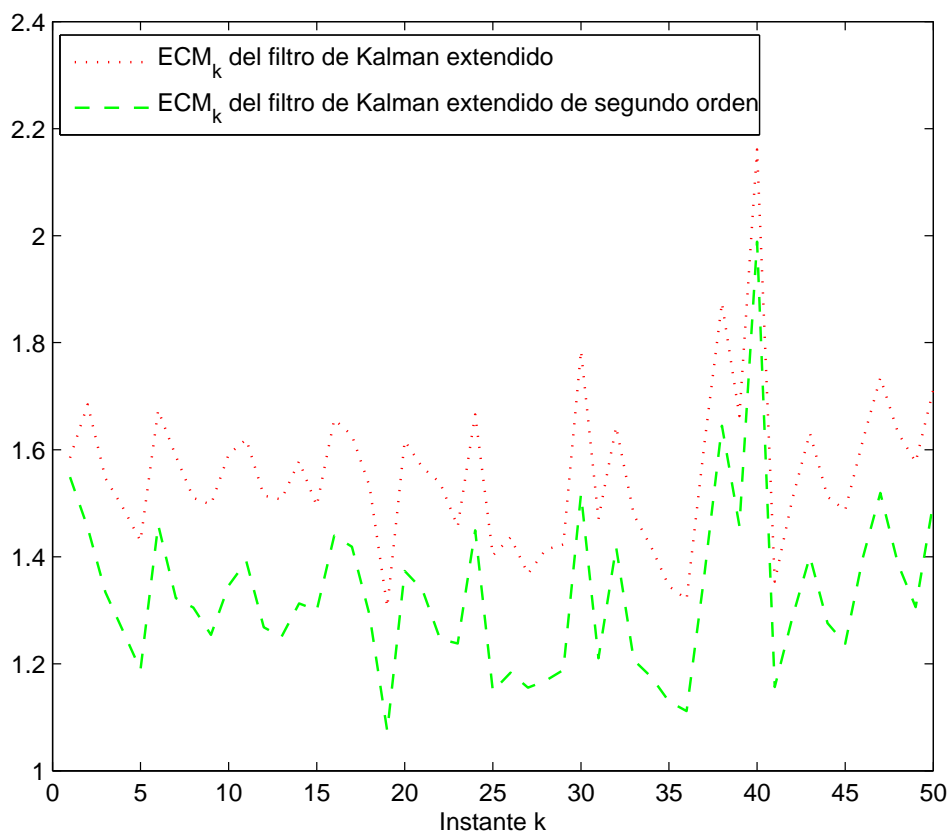


Figura 1.2: Error cuadrático medio del filtro de Kalman extendido y del filtro de Kalman extendido de segundo orden.

A continuación, en la Figura 1.3 se muestran los errores cuadráticos medios de las estimaciones obtenidas con el filtro de Kalman extendido y el filtro de Kalman ensemble para las mismas simulaciones de la señal y las observaciones consideradas en las figuras anteriores. Los resultados obtenidos con los algoritmos de las dos versiones del filtro de Kalman ensemble expuestas en este capítulo son prácticamente los mismos y, por lo tanto, no se distingue entre ellos. Tomando 100 puntos muestrales simulados de las distribuciones de la señal inicial y de los ruidos, se observa que las estimaciones proporcionadas por el

filtro de Kalman ensemble mejoran notablemente a las del filtro de Kalman extendido.

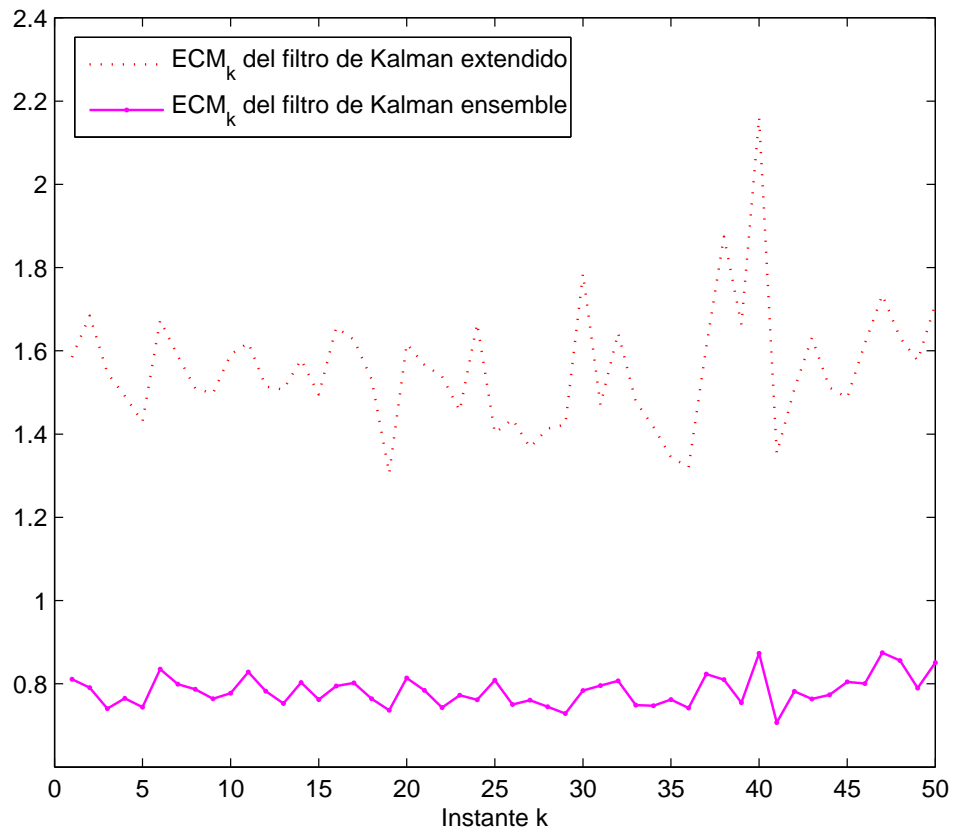


Figura 1.3: Error cuadrático medio del filtro de Kalman extendido y del filtro de Kalman ensemble.

Para finalizar, en la Figura 1.4, se presentan conjuntamente los errores cuadráticos medios de las estimaciones obtenidas con los distintos algoritmos con el fin de visualizar globalmente el comportamiento de los mismos.

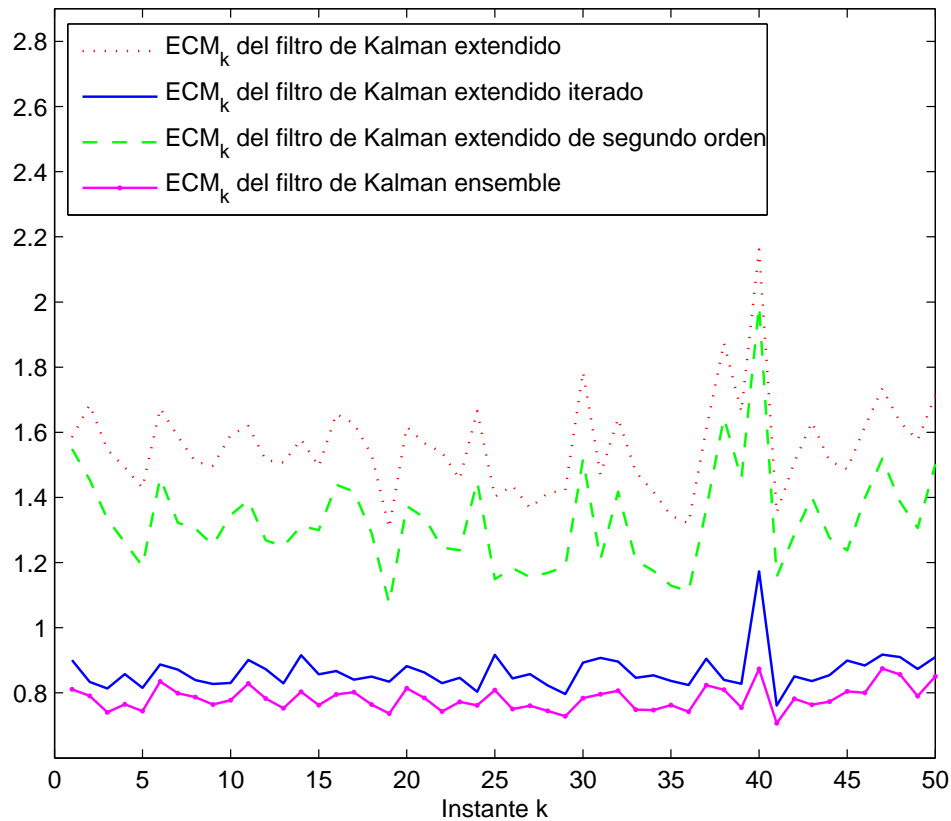


Figura 1.4: Error cuadrático medio de los filtros de Kalman Extendido, Kalman Extendido iterado, Kalman Extendido de segundo orden y Kalman Ensemble.



## Apéndice: Filtro de Kalman

El filtro de Kalman es un algoritmo recursivo para el problema de estimación mínimo cuadrática de una señal, a partir de observaciones ruidosas relacionadas con ella, que proporciona el estimador óptimo bajo hipótesis de gaussianidad de los procesos que intervienen en el problema.

Se considera el siguiente sistema:

$$x_k = F_{k-1}x_{k-1} + G_{k-1}u_{k-1} + w_{k-1}, \quad k \geq 1$$

$$y_k = H_k x_k + v_k, \quad k \geq 1,$$

donde  $\{x_k; k \geq 0\}$  es un proceso estocástico  $n$ -dimensional definido sobre un espacio probabilístico  $(\Omega, A, P)$ , que describe la señal a estimar, e  $\{y_k; k \geq 1\}$  es el proceso que describe las observaciones usadas para la estimación. Se suponen las siguientes hipótesis:

- i) La condición inicial,  $x_0$ , es un vector aleatorio gaussiano, con media cero y matriz de covarianzas conocida,  $P_0$ .
- ii)  $\{u_k; k \geq 0\}$  es un proceso de control  $p$ -dimensional.
- iii)  $F_k, G_k$  y  $H_k$  son matrices conocidas de dimensión  $n \times n$ ,  $n \times p$  y  $m \times n$ , respectivamente.
- iv)  $\{w_k; k \geq 0\}$  es un ruido blanco gaussiano, centrado, con función de covarianzas  $E[w_k w_j^T] = Q_k \delta_{k-j}$ , donde  $\delta$  denota la función delta de Kronecker.
- v)  $\{v_k; k \geq 1\}$  es un ruido blanco gaussiano, centrado, con función de covarianzas  $E[v_k v_j^T] = R_k \delta_{k-j}$ .
- vi) El estado inicial,  $x_0$ , y los ruidos  $\{w_k; k \geq 0\}$  y  $\{v_k; k \geq 1\}$  son mutuamente independientes.

El algoritmo viene dado por las siguientes ecuaciones:

$$\hat{x}_{k/k} = \hat{x}_{k/k-1} + K_k[y_k - H_k\hat{x}_{k/k-1}], \quad k \geq 1$$

$$\hat{x}_{k/k-1} = F_{k-1}\hat{x}_{k-1/k-1} + G_{k-1}u_{k-1}, \quad k \geq 1$$

$$K_k = P_{k/k-1}H_k^T[H_kP_{k/k-1}H_k^T + R_k]^{-1}, \quad k \geq 1$$

$$P_{k/k} = [I_n - K_kH_k]P_{k/k-1}, \quad k \geq 1$$

$$P_{k/k-1} = F_{k-1}P_{k-1/k-1}F_{k-1}^T + Q_{k-1}, \quad k \geq 1,$$

partiendo de las condiciones iniciales

$$\hat{x}_{0/0} = E[x_0] = 0$$

$$P_{0/0} = E[(x_0 - \hat{x}_{0/0})(x_0 - \hat{x}_{0/0})^T] = P_0.$$



# Capítulo 2

## Estimación cuadrática en sistemas no lineales con certidumbre

### 2.1. Introducción

Los filtros extendidos que se han presentado en el Capítulo 1 resultan de la aplicación del filtro de Kalman a distintos sistemas lineales que se usan para aproximar al sistema original no lineal. Sin embargo, como es conocido, el filtro de Kalman sólo proporciona el estimador óptimo de la señal, en sistemas lineales, bajo hipótesis de gaussianidad e independencia mutua de los ruidos y de la señal inicial; cuando esta hipótesis no se verifica, el estimador óptimo no es función lineal de las observaciones, y el filtro de Kalman sólo proporciona el estimador de menor error cuadrático medio de la señal en la clase de funciones lineales.

En general, incluso en sistemas lineales, la obtención del estimador óptimo en el sentido de menor error cuadrático medio, la esperanza condicionada de la señal dadas las observaciones, no puede llevarse a cabo mediante algoritmos recursivos computacionalmente eficientes, y el problema de estimación se reconduce hacia la obtención de estimadores subóptimos que mejoren al lineal, intentando disminuir la complejidad computacional

que conlleva el cálculo del óptimo.

En 1995, De Santis, Germani y Raimondi [49] abordaron el problema de estimación en sistemas lineales no gaussianos, deduciendo un algoritmo recursivo que proporciona el estimador de menor error cuadrático medio entre todas las transformaciones polinomiales de segundo grado de las observaciones. Ya que dicha clase de transformaciones contiene a las lineales, los estimadores obtenidos mediante este algoritmo mejoran, en el sentido de menor error cuadrático medio, a los obtenidos mediante la aplicación del filtro de Kalman en sistemas no gaussianos.

El procedimiento usado en [49] para la obtención del algoritmo de estimación cuadrática consiste en construir un nuevo sistema a partir del original, aumentando los vectores señal y observación con sus potencias de segundo grado, definidas mediante el producto de Kronecker, y aplicar en este sistema aumentado un algoritmo de filtrado lineal. Este procedimiento fue posteriormente extendido para la obtención de algoritmos recursivos de estimación polinomial de grado arbitrario (véase Carravetta *et al.* [6]), habiendo quedada probada la efectividad de estos estimadores frente a la de los lineales mediante su aplicación a diversos problemas de procesamiento de señales, como restauración de imágenes (Dalla Mora *et al.* [11] y Uppala y Sarh [57]) o estimación de señales acústicas (Nakamori *et al.* [46]).

El objetivo de este capítulo es utilizar esta técnica de estimación cuadrática en los sistemas linealizados que se obtienen para aproximar los sistemas no lineales, con el fin de mejorar los estimadores proporcionados por el filtro de Kalman extendido, el filtro de Kalman extendido iterado y el filtro de Kalman extendido de segundo orden. Una primera aproximación a este estudio, que desarrollaremos en este capítulo, puede verse en [26].

El capítulo comienza con una sección dedicada a presentar la interpretación geométrica

---

de los estimadores polinomiales de segundo grado en términos de proyecciones ortogonales. En la Sección 2.3 se realiza la descripción del sistema no lineal sobre el que abordaremos el problema de estimación cuadrática. Seguidamente, en la Sección 2.4, se deduce un primer algoritmo de filtrado cuadrático, el *filtro extendido cuadrático*, aplicando la técnica de [49] al sistema linealizado en el que se obtiene el filtro de Kalman extendido. En las secciones 2.5 y 2.6, se presentan el *filtro extendido cuadrático iterado* y el *filtro extendido cuadrático de segundo orden*, respectivamente, con los que se pretende mejorar los estimadores obtenidos en la Sección 2.4 en el mismo sentido a como se ha realizado el estudio en el Capítulo 1. Por último, se realiza un ejemplo numérico para ilustrar el comportamiento de los estimadores cuadráticos propuestos en este capítulo.

## 2.2. Estimación cuadrática y proyecciones ortogonales

El objetivo de esta sección es mostrar la interpretación geométrica del problema de estimación de menor error cuadrático medio en términos de proyecciones ortogonales.

Sea  $\mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$  el conjunto de clases de equivalencia de vectores aleatorios  $n$ -dimensionales definidos sobre el espacio probabilístico  $(\Omega, \mathcal{A}, P)$ , con momentos de segundo orden finito. Como es conocido,  $\mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$  tiene estructura de espacio de Hilbert con el producto escalar  $\langle U, W \rangle = E[U^T W]$ .

A continuación mostramos que si  $X \in \mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$  e  $Y \in \mathcal{L}_{\mathbb{R}^m}^2(\Omega, \mathcal{A}, P)$ , el estimador lineal de menor error cuadrático medio de  $X$  basado en  $Y$  coincide con la proyección ortogonal de  $X$  en el subespacio de funciones lineales  $n$ -dimensionales del vector  $Y$ .

Denotemos por  $H_n(Y')$  al subespacio lineal de  $\mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$  obtenido mediante transformaciones lineales  $n$ -dimensionales del vector  $Y' = (1 \ Y^T)^T \in \mathbb{R}^{m+1}$ , y por  $H_n^\perp(Y')$  al

complemento ortogonal de  $H_n(Y')$  en  $\mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$ :

$$H_n^\perp(Y') = \{W \in \mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P) : \langle W, U \rangle = 0, \forall U \in H_n(Y')\}.$$

Ya que  $H_n(Y')$  es un subespacio lineal cerrado de  $\mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$ , el *teorema de la proyección ortogonal* establece que todo vector  $X \in \mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$  se puede descomponer de forma única como suma de un vector  $\bar{X} \in H_n(Y')$ , denominado *proyección ortogonal de  $X$  sobre  $H_n(Y')$* , y un vector  $\tilde{X} \in H_n^\perp(Y')$ :

$$X = \bar{X} + \tilde{X}, \quad \bar{X} \in H_n(Y'), \quad \tilde{X} \in H_n^\perp(Y').$$

Mostramos a continuación que la proyección ortogonal,  $\bar{X}$ , minimiza el error cuadrático medio cometido al aproximar  $X$  por una transformación lineal de  $Y'$ . En efecto, ya que  $H_n(Y')$  es un espacio lineal, si tomamos un vector  $U \in H_n(Y')$ , el vector  $\bar{X} - U \in H_n(Y')$  y, por tanto, es ortogonal a  $X - \bar{X} = \tilde{X} \in H_n^\perp(Y')$ ; entonces:

$$\begin{aligned} E[(X - U)^T(X - U)] &= E[((X - \bar{X}) + (\bar{X} - U))^T((X - \bar{X}) + (\bar{X} - U))] \\ &= E[(X - \bar{X})^T(X - \bar{X})] + E[(\bar{X} - U)^T(\bar{X} - U)] \\ &\geq E[(X - \bar{X})^T(X - \bar{X})], \end{aligned}$$

lo que efectivamente muestra que el error cuadrático medio cometido al aproximar  $X$  por un elemento del espacio  $H_n(Y')$  se minimiza en la proyección ortogonal; ésta es, por tanto, el estimador lineal de menor error cuadrático medio de  $X$  basado en  $Y$ , que será notado en lo sucesivo como

$$\hat{X}^L = \Pi(X)|_{H_n(Y')}.$$

Como hemos indicado anteriormente, el objetivo de este capítulo es obtener aproximaciones de los estimadores de menor error cuadrático medio en la clase de funciones polinómicas de segundo grado de los vectores en los que se basa la estimación. Estos estimadores, que serán referidos en lo sucesivo como *estimadores cuadráticos*, pueden

también obtenerse en términos de proyecciones ortogonales, considerando los subespacios apropiados, como mostramos a continuación.

Si el objetivo es estimar el vector  $X \in \mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$  mediante una función polinómica de segundo grado del vector  $Y$ , se considera el vector

$$Y'' = (1 \quad Y^T \quad Y^{[2]T})^T \in \mathbb{R}^{m^2+m+1},$$

donde  $Y^{[2]} = Y \otimes Y \in \mathbb{R}^{m^2}$ , denotando  $\otimes$  el producto de Kronecker [37].

Suponiendo que el vector  $Y$  tiene momentos de cuarto orden finitos,  $E[Y^{[2]T}Y^{[2]}] < \infty$ , es claro que el vector aumentado  $Y''$  pertenece al espacio  $\mathcal{L}_{\mathbb{R}^{m^2+m+1}}^2(\Omega, \mathcal{A}, P)$ .

Entonces, como hemos mostrado anteriormente, el estimador cuadrático de menor error cuadrático medio de  $X$  basado en  $Y$  (función lineal de  $Y$  e  $Y^{[2]}$ ) es la proyección ortogonal de  $X$  sobre  $H_n(Y'')$ , el subespacio lineal cerrado de  $\mathcal{L}_{\mathbb{R}^n}^2(\Omega, \mathcal{A}, P)$  obtenido mediante transformaciones lineales de  $Y''$ ; este estimador será denotado por

$$\hat{X}^Q = \Pi(X)|_{H_n(Y'')}.$$

Así, el estimador cuadrático de menor error cuadrático medio se puede obtener como un estimador lineal de menor error cuadrático medio considerando los espacios apropiados.

Notemos que ambos estimadores, tanto el lineal como el cuadrático, son proyecciones ortogonales sobre subespacios lineales, y teniendo en cuenta que  $H_n(Y') \subset H_n(Y'')$ , se deduce que el error asociado al estimador cuadrático es menor o igual que el asociado al estimador lineal,

$$E[(X - \hat{X}^Q)^T(X - \hat{X}^Q)] \leq E[(X - \hat{X}^L)^T(X - \hat{X}^L)],$$

lo que motiva el estudio del problema de estimación cuadrática en el intento de mejorar los resultados obtenidos con la consideración de estimadores lineales.



A continuación establecemos las propiedades de los procesos que intervienen en el sistema necesarias para abordar el problema de estimación cuadrática.

## 2.3. Descripción del sistema

Sea  $\{x_k; k \geq 0\}$  el proceso estocástico  $n$ -dimensional que describe la señal a estimar, e  $\{y_k; k \geq 1\}$  el proceso  $m$ -dimensional que describe las observaciones usadas para la estimación.

Se supone que la señal evoluciona en el tiempo a través de una función no lineal, y que dicha evolución está perturbada por un ruido aditivo  $\{w_k; k \geq 0\}$ ; se supone también que las observaciones que se usarán para la estimación de la señal son de funciones no lineales de la misma, afectadas por un ruido aditivo,  $\{v_k; k \geq 1\}$ . Por lo tanto, el sistema bajo estudio se describe por las siguientes ecuaciones:

$$\begin{aligned} x_k &= f_{k-1}(x_{k-1}) + w_{k-1}, & k \geq 1 \\ y_k &= h_k(x_k) + v_k, & k \geq 1. \end{aligned} \tag{2.1}$$

Para abordar el problema de estimación cuadrática de la señal  $x_k$  a partir de las observaciones  $y_1, \dots, y_k$ , se consideran las potencias Kronecker de segundo orden de los vectores correspondientes,

$$\begin{aligned} x_k^{[2]} &= x_k \otimes x_k \in \mathbb{R}^{n^2}, \\ y_k^{[2]} &= y_k \otimes y_k \in \mathbb{R}^{m^2}, \end{aligned}$$

y en virtud de los resultados presentados en la Sección 2.2, si se supone que todos los vectores considerados en este problema tienen momentos de segundo orden finitos (esto es,  $E[x_k^T x_k] < \infty$ , y  $E[y_j^{[2]T} y_j^{[2]}] < \infty$ ,  $j = 1, \dots, k$ ), el estimador cuadrático de menor error cuadrático medio de la señal  $x_k$  es su proyección ortogonal sobre el espacio de transformaciones lineales de  $y_1, \dots, y_k$  y sus potencias de segundo orden,  $y_1^{[2]}, \dots, y_k^{[2]}$ .

Debemos indicar que la existencia de los momentos de cuarto orden de las observaciones requiere la existencia de los correspondientes de la señal. Así, para el estudio del problema planteado se suponen las siguientes hipótesis sobre el sistema (2.1):

- i)* La señal en el instante inicial,  $x_0$ , es un vector aleatorio  $n$ -dimensional, con momentos hasta el cuarto orden finitos y conocidos, que se denotan de la siguiente forma:

$$E[x_0] = \mu_0,$$

$$E[(x_0 - \mu_0)(x_0 - \mu_0)^T] = P_0,$$

$$E\left[(x_0 - \mu_0)\left(x_0^{[2]} - E[x_0^{[2]}]\right)^T\right] = P_0^{(3)},$$

$$E\left[\left(x_0^{[2]} - E[x_0^{[2]}]\right)\left(x_0^{[2]} - E[x_0^{[2]}]\right)^T\right] = P_0^{(4)}.$$

- ii)*  $\{w_k; k \geq 0\}$  es un ruido blanco  $n$ -dimensional con momentos hasta el cuarto orden conocidos:

$$E[w_k] = 0,$$

$$E[w_k w_k^T] = Q_k,$$

$$E\left[w_k\left(w_k^{[2]} - E[w_k^{[2]}]\right)^T\right] = Q_k^{(3)},$$

$$E\left[\left(w_k^{[2]} - E[w_k^{[2]}]\right)\left(w_k^{[2]} - E[w_k^{[2]}]\right)^T\right] = Q_k^{(4)}.$$

- iii)*  $\{v_k; k \geq 1\}$  es un ruido blanco  $m$ -dimensional con momentos hasta el cuarto orden conocidos:

$$E[v_k] = 0,$$

$$E[v_k v_k^T] = R_k,$$

$$E\left[v_k\left(v_k^{[2]} - E[v_k^{[2]}]\right)^T\right] = R_k^{(3)},$$

$$E\left[\left(v_k^{[2]} - E[v_k^{[2]}]\right)\left(v_k^{[2]} - E[v_k^{[2]}]\right)^T\right] = R_k^{(4)}.$$

- iv)*  $f_k : \mathbb{R}^n \rightarrow \mathbb{R}^n$  y  $h_k : \mathbb{R}^n \rightarrow \mathbb{R}^m$  son funciones no lineales, continuas y diferenciables.
- v)* La señal inicial,  $x_0$ , y los ruidos  $\{w_k; k \geq 0\}$  y  $\{v_k; k \geq 1\}$  son mutuamente independientes.

## 2.4. Filtro extendido cuadrático

Como se ha comentado en el Capítulo 1, uno de los métodos de estimación más usado en sistemas no lineales es el filtro de Kalman extendido, cuya base es la linealización del sistema aplicando el desarrollo en serie de Taylor a las funciones no lineales que lo definen; una vez obtenida la aproximación lineal, se aplica el filtro de Kalman.

El objetivo de esta sección es aplicar la metodología del filtro de Kalman extendido para obtener un algoritmo recursivo de estimación cuadrática; esto es, linealizar el sistema, y obtener estimadores basados en las observaciones linealizadas y sus potencias de segundo grado.

El procedimiento a seguir, una vez obtenido el sistema linealizado, es construir un nuevo sistema en el que los vectores señal y observación se obtienen añadiendo a los vectores señal y observación del sistema linealizado sus potencias de segundo grado, definidas mediante el producto de Kronecker. El estimador cuadrático de la señal del sistema inicial se obtendrá entonces a partir del estimador lineal de la señal aumentada basado en las observaciones aumentadas.

### 2.4.1. Linealización del sistema

Para tratar el problema de estimación cuadrática con la metodología del filtro de Kalman extendido, comenzamos linealizando las ecuaciones que definen el sistema (2.1), como se ha expuesto en el Capítulo 1.

En cada instante de tiempo  $k \geq 1$ , la linealización de la ecuación de la señal se realiza aplicando el desarrollo en serie de Taylor de la función  $f_{k-1}$  en el filtro obtenido en el instante anterior, que notaremos  $\hat{x}_{k-1/k-1}^Q$ , y la ecuación linealizada de las observaciones se obtiene aplicando el desarrollo en serie de Taylor de la función  $h_k$  en el predictor,  $\hat{x}_{k/k-1}^Q$ . Así, considerando los términos de primer orden de los desarrollos correspondientes, las ecuaciones del sistema en cada instante se aproximan por las siguientes:

$$\begin{aligned} x_k &= F_{k-1}x_{k-1} + u_{k-1} + w_{k-1}, & k \geq 1 \\ y_k &= H_k x_k + z_k + v_k, & k \geq 1, \end{aligned} \quad (2.2)$$

donde  $F_{k-1}$ ,  $u_{k-1}$ ,  $H_k$  y  $z_k$  están definidas como sigue:

$$\begin{aligned} F_{k-1} &= \left. \frac{\partial f_{k-1}(x)}{\partial x} \right|_{x=\hat{x}_{k-1/k-1}^Q} \\ u_{k-1} &= f_{k-1}(\hat{x}_{k-1/k-1}^Q) - F_{k-1}\hat{x}_{k-1/k-1}^Q \\ H_k &= \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k-1}^Q} \\ z_k &= h_k(\hat{x}_{k/k-1}^Q) - H_k\hat{x}_{k/k-1}^Q. \end{aligned}$$

Con objeto de simplificar los desarrollos posteriores, centramos los vectores señal y observación, obteniendo

$$\begin{aligned} \bar{x}_k &= F_{k-1}\bar{x}_{k-1} + w_{k-1}, & k \geq 1 \\ \bar{y}_k &= H_k\bar{x}_k + v_k, & k \geq 1, \end{aligned} \quad (2.3)$$

donde  $\bar{x}_k = x_k - E[x_k]$  e  $\bar{y}_k = y_k - E[y_k]$ .

Teniendo en cuenta las ecuaciones del sistema (2.2), las medias de la señal aproximada,  $E[x_k]$ , pueden obtenerse recursivamente y, a partir de ellas, las de las observaciones,  $E[y_k]$ , mediante las siguientes expresiones:

$$\begin{aligned} E[x_k] &= F_{k-1}E[x_{k-1}] + u_{k-1}, & k \geq 1 \\ E[y_k] &= H_k E[x_k] + z_k, & k \geq 1. \end{aligned}$$

Ya que el espacio lineal generado por un conjunto de vectores coincide con el generado por el conjunto de vectores centrados, el estimador cuadrático de  $x_k$  en el sistema (2.2) se obtendrá a partir de las observaciones centradas,  $\bar{y}_1, \dots, \bar{y}_k$ , correspondientes al sistema (2.3), y sus potencias de segundo orden,  $\bar{y}_1^{[2]}, \dots, \bar{y}_k^{[2]}$ .

Por tanto, en lo que sigue, abordamos el problema de estimación cuadrática en el sistema centrado (2.3). Comenzamos definiendo el sistema aumentado y estudiando sus propiedades.

### 2.4.2. Sistema aumentado. Propiedades

Como hemos indicado en la Sección 2.2, el problema de estimación cuadrática se puede reformular como un problema de estimación lineal, aumentando los vectores observación con sus potencias de segundo grado. Para llevar a cabo este tratamiento del problema en el sistema (2.3), aumentamos los vectores señal y observación añadiendo sus correspondientes potencias Kronecker de segundo grado:

$$X_k = \begin{pmatrix} \bar{x}_k \\ \bar{x}_k^{[2]} \end{pmatrix} \in \mathbb{R}^{n+n^2}, \quad Y_k = \begin{pmatrix} \bar{y}_k \\ \bar{y}_k^{[2]} \end{pmatrix} \in \mathbb{R}^{m+m^2}.$$

Entonces, el vector formado por las  $n$  primeras componentes del estimador lineal de la señal aumentada,  $X_k$ , basado en las observaciones aumentadas,  $Y_1, \dots, Y_k$ , proporciona el estimador cuadrático de  $\bar{x}_k$  basado en  $\bar{y}_1, \dots, \bar{y}_k$  o, equivalentemente, el estimador cuadrático de  $\bar{x}_k$  basado en  $y_1, \dots, y_k$ . Una vez obtenido éste, el estimador requerido (estimador cuadrático de  $x_k$  basado en  $y_1, \dots, y_k$ ) se obtendrá sumándole la media del vector original,  $E[x_k]$ .

Así, nuestro objetivo concreto en esta sección es abordar el problema de estimación lineal de  $X_k$  a partir de  $Y_1, \dots, Y_k$ .

Comenzamos analizando la dinámica de los vectores aumentados, con el fin de comprobar si el modelo corresponde a un sistema lineal con las propiedades requeridas para aplicar el filtro de Kalman. Para ello, es necesario calcular las potencias de segundo grado de los vectores señal y observación centrados,  $\bar{x}_k^{[2]}$  e  $\bar{y}_k^{[2]}$ , para las que, teniendo en cuenta las propiedades del producto de Kronecker, se obtienen las expresiones especificadas a continuación (en ellas, y en todo lo que sigue,  $K_a$  denota la matriz conmutación, e  $I_a$  la matriz identidad de dimensiones  $a \times a$ ).

### Potencias de segundo grado de la señal

Teniendo en cuenta la ecuación de la señal del sistema (2.3) se tiene que

$$\bar{x}_k^{[2]} = F_{k-1}^{[2]} \bar{x}_{k-1}^{[2]} + (F_{k-1} \bar{x}_{k-1}) \otimes w_{k-1} + w_{k-1} \otimes (F_{k-1} \bar{x}_{k-1}) + w_{k-1}^{[2]},$$

y, definiendo

$$\Phi_k = (I_{n^2} + K_{n^2})[(F_k \bar{x}_k) \otimes w_k] + w_k^{[2]}, \quad k \geq 0,$$

obtenemos la siguiente expresión:

$$\bar{x}_k^{[2]} = F_{k-1}^{[2]} \bar{x}_{k-1}^{[2]} + \Phi_{k-1}, \quad k \geq 1.$$

### Potencias de segundo grado de la observación

Procediendo de forma análoga con la observación del sistema (2.3), se obtiene que

$$\bar{y}_k^{[2]} = H_k^{[2]} \bar{x}_k^{[2]} + (H_k \bar{x}_k) \otimes v_k + v_k \otimes (H_k \bar{x}_k) + v_k^{[2]},$$

y, por tanto, si notamos

$$\Psi_k = (I_{m^2} + K_{m^2})[(H_k \bar{x}_k) \otimes v_k] + v_k^{[2]}, \quad k \geq 1,$$

tenemos que

$$\bar{y}_k^{[2]} = H_k^{[2]} \bar{x}_k^{[2]} + \Psi_k, \quad k \geq 1.$$

Teniendo en cuenta las expresiones anteriores, es fácil comprobar que los vectores señal y observación aumentados satisfacen

$$\begin{aligned} X_k &= \bar{F}_{k-1} X_{k-1} + W_{k-1}, \quad k \geq 1, \\ Y_k &= \bar{H}_k X_k + V_k, \quad k \geq 1, \end{aligned}$$

donde

$$\begin{aligned} \bar{F}_k &= \begin{pmatrix} F_k & 0 \\ 0 & F_k^{[2]} \end{pmatrix}, & \bar{H}_k &= \begin{pmatrix} H_k & 0 \\ 0 & H_k^{[2]} \end{pmatrix}, \\ W_k &= \begin{pmatrix} w_k \\ \Phi_k \end{pmatrix}, & V_k &= \begin{pmatrix} v_k \\ \Psi_k \end{pmatrix}. \end{aligned}$$

De nuevo, para simplificar los desarrollos posteriores, centramos los vectores de este sistema, obteniendo:

$$\begin{aligned} \bar{X}_k &= \bar{F}_{k-1} \bar{X}_{k-1} + \bar{W}_{k-1}, \quad k \geq 1 \\ \bar{Y}_k &= \bar{H}_k \bar{X}_k + \bar{V}_k, \quad k \geq 1. \end{aligned} \tag{2.4}$$

donde

$$\bar{X}_k = X_k - E[X_k], \quad \bar{W}_k = W_k - E[W_k], \quad \bar{V}_k = V_k - E[V_k], \quad \bar{Y}_k = Y_k - E[Y_k].$$

En este sistema lineal centrado, que denominaremos *sistema aumentado*, obtendremos un algoritmo recursivo de estimación lineal, y el estimador cuadrático de la señal  $x_k$  se obtendrá a partir del estimador lineal de la señal aumentada,  $\bar{X}_k$ , basado en las observaciones aumentadas  $\bar{Y}_1, \dots, \bar{Y}_k$ .

Para abordar el problema de estimación lineal en el sistema aumentado, necesitamos conocer las propiedades de segundo orden de los procesos que intervienen en el mismo, que se especifican en las siguientes proposiciones.

**Proposición 2.1.** *El proceso  $\{\bar{W}_k; k \geq 0\}$  es blanco, centrado, y con matrices de covarianzas dadas por*

$$\bar{Q}_k = \begin{pmatrix} Q_k & Q_k^{(3)} \\ Q_k^{(3)T} & \bar{Q}_k^{22} \end{pmatrix},$$

siendo

$$\bar{Q}_k^{22} = (I_{n^2} + K_{n^2}) [(F_k P_k F_k^T) \otimes Q_k] (I_{n^2} + K_{n^2}) + Q_k^{(4)},$$

donde  $P_k = E[\bar{x}_k \bar{x}_k^T]$  se obtiene recursivamente a partir de  $P_0 = E[\bar{x}_0 \bar{x}_0^T]$  por la relación

$$P_k = F_{k-1} P_{k-1} F_{k-1}^T + Q_{k-1}, \quad k \geq 1.$$

**Demostración.** Claramente, el ruido  $\{\bar{W}_k; k \geq 0\}$  es centrado por definición. A continuación, haciendo uso de las propiedades del producto de Kronecker y de las hipótesis de independencia impuestas en la Sección 2.3 sobre el sistema original, se obtienen las expresiones especificadas para las cajas de la matriz  $E[\bar{W}_k \bar{W}_s^T]$ :

- $E[w_k w_s^T] = Q_k \delta_{k-s}$ , donde  $\delta_{k-s}$  denota la función delta de Kronecker, lo cual es inmediato por ser  $\{w_k; k \geq 0\}$  un ruido blanco.
- Teniendo en cuenta la expresión de  $\Phi_k$ , y usando de nuevo que  $\{w_k; k \geq 0\}$  es blanco, se tiene:

$$\begin{aligned} E[w_k \Phi_s^T] &= E[w_k [(F_s \bar{x}_s)^T \otimes w_s^T]] (I_{n^2} + K_{n^2}) + E[w_k w_s^{[2]T}] \\ &= E[w_k [(F_s \bar{x}_s)^T \otimes w_s^T]] (I_{n^2} + K_{n^2}) + Q_k^{(3)} \delta_{k-s}. \end{aligned}$$

Entonces, basta probar que  $E[w_k [(F_s \bar{x}_s)^T \otimes w_s^T]] = 0$  para cualesquiera valores de  $k$  y  $s$ , lo que comprobamos a continuación, según la relación entre tales índices:

- Si  $s < k$ , entonces  $w_k$  es independiente de  $(\bar{x}_s, w_s)$ , y teniendo en cuenta que  $E[w_k] = 0$ , se tiene:

$$E[w_k [(F_s \bar{x}_s)^T \otimes w_s^T]] = E[w_k] E[(F_s \bar{x}_s)^T \otimes w_s^T] = 0.$$



- Si  $s > k$ , entonces  $w_s$  es independiente de  $(\bar{x}_s, w_k)$  y, claramente:

$$E [w_k [(F_s \bar{x}_s)^T \otimes w_s^T]] = E [w_k (F_s \bar{x}_s)^T] \otimes E [w_s^T] = 0.$$

- Por último, si  $s = k$ ,  $w_k$  es independiente de  $\bar{x}_k$  y, por tanto:

$$E [w_k [(F_k \bar{x}_k)^T \otimes w_k^T]] = E [(F_k \bar{x}_k)^T] \otimes E [w_k w_k^T] = 0.$$

- Calculamos finalmente la caja (2, 2) de la matriz  $\bar{Q}_k$ , definida mediante la expresión  $E[\Phi_k \Phi_s^T] - E[\Phi_k]E[\Phi_s^T]$ .

Desarrollando cada una de estas esperanzas, y teniendo en cuenta la relación entre  $k$  y  $s$ , se tiene:

- Si  $s \neq k$  y, por ejemplo,  $s < k$ , entonces  $w_k$  es independiente de  $(\bar{x}_s, w_s)$ , y teniendo en cuenta que  $E[w_k] = 0$  y que  $\bar{x}_s$  es independiente de  $w_s$ , se obtiene:

$$\begin{aligned} E [\Phi_k \Phi_s^T] &= (I_{n^2} + K_{n^2}) E[w_k] E [(F_k \bar{x}_k \bar{x}_s^T F_s^T) \otimes w_s^T] (I_{n^2} + K_{n^2}) \\ &\quad + E[w_k^{[2]}] E [w_s^{[2]T}] + (I_{n^2} + K_{n^2}) [E [F_k \bar{x}_k w_s^{[2]T}] \otimes E[w_k]] \\ &\quad + E [w_k^{[2]}] E [(F_s \bar{x}_s)^T \otimes w_s^T] (I_{n^2} + K_{n^2}) \\ &= E [w_k^{[2]}] E [w_s^{[2]T}]. \end{aligned}$$

- Si  $s = k$ , entonces  $\bar{x}_k$  y  $w_k$  son independientes y, por tanto:

$$\begin{aligned} E [\Phi_k \Phi_k^T] &= (I_{n^2} + K_{n^2}) [(F_k E [\bar{x}_k \bar{x}_k^T] F_k^T) \otimes E [w_k w_k^T]] (I_{n^2} + K_{n^2}) \\ &\quad + E [w_k^{[2]} w_k^{[2]T}]. \end{aligned}$$

Así pues, ya que  $E[\Phi_k]E[\Phi_s^T] = E[w_k^{[2]}]E[w_s^{[2]T}]$ , es evidente que el valor de  $E[\Phi_k \Phi_s^T] - E[\Phi_k]E[\Phi_s^T] = 0$  para todo  $s \neq k$ , mientras que para  $s = k$  obtenemos:

$$E [\Phi_k \Phi_k^T] - E[\Phi_k]E [\Phi_k^T] = (I_{n^2} + K_{n^2}) [(F_k P_k F_k^T) \otimes Q_k] (I_{n^2} + K_{n^2}) + Q_k^{(4)},$$

donde  $P_k = E[\bar{x}_k \bar{x}_k^T]$ . La fórmula recursiva para la obtención de  $P_k$ ,  $k \geq 1$  se deduce de forma inmediata de la ecuación de la señal  $\bar{x}_k$ .

**Proposición 2.2.** *El proceso  $\{\bar{V}_k; k \geq 1\}$  es blanco, centrado, y con matriz de covarianzas dada por*

$$\bar{R}_k = \begin{pmatrix} R_k & R_k^{(3)} \\ R_k^{(3)T} & \bar{R}_k^{22} \end{pmatrix}$$

siendo:

$$\bar{R}_k^{22} = (I_{m^2} + K_{m^2}) [(H_k P_k H_k^T) \otimes R_k] (I_{m^2} + K_{m^2}) + R_k^{(4)}.$$

**Demostración.** La demostración es totalmente análoga a la de la Proposición 1. ■

**Proposición 2.3.** *Los ruidos  $\{\bar{W}_k; k \geq 0\}$  y  $\{\bar{V}_k; k \geq 1\}$  son incorrelados e incorrelados con el estado inicial  $\bar{X}_0$ .*

**Demostración.** En virtud de la hipótesis (v) impuesta sobre el sistema original (2.1) se prueba fácilmente que la señal inicial,  $\bar{X}_0$ , es incorrelada tanto con  $\{\bar{W}_k; k \geq 0\}$  como con  $\{\bar{V}_k; k \geq 1\}$ .

A continuación comprobamos la incorrelación de los ruidos  $\{\bar{W}_k; k \geq 0\}$  y  $\{\bar{V}_k; k \geq 1\}$ ; esto es:

$$Cov(\bar{W}_k, \bar{V}_s^T) = \begin{pmatrix} E[w_k v_s^T] & E[w_k \Psi_s^T] \\ E[\Phi_k v_s^T] & E[\Phi_k \Psi_s^T] - E[\Phi_k]E[\Psi_s^T] \end{pmatrix} = 0.$$

- $E[w_k v_s^T] = 0$ ; inmediato por ser  $\{w_k; k \geq 0\}$  y  $\{v_k; k \geq 1\}$  independientes y centrados.
- Teniendo en cuenta que el proceso señal  $\{\bar{x}_k; k \geq 0\}$  sólo depende del proceso ruido  $\{w_k; k \geq 0\}$ , que éste es independiente de  $\{v_k; k \geq 1\}$ , y que ambos ruidos son

centrados, se tiene que

$$\begin{aligned} E [w_k \Psi_s^T] &= E [w_k [(H_s \bar{x}_s)^T \otimes v_s^T]] (I_{m^2} + K_{m^2}) + E[w_k] E [v_s^{[2]T}] \\ &= [E [(w_k H_s \bar{x}_s)^T] \otimes E [v_s^T]] (I_{m^2} + K_{m^2}) = 0. \end{aligned}$$

- Puesto que  $\Phi_k$  sólo depende de  $(\bar{x}_k, w_k)$ , y estos vectores son independientes de  $v_s$ , se tiene que  $E [\Phi_k v_s^T] = 0$ .
- Por último, teniendo en cuenta las propiedades del sistema, obtenemos:

$$\begin{aligned} E [\Phi_k \Psi_s^T] &= (I_{n^2} + K_{n^2}) E [(F_k \bar{x}_k \bar{x}_s^T H_s^T) \otimes (w_k v_s^T)] (I_{m^2} + K_{m^2}) \\ &\quad + E [w_k^{[2]} v_s^{[2]T}] + (I_{n^2} + K_{n^2}) E [(F_k \bar{x}_k) \otimes (w_k v_s^{[2]T})] \\ &\quad + E \left[ \left[ w_k^{[2]} (H_s \bar{x}_s)^T \right] \otimes v_s^T \right] (I_{m^2} + K_{m^2}) \\ &= E [w_k^{[2]}] E [v_s^{[2]T}] = E[\Phi_k] E[\Psi_s], \end{aligned}$$

con lo que queda probada la incorrelación entre los ruidos del sistema aumentado,  $\{\bar{W}_k; k \geq 0\}$  y  $\{\bar{V}_k; k \geq 1\}$ . ■

### 2.4.3. Algoritmo de filtrado en el sistema aumentado

Una vez probado que los ruidos del sistema aumentado (2.4) son blancos, incorrelados, e incorrelados con la señal en el instante inicial, el estimador lineal de menor error cuadrático medio de la señal  $\bar{X}_k$ , basado en el conjunto de observaciones  $\{\bar{Y}_1, \dots, \bar{Y}_k\}$ , se obtiene aplicando el filtro de Kalman a dicho sistema (ver Apéndice del Capítulo 1). Por lo tanto, el filtro lineal de menor error cuadrático medio,  $\hat{X}_{k/k}$ , el predictor en una etapa,  $\hat{X}_{k/k-1}$ , y las matrices de covarianzas de los errores de filtrado y predicción,  $\bar{P}_{k/k}$  y  $\bar{P}_{k/k-1}$

son obtenidos recursivamente mediante el siguiente algoritmo:

$$\begin{aligned}\hat{X}_{k/k} &= \hat{X}_{k/k-1} + \bar{K}_k \left[ \bar{Y}_k - \bar{H}_k \hat{X}_{k/k-1} \right], \quad k \geq 1 \\ \hat{X}_{k/k-1} &= \bar{F}_{k-1} \hat{X}_{k-1/k-1}, \quad k \geq 1 \\ \bar{K}_k &= \bar{P}_{k/k-1} \bar{H}_k^T \left[ \bar{H}_k \bar{P}_{k/k-1} \bar{H}_k^T + \bar{R}_k \right]^{-1}, \quad k \geq 1 \\ \bar{P}_{k/k} &= \left[ I_{n+n^2} - \bar{K}_k \bar{H}_k \right] \bar{P}_{k/k-1}, \quad k \geq 1 \\ \bar{P}_{k/k-1} &= \bar{F}_{k-1} \bar{P}_{k-1/k-1} \bar{F}_{k-1}^T + \bar{Q}_{k-1}, \quad k \geq 1,\end{aligned}$$

cuyas condiciones iniciales son

$$\hat{X}_{0/0} = 0; \quad \bar{P}_{0/0} = \begin{pmatrix} P_0 & P_0^{(3)} \\ P_0^{(3)T} & P_0^{(4)} \end{pmatrix}.$$

Como se ha indicado anteriormente, el estimador cuadrático de menor error cuadrático medio de la señal original,  $x_k$ , se obtiene a partir del vector constituido por las primeras  $n$  componentes del estimador lineal de menor error cuadrático medio de la señal aumentada,  $\hat{X}_{k/k}$ : ya que la señal original,  $x_k$ , había sido centrada previamente, el estimador cuadrático de  $x_k$  se obtiene sumando la media de  $x_k$  al vector constituido por las primeras  $n$  componentes de  $\hat{X}_{k/k}$ ; dicha media, como también hemos indicado previamente, se obtiene de forma recursiva mediante la expresión

$$E[x_k] = F_{k-1} E[x_{k-1}] + u_{k-1}, \quad k \geq 1.$$

## 2.5. Filtro extendido cuadrático iterado

Como ya se ha comentado en la Sección 1.3, el filtro de Kalman extendido iterado es un algoritmo de estimación de señales en sistemas no lineales con el que se intenta mejorar las estimaciones proporcionadas por el filtro de Kalman extendido, basándose en sucesivos desarrollos en serie de Taylor de la función que define la ecuación de observación del sistema.

El objetivo de esta sección es desarrollar un algoritmo de filtrado, con la misma metodología del filtro de Kalman extendido iterado, para mejorar las estimaciones proporcionadas por el algoritmo de filtrado extendido cuadrático obtenido en la sección anterior.

Concretamente, considerando el sistema no lineal (2.1),

$$\begin{aligned}x_k &= f_{k-1}(x_{k-1}) + w_{k-1}, \quad k \geq 1 \\y_k &= h_k(x_k) + v_k, \quad k \geq 1,\end{aligned}$$

y aplicando el algoritmo de filtrado dado en la Sección 2.4.3 al sistema aumentado, se obtiene el predictor y el filtro de la señal aumentada,  $\bar{X}_k$ , y sus correspondientes matrices de covarianzas del error:

$$\begin{aligned}\hat{X}_{k/k} &= \hat{X}_{k/k-1} + \bar{K}_k \left[ \bar{Y}_k - \bar{H}_k \hat{X}_{k/k-1} \right], \quad k \geq 1 \\ \hat{X}_{k/k-1} &= \bar{F}_{k-1} \hat{X}_{k-1/k-1}, \quad k \geq 1 \\ \bar{K}_k &= \bar{P}_{k/k-1} \bar{H}_k^T \left[ \bar{H}_k \bar{P}_{k/k-1} \bar{H}_k^T + \bar{R}_k \right]^{-1}, \quad k \geq 1 \\ \bar{P}_{k/k} &= \left[ I_{n+n^2} - \bar{K}_k \bar{H}_k \right] \bar{P}_{k/k-1}, \quad k \geq 1 \\ \bar{P}_{k/k-1} &= \bar{F}_{k-1} \bar{P}_{k-1/k-1} \bar{F}_{k-1}^T + \bar{Q}_{k-1}, \quad k \geq 1.\end{aligned}$$

Así, el filtro de la señal,  $\hat{x}_{k/k}^Q$ , lo obtenemos sumando la media de  $x_k$  al vector constituido por las  $n$  primeras componentes de  $\hat{X}_{k/k}$ . Una vez obtenido este filtro se desarrolla de nuevo la función  $h_k$  en serie de Taylor alrededor de él y se vuelven a realizar los cálculos para conseguir una nueva estimación. Este proceso se aplica iterativamente; por tanto, notando

$$\begin{aligned}\hat{X}_{k/k}^0 &= \hat{X}_{k/k}, \quad k \geq 1 \\ \bar{P}_{k/k}^0 &= \bar{P}_{k/k}, \quad k \geq 1,\end{aligned}$$

el bucle de iteraciones para el cálculo del filtro de la señal aumentada y su correspondiente

matriz de covarianzas del error sería el siguiente:

$$\begin{aligned}
H_k^i &= \frac{\partial h_k(x)}{\partial x} \Big|_{x=\hat{x}_{k/k}^{Q^i}}, \quad k \geq 1 \\
\bar{H}_k^i &= \begin{pmatrix} H_k^i & 0 \\ 0 & H_k^{i[2]} \end{pmatrix}, \quad k \geq 1 \\
\bar{R}_k^i &= \begin{pmatrix} R_k & R_k^{(3)} \\ R_k^{(3)} & \bar{R}_{ki}^{22} \end{pmatrix}, \quad k \geq 1 \\
\bar{R}_{ki}^{22} &= (I_{m^2} + K_{m^2}) [(H_k^i P_k H_k^i) \otimes R_k] (I_{m^2} + K_{m^2}) + R_k^{(4)}, \quad k \geq 1 \\
\bar{K}_k^i &= \bar{P}_{k/k-1} \bar{H}_k^{iT} [\bar{H}_k^i \bar{P}_{k/k-1} \bar{H}_k^{iT} + \bar{R}_k^i]^{-1}, \quad k \geq 1 \\
\hat{X}_{k/k}^{i+1} &= \hat{X}_{k/k-1} + \bar{K}_k^i [\bar{Y}_k - \bar{H}_k^i \hat{X}_{k/k-1}], \quad k \geq 1 \\
\bar{P}_{k/k}^{i+1} &= [I_{n+n^2} - \bar{K}_k^i \bar{H}_k^i] \bar{P}_{k/k-1}, \quad k \geq 1,
\end{aligned}$$

donde  $i = 1, \dots, N$ , siendo  $N$  el número de iteraciones.

Una vez finalizado el ciclo de iteraciones, y obtenido el filtro,  $\hat{X}_{k/k}^{N+1}$ , y la matriz de covarianzas del error,  $\bar{P}_{k/k}^{N+1}$ , el predictor y la matriz de covarianzas del error de predicción se obtienen, como en el filtro de Kalman, a partir de las siguientes expresiones:

$$\begin{aligned}
\hat{X}_{k+1/k} &= \bar{F}_k \hat{X}_{k/k}^{N+1}, \quad k \geq 1, \\
\bar{P}_{k+1/k} &= \bar{F}_k \bar{P}_{k/k}^{N+1} \bar{F}_k^T + \bar{Q}_k, \quad k \geq 1.
\end{aligned}$$

## 2.6. Filtro extendido cuadrático de segundo orden

En la Sección 1.4 hemos descrito el filtro de Kalman extendido de segundo orden, un algoritmo de estimación de señales en sistemas no lineales que se aplica con el propósito de reducir el error cometido en la aproximación lineal de las ecuaciones del sistema que se realiza en el filtro de Kalman extendido; el filtro de segundo orden aproxima las funciones no lineales que definen las ecuaciones del sistema incluyendo los términos de segundo orden de sus desarrollos en serie de Taylor, obteniendo así una mejor aproximación del

sistema, que debe reflejarse en un mejor comportamiento de los estimadores obtenidos.

Con esta idea, y centrados en el problema de estimación cuadrática, nos planteamos ahora obtener un algoritmo de filtrado cuadrático de segundo orden que proporcione estimadores con menor error cuadrático medio que los obtenidos mediante el algoritmo presentado en la Sección 2.4.3.

La deducción del algoritmo es totalmente similar a la del filtro extendido cuadrático de la Sección 2.4, una vez aproximado el sistema original por el obtenido desarrollando hasta los términos de segundo orden las funciones no lineales que lo describen.

De manera esquemática, considerando el sistema no lineal (2.1),

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1}, \quad k \geq 1$$

$$y_k = h_k(x_k) + v_k, \quad k \geq 1,$$

y desarrollando en serie de Taylor hasta el segundo orden las funciones  $f_{k-1}$  (en el filtro  $\hat{x}_{k-1/k-1}^Q$ ) y  $h_k$  (en el predictor  $\hat{x}_{k/k-1}^Q$ ), se obtiene la siguiente aproximación lineal:

$$x_k = F_{k-1}x_{k-1} + u_{k-1} + w_{k-1}, \quad k \geq 1$$

$$y_k = H_k x_k + z_k + v_k, \quad k \geq 1,$$

donde

$$F_{k-1} = \left. \frac{\partial f_{k-1}(x)}{\partial x} \right|_{x=\hat{x}_{k-1/k-1}^Q} \quad H_k = \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k-1}^Q}$$

$$u_{k-1} = f_{k-1}(\hat{x}_{k-1/k-1}^Q) - F_{k-1}\hat{x}_{k-1/k-1}^Q + \frac{1}{2} \sum_{i=1}^n \phi_i \text{Tr} \left[ \left. \frac{\partial^2 f_{k-1}^i(x)}{\partial x^2} \right|_{x=\hat{x}_{k-1/k-1}^Q} P_{k-1/k-1} \right]$$

$$z_k = h_k(\hat{x}_{k/k-1}^Q) - H_k \hat{x}_{k/k-1}^Q + \frac{1}{2} \sum_{i=1}^m \phi_i \text{Tr} \left[ \left. \frac{\partial^2 h_k^i(x)}{\partial x^2} \right|_{x=\hat{x}_{k/k-1}^Q} P_{k/k-1} \right],$$

siendo  $f_{k-1}^i$  y  $h_k^i$  las componentes  $i$ -ésimas de  $f_{k-1}$  y  $h_k$ , respectivamente, y  $\phi_i$  el vector  $n$ -dimensional cuyos elementos son cero excepto el de la posición  $i$ , que es uno.

Como en la Sección 2.4, centramos los vectores señal y observación de este sistema aproximado, con objeto de simplificar los desarrollos; por tanto, notando  $\bar{x}_k = x_k - E[x_k]$  e  $\bar{y}_k = y_k - E[y_k]$ , se tiene:

$$\bar{x}_k = F_{k-1}\bar{x}_{k-1} + w_{k-1}, \quad k \geq 1$$

$$\bar{y}_k = H_k\bar{x}_k + v_k, \quad k \geq 1,$$

donde, teniendo en cuenta las ecuaciones del sistema linealizado,

$$E[x_k] = F_{k-1}E[x_{k-1}] + u_{k-1}, \quad k \geq 1$$

$$E[y_k] = H_kE[x_k] + z_k, \quad k \geq 1.$$

Este sistema lineal centrado tiene la misma estructura y propiedades que el obtenido en la Sección 2.4.1; la diferencia entre ambos sólo radica en las constantes  $u_k$  y  $z_k$ , que únicamente intervienen en las ecuaciones para el cálculo de las medias de  $x_k$  e  $y_k$ , respectivamente. Por consiguiente, el sistema aumentado es el mismo que el obtenido en la Sección 2.4.2, satisface las mismas propiedades, y los estimadores de la señal se obtienen mediante el algoritmo establecido en la Sección 2.4.3.

## 2.7. Ejemplo numérico

Con el fin de demostrar la efectividad de los estimadores cuadráticos propuestos en este capítulo, se han implementado en Matlab dos programas. El primero de ellos, compara los algoritmos presentados en el Capítulo 1 con el filtro extendido cuadrático propuesto en la Sección 2.4, mediante su aplicación a un modelo no lineal definido por las mismas funciones que el considerado en la Sección 1.6. En el segundo programa, se han implementado los distintos algoritmos cuadráticos propuestos en este capítulo, aplicándose, para su comparación, al mismo modelo no lineal.

Cada uno de los programas realizados simulan, en cada iteración, la señal que se desea estimar y la observación correspondiente, proporcionando las distintas estimaciones



lineales y cuadráticas, así como los errores cuadráticos medios con los que se compara la bondad de las estimaciones. En todos los casos se han realizado 1000 simulaciones y 50 iteraciones de cada algoritmo para cada simulación.

El modelo considerado es el siguiente:

$$\begin{aligned}x_k &= \frac{1}{x_{k-1}^2 + 3} + w_{k-1}, & k \geq 1 \\y_k &= x_k^2 + \exp(x_k) + v_k, & k \geq 1.\end{aligned}$$

Para aplicar los algoritmos de estimación cuadrática propuestos en este capítulo, es necesario conocer los momentos hasta el cuarto orden, tanto de la señal en el instante inicial, como de los ruidos aditivos que intervienen en el sistema. Así, se suponen las siguientes hipótesis:

*i)* La señal inicial,  $x_0$ , es una variable aleatoria gaussiana con media cero y varianza uno,  $P_0 = 1$ ; por tanto, los momentos de tercer y cuarto orden están dados por  $P_0^{(3)} = 0$  y  $P_0^{(4)} = 3 - P_0^2$ , respectivamente.

*ii)*  $\{w_k; k \geq 0\}$  es un ruido blanco con distribución

$$P[w_k = 1] = \frac{15}{18} \quad P[w_k = -3] = \frac{2}{18} \quad P[w_k = -9] = \frac{1}{18}.$$

Por tanto, es un proceso centrado, con varianzas  $Q_k = \frac{19}{3}$ , y momentos de tercer y cuarto orden dados por  $Q_k^{(3)} = -\frac{128}{3}$  y  $Q_k^{(4)} = \frac{1123}{3} - Q_k^2$ , respectivamente.

*iii)*  $\{v_k; k \geq 1\}$  es un ruido blanco con distribución

$$P[v_k = 1] = \frac{15}{18} \quad P[v_k = -3] = \frac{2}{18} \quad P[v_k = -9] = \frac{1}{18},$$

y, por tanto, centrado, con varianzas  $R_k = \frac{19}{3}$ , y momentos de tercer y cuarto orden dados por  $R_k^{(3)} = -\frac{128}{3}$  y  $R_k^{(4)} = \frac{1123}{3} - R_k^2$ , respectivamente.

*iv)* La señal inicial,  $x_0$ , y los ruidos  $\{w_k; k \geq 0\}$  y  $\{v_k; k \geq 1\}$  son mutuamente independientes.

Para medir la bondad de las estimaciones proporcionadas por cada algoritmo, en cada instante de tiempo se utiliza el error cuadrático medio correspondiente a las diferentes simulaciones, dado por:

$$ECM_k = \frac{1}{1000} \sum_{s=1}^{1000} (x_k^s - \hat{x}_{k/k}^s)^2, \quad k = 1, \dots, 50,$$

donde,  $x_k^s$  denota el valor de la señal generado en la simulación  $s$  y la iteración  $k$ , y  $\hat{x}_{k/k}^s$  denota el filtro calculado con el correspondiente algoritmo en dicha simulación e iteración.

En la Figura 2.1, se muestran los errores cuadráticos medios de los filtros propuestos en el Capítulo 1 y del filtro extendido cuadrático.

Como puede comprobarse, el comportamiento de los filtros extendidos lineales y del filtro de Kalman ensemble es similar al obtenido en el ejemplo del Capítulo 1, en el que se consideraron ruidos aditivos gaussianos: los filtros extendidos iterado y de segundo orden presentan menores errores cuadráticos medios que el filtro de Kalman extendido, siendo, también en este caso, el filtro iterado el que produce una mejora mayor. Por su parte, el filtro de Kalman ensemble, proporciona mejores estimaciones que todos los anteriores.

Por otra parte, podemos observar que, en cada iteración, el error cuadrático medio asociado al filtro extendido cuadrático no sólo es menor que el asociado al filtro de Kalman extendido, sino también que el asociado a cada uno de los filtros considerados en el Capítulo 1, lo que demuestra, como era de esperar, la efectividad de la estimación a partir de funciones polinomiales de segundo grado de las observaciones.

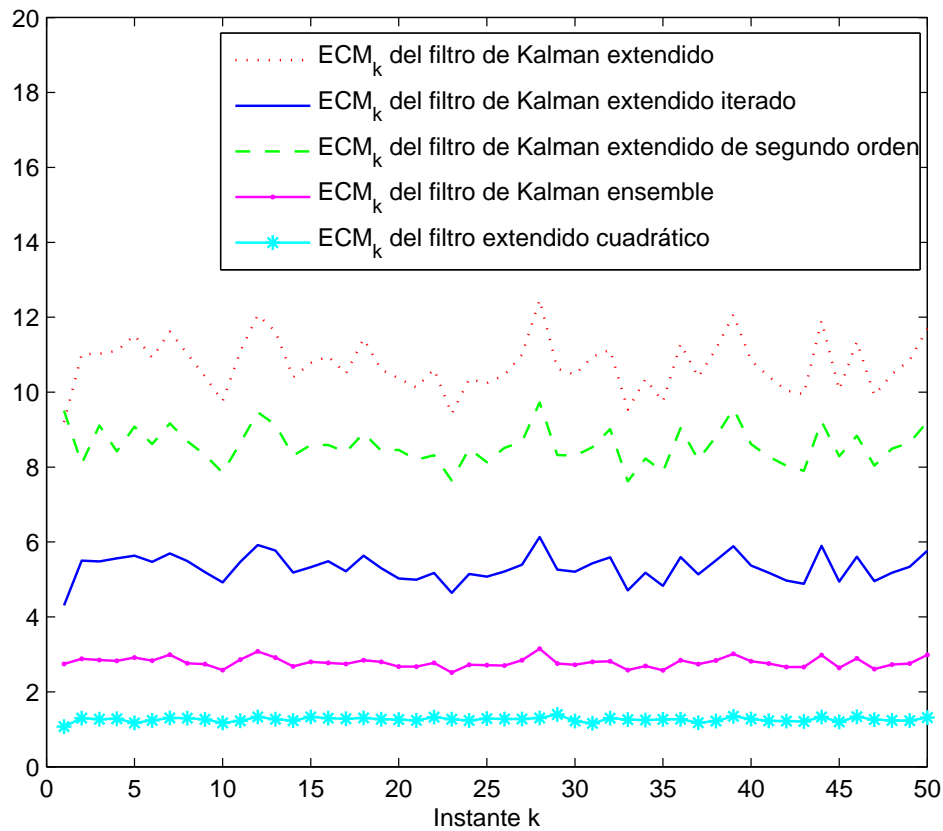


Figura 2.1: Error cuadrático medio del filtro de Kalman extendido, filtro de Kalman extendido iterado, filtro de Kalman extendido de segundo orden, filtro de Kalman ensemble y filtro extendido cuadrático.

Una vez comprobada la mejora en la estimación de la señal que obtenemos al considerar el estimador extendido cuadrático frente a los extendidos lineales y al Ensemble, nos centramos ahora en la comparación de los filtros extendidos cuadráticos propuestos en este capítulo.

En la Figura 2.2 se muestran los errores cuadráticos medios de las estimaciones obtenidas mediante la aplicación de estos algoritmos. En esta figura se aprecia una leve mejora en el comportamiento del filtro extendido cuadrático de segundo orden frente al extendido cuadrático, pero, sin embargo, un evidente mejor comportamiento del filtro extendido cuadrático iterado frente a los dos anteriores.

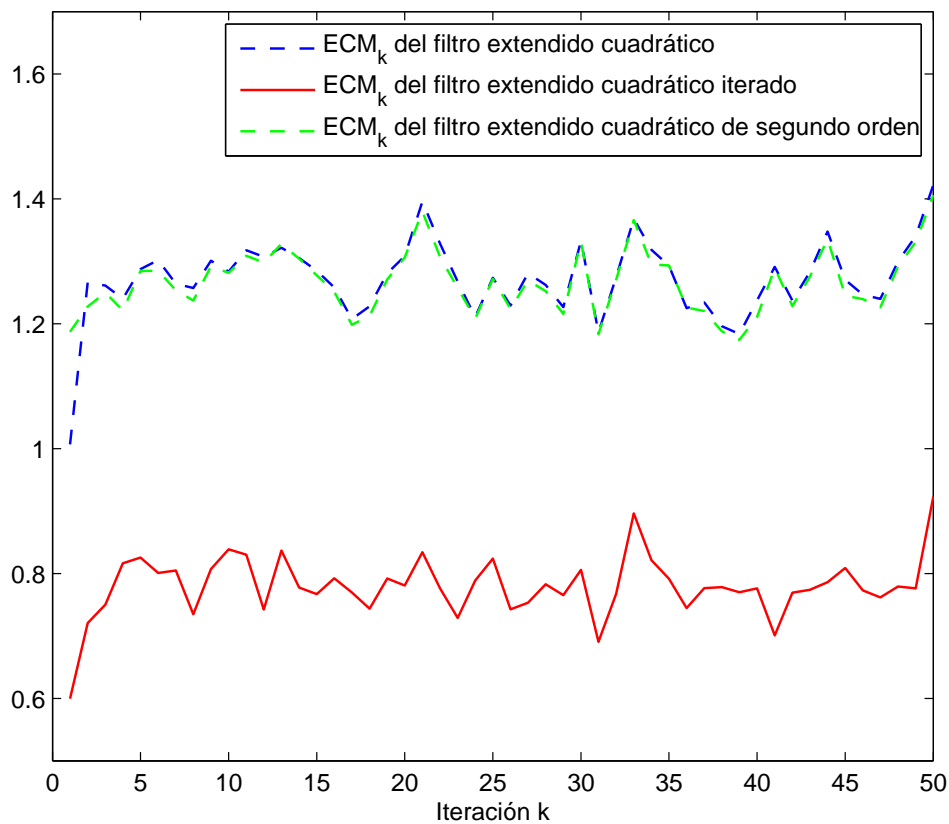


Figura 2.2: Error cuadrático medio del filtro del extendido cuadrático, filtro extendido cuadrático iterado y filtro extendido cuadrático de segundo orden.



# Capítulo 3

## Estimación cuadrática en sistemas no lineales con observaciones inciertas

### 3.1. Introducción

En el Capítulo 2 se ha abordado el problema de estimación cuadrática en sistemas no lineales suponiendo que la señal que se desea estimar está siempre presente en las observaciones disponibles; sin embargo, existen numerosas situaciones prácticas en las que posibles interrupciones en el mecanismo de medida, o diversos tipos de fallo en la transmisión de la señal, pueden ocasionar la pérdida de la misma, de manera que la observación procesada en la estimación puede consistir exclusivamente en ruido; ésta es una situación que ocurre frecuentemente en sistemas de comunicación, rastreo de trayectorias o procesamiento de imágenes.

El primero que trató el procesamiento de señales en situaciones de este tipo fué Nahi [42], en 1969, quien abordó el problema de estimación en sistemas lineales modelizando la incertidumbre sobre la presencia o no de señal en las observaciones mediante un proceso estocástico definido por variables de Bernoulli. Considerando sistemas lineales con ruidos aditivos blancos e independientes, y bajo el supuesto de independencia de las va-

riables que modelizan la incertidumbre, Nahi obtuvo un algoritmo recursivo de filtrado lineal de menor error cuadrático medio con una estructura similar al filtro de Kalman, pero dependiendo de las probabilidades de que las observaciones consistan sólo en ruido, denominadas *probabilidades de falsa alarma*.

Hay que indicar que la modelización de la incertidumbre mediante un proceso aleatorio hace que este tipo de sistemas, denominados *sistemas con observaciones inciertas*, sean siempre no gaussianos, incluso bajo hipótesis de gaussianidad e independencia de los ruidos aditivos y la señal inicial  $y$ , por tanto, a diferencia del filtro de Kalman, el de Nahi sólo proporciona el estimador óptimo en la clase de estimadores lineales. Por este motivo, el trabajo pionero de Nahi fue el primero de una extensa literatura desarrollada con el fin de mejorar dicho filtro (Monzingo [40], [41], Tugnait [54], Hadidi y Schwartz [22], Jaffer y Gupta [28], Wang [56], Chow y Birkemeier [9], [10], Hermoso y Linares [23], [24]; entre otros).

Sin embargo, a pesar de su manifiesta importancia para contemplar la posibilidad de que las observaciones consistan exclusivamente en ruido, el volumen de aportaciones al problema de estimación a partir de observaciones inciertas en sistema no lineales es significativamente menor que en el caso de sistemas lineales. Algunas aportaciones a este estudio pueden verse en los trabajos de Nanakara y Yaz [43], Hermoso y Linares [25] y Nakamori *et al.* [47], Hermoso *et al.* [27] que, usando diferentes aproximaciones, presentan distintos algoritmos recursivos de estimación a partir de observaciones inciertas en sistema no lineales.

En el trabajo de Hermoso y Linares [25] se propone, entre otros, un algoritmo de estimación a partir de la linealización del sistema, que generaliza el filtro de Kalman extendido al caso de observaciones inciertas; este algoritmo está basado en la forma mixtura

de los estadísticos de primer y segundo orden de las observaciones, y se denomina *filtro linealizado mixturas*.

El objetivo en este capítulo es avanzar en el estudio del problema de estimación en sistemas no lineales con observaciones inciertas, enfocando este estudio en las dos direcciones siguientes.

Por una parte, nos proponemos obtener un algoritmo de estimación cuadrática para generalizar el filtro extendido cuadrático propuesto en el capítulo anterior al caso de incertidumbre sobre la presencia de señal en las observaciones; debemos indicar que las primeras aproximaciones a este estudio fueron presentadas en [27].

Por otra parte, nos planteamos también la extensión de la metodología de filtrado ensemble, basada en simulaciones de la señal inicial y los ruidos del sistema, como se expuso en la Sección 1.5 para sistemas con certidumbre, al caso de sistemas con observaciones inciertas; en tales sistemas, como hemos indicado, además de los ruidos aditivos, existe una perturbación aleatoria en las observaciones para indicar la presencia o ausencia de señal, y este nuevo ruido debe ser también simulado para llevar a cabo el proceso de actualización de los predictores en cada iteración del algoritmo.

El capítulo se estructura en siete secciones. Comenzamos, en la Sección 3.2, describiendo las ecuaciones del sistema objeto de estudio, y especificando las hipótesis sobre los procesos que intervienen en el mismo, necesarias para abordar el problema de estimación cuadrática. Con objeto de aplicar la misma metodología que en el capítulo anterior para el estudio de dicho problema, en la Sección 3.3 obtendremos el sistema aumentado correspondiente al sistema linealizado asociado al original, y estudiaremos sus propiedades. En la Sección 3.4 deduciremos un algoritmo de filtrado lineal para el sistema aumentado, que proporcionará el *filtro extendido cuadrático*, primer objetivo de este capítulo.



Las secciones 3.5 y 3.6 están dedicadas a la aplicación de la metodología de filtrado ensemble a sistemas con observaciones inciertas; en la Sección 3.5 describimos las etapas para llevar a cabo el proceso de estimación, y en la Sección 3.6 se aplica este procedimiento para la obtención de un *filtro ensemble cuadrático*.

Finalizamos el capítulo presentando un ejemplo de simulación, en el que se muestra la aplicación de los distintos algoritmos propuestos en este capítulo, la comparación entre los mismos y, también, la comparación de los filtros propuestos con el filtro linealizado mixturas de [25]. En el mismo sentido que en los ejemplos considerados en los capítulos anteriores, la eficacia de los algoritmos se compara en términos de los errores cuadráticos medios de las estimaciones obtenidas en distintas simulaciones de los mismos.

## 3.2. Descripción del sistema

Como ya hemos indicado, en este capítulo nos ocupamos de estudiar el problema de estimación en sistemas estocásticos no lineales bajo el supuesto de que ciertos fallos aleatorios en el mecanismo de transmisión de las observaciones han podido producir pérdida de la información sobre la señal, de manera que algunas de las observaciones pueden consistir sólo en ruido. Obviamente, la incertidumbre sobre la presencia o no de señal en cada una de las observaciones debe ser modelizada e incluida en el modelo de observación, para reflejar de forma adecuada la situación que se pretende analizar. La modelización de la incertidumbre mediante variables aleatorias binarias que describan la presencia o no de señal en las observaciones ocasiona que la ecuación de observación dependa en estas situaciones de un proceso ruido multiplicativo, además del ruido aditivo.

Concretamente, consideramos una señal estocástica modelizada por una dinámica no lineal,

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1}, \quad k \geq 1,$$

y suponemos que, en cada instante de tiempo  $k \geq 1$ , el mecanismo de observación proporciona funciones no lineales de la misma,  $h_k(x_k)$ , que durante el proceso de transmisión son perturbadas por un ruido aditivo,  $v_k$ , y que, además, pueden llegar a perderse en dicho proceso, de manera que la observación que realmente se usa en la estimación en dicho caso es sólo ruido; esto es, si  $1 - p_k$  denota la probabilidad de que la observación contenga sólo ruido (*probabilidad de falsa alarma*), la observación en dicho instante es

$$y_k = \begin{cases} h_k(x_k) + v_k, & \text{con probabilidad } p_k \\ v_k, & \text{con probabilidad } 1 - p_k, \end{cases}$$

y puede describirse de forma unificada como

$$y_k = \gamma_k h_k(x_k) + v_k,$$

donde  $\gamma_k$  es una variable aleatoria de Bernoulli con  $P(\gamma_k = 1) = p_k$ .

Así, los sistemas con observaciones inciertas, objeto de estudio en este capítulo, difieren de los considerados en el Capítulo 2 en la estructura de la ecuación de observación, que ahora, además del ruido aditivo, incluye un ruido multiplicativo cuyos valores, uno y cero, reflejan la presencia o ausencia de señal en cada observación:

$$\begin{aligned} x_k &= f_{k-1}(x_{k-1}) + w_{k-1}, \quad k \geq 1 \\ y_k &= \gamma_k h_k(x_k) + v_k, \quad k \geq 1, \end{aligned} \tag{3.1}$$

donde  $\{x_k; k \geq 0\}$  es el proceso estocástico  $n$ -dimensional que describe la señal a estimar,  $\{y_k; k \geq 1\}$  el que describe las observaciones usadas para la estimación,  $\{w_k; k \geq 0\}$  y  $\{v_k; k \geq 1\}$  son los ruidos aditivos de las ecuaciones de la señal y de la observación, respectivamente, y  $\{\gamma_k; k \geq 1\}$  es el proceso ruido multiplicativo que describe la presencia

o ausencia de la señal en las observaciones, a lo que nos referiremos como incertidumbre en las observaciones.

Para abordar el primer objetivo de este capítulo, generalizar los resultados obtenidos en el Capítulo 2 sobre estimación cuadrática, debe suponerse el conocimiento de los momentos hasta el cuarto orden de los procesos que intervienen en el sistema (3.1). Concretamente, se suponen las siguientes hipótesis:

- i)* La señal inicial,  $x_0$ , es un vector aleatorio con momentos hasta el cuarto orden conocidos, y se denotan de la siguiente forma:

$$E[x_0] = \mu_0,$$

$$E[(x_0 - \mu_0)(x_0 - \mu_0)^T] = P_0,$$

$$E\left[(x_0 - \mu_0)\left(x_0^{[2]} - E[x_0^{[2]}\right]\right)^T] = P_0^{(3)},$$

$$E\left[\left(x_0^{[2]} - E[x_0^{[2]}\right)\left(x_0^{[2]} - E[x_0^{[2]}\right]\right)^T] = P_0^{(4)}.$$

- ii)*  $\{w_k; k \geq 0\}$  es un ruido blanco con momentos hasta el cuarto orden conocidos:

$$E[w_k] = 0,$$

$$E[w_k w_k^T] = Q_k,$$

$$E\left[w_k\left(w_k^{[2]} - E[w_k^{[2]}\right]\right)^T] = Q_k^{(3)},$$

$$E\left[\left(w_k^{[2]} - E[w_k^{[2]}\right)\left(w_k^{[2]} - E[w_k^{[2]}\right]\right)^T] = Q_k^{(4)}.$$

iii)  $\{v_k; k \geq 1\}$  es un ruido blanco con momentos hasta el cuarto orden conocidos:

$$E[v_k] = 0,$$

$$E[v_k v_k^T] = R_k,$$

$$E\left[v_k \left(v_k^{[2]} - E[v_k^{[2]}]\right)^T\right] = R_k^{(3)},$$

$$E\left[\left(v_k^{[2]} - E[v_k^{[2]}]\right) \left(v_k^{[2]} - E[v_k^{[2]}]\right)^T\right] = R_k^{(4)}.$$

iv)  $\{\gamma_k; k \geq 1\}$  es una sucesión de variables aleatorias de Bernoulli independientes con  $P(\gamma_k = 1) = p_k, \forall k \geq 1$ .

v)  $f_k : \mathbb{R}^n \rightarrow \mathbb{R}^n$  y  $h_k : \mathbb{R}^n \rightarrow \mathbb{R}^m$  son funciones no lineales, continuas y diferenciables.

vi) La señal inicial,  $x_0$ , y los ruidos  $\{w_k; k \geq 0\}$ ,  $\{v_k; k \geq 1\}$  y  $\{\gamma_k; k \geq 1\}$  son mutuamente independientes.

### 3.3. Sistema aumentado

Considerando el sistema no lineal con observaciones inciertas descrito en el apartado anterior, nos planteamos obtener un algoritmo de estimación cuadrática extendiendo la metodología del filtro de Kalman extendido de forma similar a como se ha expuesto en la Sección 2.4. En primer lugar, linealizamos el sistema (subsección 3.3.1) para, seguidamente, construir el sistema aumentado correspondiente al linealizado y estudiar las propiedades (subsección 3.3.2) necesarias para abordar el problema de estimación, que se describirá en la Sección 3.4.

### 3.3.1. Linealización del sistema

Como en el caso de certidumbre en las observaciones, en primer lugar se realiza la aproximación lineal del sistema (3.1), desarrollando en serie de Taylor hasta el primer orden las funciones que lo definen. Así, el sistema original es aproximado por el siguiente, también con observaciones inciertas:

$$\begin{aligned}x_k &= F_{k-1}x_{k-1} + u_{k-1} + w_{k-1}, \quad k \geq 1 \\y_k &= \gamma_k(H_k x_k + z_k) + v_k, \quad k \geq 1,\end{aligned}$$

donde  $F_{k-1}$ ,  $u_{k-1}$ ,  $H_k$  y  $z_k$  están dadas por:

$$\begin{aligned}F_{k-1} &= \left. \frac{\partial f_{k-1}(x)}{\partial x} \right|_{x=\hat{x}_{k-1/k-1}^Q} \\u_{k-1} &= f_{k-1}(\hat{x}_{k-1/k-1}^Q) - F_{k-1}\hat{x}_{k-1/k-1}^Q \\H_k &= \left. \frac{\partial h_k(x)}{\partial x} \right|_{x=\hat{x}_{k/k-1}^Q} \\z_k &= h_k(\hat{x}_{k/k-1}^Q) - H_k\hat{x}_{k/k-1}^Q.\end{aligned}$$

Con el fin de simplificar los cálculos en desarrollos posteriores, centramos los vectores señal y observación, obteniendo

$$\begin{aligned}\bar{x}_k &= F_{k-1}\bar{x}_{k-1} + w_{k-1}, \quad k \geq 1 \\ \bar{y}_k &= \gamma_k H_k \bar{x}_k + \bar{v}_k, \quad k \geq 1,\end{aligned}\tag{3.2}$$

donde  $\bar{x}_k = x_k - E[x_k]$ ,  $\bar{y}_k = y_k - E[y_k]$  y  $\bar{v}_k = (\gamma_k - p_k)(H_k E[x_k] + z_k) + v_k$ .

Debe notarse que, a diferencia de lo que ocurría en el caso de certidumbre, el ruido aditivo de la ecuación de la observación centrada no coincide con el original, sino que en este caso está interferido, además, por el ruido multiplicativo, por la media de la señal original, y por el valor de  $z_k$ .

Sin embargo, este nuevo ruido claramente es centrado, y puede probarse sin dificultad que es también blanco, sin más que tener en cuenta la independencia de  $\{\gamma_k; k \geq 1\}$  y

$\{v_k; k \geq 1\}$ . Además, sus momentos hasta el cuarto orden, necesarios en el estudio del problema de estimación cuadrática, están dados por:

$$\begin{aligned}\bar{R}_k &= E [\bar{v}_k \bar{v}_k^T] = p_k(1 - p_k)c_k c_k^T + R_k, \\ \bar{R}_k^{(3)} &= E \left[ \bar{v}_k \left( \bar{v}_k^{[2]} - E [\bar{v}_k^{[2]}] \right)^T \right] = p_k(1 - p_k)(1 - 2p_k)c_k c_k^{[2]T} + R_k^{(3)}, \\ \bar{R}_k^{(4)} &= E \left[ \left( \bar{v}_k^{[2]} - E [\bar{v}_k^{[2]}] \right) \left( \bar{v}_k^{[2]} - E [\bar{v}_k^{[2]}] \right)^T \right] \\ &= [p_k(1 - p_k)(3p_k^2 - 3p_k + 1) - p_k^2(1 - p_k)^2] c_k^{[2]} c_k^{[2]T} + R_k^{(4)} \\ &\quad + p_k(1 - p_k)(I_{m^2} + K_{m^2}) [(c_k c_k^T) \otimes R_k] (I_{m^2} + K_{m^2}),\end{aligned}$$

donde  $c_k = H_k E[x_k] + z_k$ .

### 3.3.2. Propiedades del sistema aumentado

Para abordar el problema de estimación cuadrática en el sistema (3.2), se consideran los vectores señal y observación aumentados, obtenidos añadiendo a los correspondientes originales sus potencias Kronecker de segundo orden:

$$X_k = \begin{pmatrix} \bar{x}_k \\ \bar{x}_k^{[2]} \end{pmatrix} \in \mathbb{R}^{n+n^2}, \quad Y_k = \begin{pmatrix} \bar{y}_k \\ \bar{y}_k^{[2]} \end{pmatrix} \in \mathbb{R}^{m+m^2}.$$

Entonces, como en la Sección 2.4, el estimador cuadrático de menor error cuadrático medio de  $x_k$  basado en el conjunto de observaciones  $y_1, \dots, y_k$ , se obtendrá a partir del vector constituido por las  $n$  primeras componentes del estimador lineal de menor error cuadrático medio de la señal aumentada,  $X_k$ , basado en las observaciones aumentadas,  $Y_1, \dots, Y_k$ .

Con objeto de deducir un algoritmo recursivo que proporcione dicho estimador en cada instante, comenzamos obteniendo las ecuaciones que describen la evolución de la señal y las observaciones aumentadas, para lo cual calculamos las potencias de segundo grado de los vectores  $\bar{x}_k$  e  $\bar{y}_k$ .

### Potencias de segundo grado de la señal centrada $\bar{x}_k$

La diferencia entre los sistemas con observaciones inciertas y los considerados en el Capítulo 2 radica exclusivamente en la ecuación de observación; por lo tanto, la expresión de las potencias de segundo grado de la señal centrada,  $\bar{x}_k^{[2]}$ , coincide con la calculada en la Sección 2.4.2:

$$\bar{x}_k^{[2]} = F_{k-1}^{[2]} \bar{x}_{k-1}^{[2]} + \Phi_{k-1}, \quad k \geq 1,$$

donde

$$\Phi_k = (I_{n^2} + K_{n^2})[(F_k \bar{x}_k) \otimes w_k] + w_k^{[2]}, \quad k \geq 0.$$

### Potencias de segundo grado de la observación centrada $\bar{y}_k$

Teniendo en cuenta la ecuación de observación del sistema (3.2), se obtiene:

$$\bar{y}_k^{[2]} = \gamma_k H_k^{[2]} \bar{x}_k^{[2]} + (H_k \bar{x}_k) \otimes (\gamma_k \bar{v}_k) + (\gamma_k \bar{v}_k) \otimes (H_k \bar{x}_k) + \bar{v}_k^{[2]},$$

y notando

$$\Psi_k = (I_{m^2} + K_{m^2})[(H_k \bar{x}_k) \otimes (\gamma_k \bar{v}_k)] + \bar{v}_k^{[2]},$$

se tiene:

$$\bar{y}_k^{[2]} = \gamma_k H_k^{[2]} \bar{x}_k^{[2]} + \Psi_k, \quad k \geq 1.$$

Por lo tanto, los vectores señal y observación aumentados están descritos por las siguientes ecuaciones, correspondientes a un sistema lineal con observaciones inciertas, con incertidumbre modelizada por la misma sucesión de variables de Bernoulli del sistema original:

$$X_k = \bar{F}_{k-1} X_{k-1} + W_{k-1}, \quad k \geq 1$$

$$Y_k = \gamma_k \bar{H}_k X_k + V_k, \quad k \geq 1,$$

donde

$$\begin{aligned}\bar{F}_k &= \begin{pmatrix} F_k & 0 \\ 0 & F_k^{[2]} \end{pmatrix}, & \bar{H}_k &= \begin{pmatrix} H_k & 0 \\ 0 & H_k^{[2]} \end{pmatrix}, \\ W_k &= \begin{pmatrix} w_k \\ \Phi_k \end{pmatrix}, & V_k &= \begin{pmatrix} \bar{v}_k \\ \Psi_k \end{pmatrix}.\end{aligned}$$

A continuación, con objeto de simplificar la obtención de los estimadores requeridos, centramos los vectores aumentados, obteniendo las ecuaciones del que denominamos *sistema aumentado*, en el que abordaremos el problema de estimación lineal:

$$\begin{aligned}\bar{X}_k &= \bar{F}_{k-1}\bar{X}_{k-1} + \bar{W}_{k-1}, & k \geq 1, \\ \bar{Y}_k &= \gamma_k \bar{H}_k \bar{X}_k + \bar{V}_k, & k \geq 1,\end{aligned}\tag{3.3}$$

donde

$$\begin{aligned}\bar{X}_k &= X_k - E[X_k], & \bar{Y}_k &= Y_k - E[Y_k] \\ \bar{W}_k &= \begin{pmatrix} w_k \\ \Phi_k - E[\Phi_k] \end{pmatrix}, & \bar{V}_k &= \begin{pmatrix} \bar{v}_k \\ \Psi_k - E[\Psi_k] \end{pmatrix} + (\gamma_k - p_k)\bar{H}_k E[X_k].\end{aligned}$$

Notemos que el proceso ruido de la ecuación de la señal en el sistema aumentado (3.3),  $\{\bar{W}_k; k \geq 0\}$ , coincide con el correspondiente en el sistema aumentado (2.4) y, por tanto, sus propiedades y momentos necesarios para la determinación de los estimadores son los especificados en la Proposición 2.1. A continuación, estudiamos las propiedades del ruido de la observación,  $\{\bar{V}_k; k \geq 1\}$ , que, a diferencia del correspondiente en el sistema (2.4), no es un ruido blanco, como probamos seguidamente.

**Proposición 3.1.** *El proceso  $\{\bar{V}_k; k \geq 1\}$  es centrado, y su función de covarianzas está dada por*

$$\mathcal{R}_{ks} = \begin{pmatrix} \bar{R}_k \delta_{k-s} & \bar{R}_k^{12} \delta_{k-s} \\ \bar{R}_k^{12T} \delta_{k-s} & \bar{R}_{ks}^{22} \end{pmatrix}$$



siendo:

$$\begin{aligned}\bar{R}_k^{12} &= \bar{R}_k^{(3)} + p_k(1-p_k)c_k \text{vec}^T(P_{k,k})H_k^{[2]T} \\ \bar{R}_{k,s}^{22} &= (I_{m^2} + K_{m^2}) \left[ (H_k P_{k,s} H_s^T) \otimes [\alpha_{ks} c_k c_s^T + p_k R_k \delta_{k-s}] \right] (I_{m^2} + K_{m^2}) \\ &\quad + \bar{R}_k^{(4)} \delta_{k-s} + p_k(1-p_k)(1-2p_k) \left[ c_k^{[2]} \text{vec}^T(P_{k,k})H_k^{[2]T} + H_k^{[2]} \text{vec}(P_{k,k})c_k^{[2]T} \right] \delta_{k-s} \\ &\quad + p_k(1-p_k)H_k^{[2]} \text{vec}(P_{k,k}) \text{vec}^T(P_{k,k})H_k^{[2]T} \delta_{k-s},\end{aligned}$$

donde  $\bar{R}_k$ ,  $\bar{R}_k^{(3)}$  y  $\bar{R}_k^{(4)}$  son los momentos del ruido  $\bar{v}_k$ , especificados en la Sección 3.3.1,  $c_k = H_k E[x_k] + z_k$ ,  $P_{k,s} = E[\bar{x}_k \bar{x}_s^T]$  verifica

$$P_{k,s} = F_{k-1} P_{k-1,s-1} F_{s-1}^T + Q_{k-1} \delta_{k-s},$$

y

$$\alpha_{ks} = \begin{cases} p_k p_s (1-p_k)(1-p_s), & s \neq k \\ p_k (1-p_k)^2, & s = k \end{cases}$$

**Demostración.** Es inmediato probar que  $\{\bar{V}_k; k \geq 1\}$  es centrado y, por tanto, nos centramos exclusivamente en el cálculo de su función de covarianzas, para lo que usaremos repetidamente las propiedades del producto de Kronecker y las hipótesis de independencia impuestas sobre el sistema original.

$$\mathcal{R}_{ks} = E[\bar{V}_k \bar{V}_s^T] = \begin{pmatrix} E[\bar{v}_k \bar{v}_s^T] & E[\bar{v}_k \bar{\Psi}_s^T] \\ E[\bar{\Psi}_k \bar{v}_s^T] & E[\bar{\Psi}_k \bar{\Psi}_s^T] \end{pmatrix},$$

donde  $\bar{v}_k = (\gamma_k - p_k)(H_k E[x_k] + z_k) + v_k$  y  $\bar{\Psi}_k = \Psi_k - E[\Psi_k] + (\gamma_k - p_k)H_k^{[2]} E[\bar{x}_k^{[2]}]$ .

- $E[\bar{v}_k \bar{v}_s^T] = \bar{R}_k \delta_{k-s}$ , lo cual es inmediato ya que  $\{\bar{v}_k; k \geq 1\}$  es blanco.
- Teniendo en cuenta la expresión de  $\bar{\Psi}_k$ , puesto que  $E[\bar{v}_k] = 0$ , se tiene que

$$E[\bar{v}_k \bar{\Psi}_s^T] = E[\bar{v}_k \Psi_s^T] + E[\bar{v}_k (\gamma_s - p_s)] E[\bar{x}_s^{[2]}]^T H_s^{[2]T}.$$

Ahora, calculamos el valor de estas esperanzas en función de la relación entre los índices  $k$  y  $s$ :

$$\bullet E [\bar{v}_k \Psi_s^T] = [(E [\bar{x}_s^T] H_s^T) \otimes E [\bar{v}_k \gamma_s \bar{v}_s^T]] (I_{m^2} + K_{m^2}) + E [\bar{v}_k \bar{v}_s^{[2]T}],$$

donde se ha usado que  $\bar{x}_k$  es independiente de  $(\gamma_k, \bar{v}_k)$ ; por tanto, puesto que  $E[\bar{x}_k] = 0$ , se tiene que

$$E [\bar{v}_k \Psi_s^T] = E [\bar{v}_k \bar{v}_s^{[2]T}] = \bar{R}_k^{(3)} \delta_{k-s}.$$

- Ya que  $\{\gamma_k; k \geq 1\}$  es una sucesión de variables aleatorias de Bernoulli independientes e independientes de  $\{v_k; k \geq 1\}$ , se tiene

$$E[\bar{v}_k(\gamma_s - p_s)] = E[(\gamma_k - p_k)(\gamma_s - p_s)]c_k + E[v_k]E[\gamma_s - p_s] = p_k(1 - p_k)c_k \delta_{k-s},$$

y queda probada la expresión de  $E [\bar{v}_k \bar{\Psi}_s^T]$ .

- Calculamos ahora  $E [\bar{\Psi}_k \bar{\Psi}_s^T]$ :

$$\begin{aligned} E [\bar{\Psi}_k \bar{\Psi}_s^T] &= E [\Psi_k \Psi_s^T] - E[\Psi_k]E [\Psi_s^T] \\ &\quad + H_k^{[2]} E [\bar{x}_k^{[2]}] E [(\gamma_k - p_k) \Psi_s^T] + E[\Psi_k(\gamma_s - p_s)]E [\bar{x}_s^{[2]}]^T H_s^{[2]T} \\ &\quad + E[(\gamma_k - p_k)(\gamma_s - p_s)]H_k^{[2]} E [\bar{x}_k^{[2]}] E [\bar{x}_s^{[2]}]^T H_s^{[2]T}. \end{aligned}$$

- Usando que  $\{\bar{v}_k; k \geq 1\}$  es blanco e independiente de  $\bar{x}_k$ , se obtiene

$$\begin{aligned} E [\Psi_k \Psi_s^T] &= (I_{m^2} + K_{m^2}) [(H_k E [\bar{x}_k \bar{x}_s^T] H_s^T) \otimes E[\gamma_k \bar{v}_k \bar{v}_s^T \gamma_s]] (I_{m^2} + K_{m^2}) \\ &\quad + E [\bar{v}_k^{[2]} \bar{v}_s^{[2]T}]. \end{aligned}$$

Además, ya que  $E[\Psi_k]E [\Psi_s^T] = E [\bar{v}_k^{[2]}] E [\bar{v}_s^{[2]T}]$  se tiene que

$$\begin{aligned} E [\Psi_k \Psi_s^T] - E[\Psi_k]E [\Psi_s^T] &= (I_{m^2} + K_{m^2}) [(H_k P_{k,s} H_s^T) \otimes (\alpha_{ks} c_k c_s^T)] (I_{m^2} + K_{m^2}) \\ &\quad + \bar{R}_k^{(4)} \delta_{k-s}. \end{aligned}$$

- $E[(\gamma_k - p_k) \Psi_s^T] = [(E [\bar{x}_s^T] H_s^T) \otimes E[(\gamma_k - p_k) \gamma_s \bar{v}_s^T]] (I_{m^2} + K_{m^2}) + E[(\gamma_k - p_k) \bar{v}_s^{[2]T}]$ .

Teniendo en cuenta que  $E[\bar{x}_s^T] = 0$ , y que para  $s \neq k$ ,  $\gamma_k$  es independiente de  $\bar{v}_s$ , obtenemos que

$$E[(\gamma_k - p_k)\Psi_s^T] = p_k(1 - p_k)(1 - 2p_k)c_k^{[2]T}\delta_{k-s}.$$

- Finalmente, al ser  $\{\gamma_k; k \geq 1\}$  una sucesión de variables aleatorias de Bernoulli independientes, se tiene que  $E[(\gamma_k - p_k)(\gamma_s - p_s)] = p_k(1 - p_k)\delta_{k-s}$ .

Por lo tanto,

$$\begin{aligned} \bar{R}_{k,s}^{22} &= (I_{m^2} + K_{m^2}) [(H_k P_{k,s} H_s^T) \otimes [\alpha_{k,s} c_k c_s^T + p_k R_k \delta_{k-s}]] (I_{m^2} + K_{m^2}) \\ &+ \bar{R}_k^{(4)} \delta_{k-s} + p_k(1 - p_k)(1 - 2p_k) \left[ c_k^{[2]} \text{vec}^T(P_{k,k}) H_k^{[2]T} + H_k^{[2]} \text{vec}(P_{k,k}) c_k^{[2]} \right] \delta_{k-s} \\ &+ p_k(1 - p_k) H_k^{[2]} \text{vec}(P_{k,k}) \text{vec}^T(P_{k,k}) H_k^{[2]T} \delta_{k-s}. \end{aligned}$$

Por último, teniendo en cuenta la ecuación de la señal  $\bar{x}_k$ , y de nuevo que  $\{w_k; k \geq 0\}$  es un ruido blanco y centrado, se tiene que  $P_{k,s} = E[\bar{x}_k \bar{x}_s^T]$  verifica

$$P_{k,s} = F_{k-1} P_{k-1,s-1} F_{s-1}^T + Q_{k-1} \delta_{k-s}. \quad \blacksquare$$

A continuación probamos la incorrelación de los ruidos aditivos del sistema aumentado, así como la incorrelación de éstos con la señal inicial, propiedades que serán usadas en la obtención de los estimadores lineales en dicho sistema.

**Proposición 3.2.** *Los ruidos  $\{\bar{W}_k; k \geq 0\}$  y  $\{\bar{V}_k; k \geq 1\}$  son incorrelados e incorrelados con el estado inicial  $\bar{X}_0$ .*

**Demostración.** En virtud de las hipótesis de independencia impuesta sobre el sistema original (3.1) se prueba fácilmente que la señal inicial,  $\bar{X}_0$ , es incorrelada tanto con  $\{\bar{W}_k; k \geq 0\}$  como con  $\{\bar{V}_k; k \geq 1\}$ .

A continuación, se comprueba que los ruidos  $\{\bar{W}_k; k \geq 0\}$  y  $\{\bar{V}_k; k \geq 1\}$  son incorrelados. Ya que ambos son centrados, notando  $\tilde{V}_k = \begin{pmatrix} \bar{v}_k \\ \Psi_k - E[\Psi_k] \end{pmatrix}$ , y teniendo en cuenta la independencia entre  $\{\gamma_k; k \geq 1\}$  y  $\{\bar{W}_k; k \geq 0\}$  se tiene:

$$Cov(\bar{W}_k, \bar{V}_s^T) = E[\bar{W}_k \bar{V}_s^T] = E[\bar{W}_k \tilde{V}_s^T] + E[(\gamma_s - p_s) \bar{W}_k] E[\bar{X}_s]^T \bar{H}_s^T = E[\bar{W}_k \tilde{V}_s^T];$$

esto es,

$$Cov(\bar{W}_k, \bar{V}_s^T) = \begin{pmatrix} E[w_k \bar{v}_s^T] & E[w_k \Psi_s^T] \\ E[\Phi_k \bar{v}_s^T] & E[\Phi_k \Psi_s^T] - E[\Phi_k] E[\Psi_s^T] \end{pmatrix},$$

y con un procedimiento similar al utilizado en la demostración de la Proposición 2.3, se obtiene que todas las cajas de esta matriz son nulas, quedando de esta forma probada la incorrelación entre los ruidos  $\{\bar{W}_k; k \geq 0\}$  y  $\{\bar{V}_k; k \geq 1\}$ . ■

Indicamos, por último, que la independencia del ruido multiplicativo,  $\{\gamma_k; k \geq 1\}$ , con el ruido aditivo  $\{\bar{W}_k; k \geq 0\}$  y con la condición inicial,  $\bar{X}_0$ , se deduce de forma inmediata teniendo en cuenta la hipótesis de independencia mutua exigida a los procesos ruido del sistema original. También, usando esta hipótesis, es inmediato deducir que  $\gamma_k$  es independiente de  $(\bar{V}_1, \dots, \bar{V}_{k-1})$  para cualquier  $k \geq 1$ .

### 3.4. Filtro extendido cuadrático

Como hemos indicado al comienzo de este capítulo, para sistemas lineales con observaciones inciertas, ruidos blancos y bajo hipótesis de independencia mutua de los ruidos y la señal inicial, el estimador lineal de menor error cuadrático medio se obtiene mediante el filtro de Nahi [42]. Sin embargo, una vez estudiadas las propiedades de los procesos que intervienen en el sistema (3.3), se ha demostrado que éste no cumple las condiciones necesarias para aplicar este filtro, ya que el ruido aditivo de la ecuación de observación,  $\{\bar{V}_k; k \geq 1\}$  no es blanco y, además, depende del ruido multiplicativo.

Así, en esta sección vamos a obtener un algoritmo de estimación lineal para sistemas con observaciones inciertas, que extiende al algoritmo convencional de Nahi al debilitar la hipótesis de independencia mutua de los ruidos, y permitir correlación en el ruido aditivo de la observación.

### 3.4.1. Filtro lineal en sistemas con ruido de observación no blanco

Consideramos un sistema con observaciones inciertas,

$$x_k = F_{k-1}x_{k-1} + w_{k-1}, \quad k \geq 1$$

$$y_k = \gamma_k H_k x_k + v_k, \quad k \geq 1,$$

satisfaciendo las siguientes hipótesis:

- i)* La señal en el instante inicial,  $x_0$ , es un vector aleatorio  $n$ -dimensional, con vector de medias cero y matriz de covarianzas conocida,  $P_0$ .
- ii)*  $F_k$  y  $H_k$  son matrices conocidas de dimensión  $n \times n$  y  $m \times n$ , respectivamente.
- iii)*  $\{w_k; k \geq 0\}$  es un ruido blanco, centrado, con función de covarianzas  $E[w_k w_k^T] = Q_k$ .
- iv)*  $\{v_k; k \geq 1\}$  es un ruido centrado, con función de covarianzas  $E[v_k v_j^T] = R_{kj}$ .
- v)*  $\{\gamma_k; k \geq 1\}$  es una sucesión de variables aleatorias de Bernoulli independientes, con  $P(\gamma_k = 1) = p_k$ , independiente de la señal inicial,  $x_0$ , y del ruido  $\{w_k; k \geq 0\}$ , y satisfaciendo que  $\gamma_k$  es independiente de  $(v_1, \dots, v_{k-1})$ ,  $\forall k \geq 1$ .
- vi)* La señal inicial,  $x_0$ , y los ruidos  $\{w_k; k \geq 0\}$  y  $\{v_k; k \geq 1\}$  son mutuamente incorrelados.

Nuestro propósito es deducir un algoritmo recursivo para la obtención del filtro lineal de menor error cuadrático medio que, en virtud de los resultados expuestos en la Sección

2.2, es la proyección ortogonal de  $x_k$  en el espacio de combinaciones lineales de  $y_1, \dots, y_k$ . Buscamos, por tanto, una combinación lineal de las observaciones,  $\hat{x}_{k/k}$ , tal que el error de estimación,  $\tilde{x}_{k/k} = x_k - \hat{x}_{k/k}$ , sea ortogonal a los vectores  $y_1, \dots, y_k$ ; esto es:

$$E [\hat{x}_{k/k} y_i^T] = E [x_k y_i^T], \quad \forall i = 1, \dots, k,$$

expresión conocida como *ecuación de Wiener-Hopf*, que expresa que el error asociado a un estimador es incorrelado con las observaciones en las que se basa dicho estimador.

Para la deducción del algoritmo procederemos por etapas, deduciendo, en primer lugar, la expresión del filtro en función del predictor en cada instante y, posteriormente, la expresión del predictor en una etapa en función del filtro en la etapa anterior. En cada una de estas etapas obtendremos también las expresiones que relacionan las matrices de covarianzas de los correspondientes errores de estimación.

### Filtro en función del predictor

Partiendo de la expresión de  $\hat{x}_{k/k}$  como combinación lineal de las observaciones  $y_1, \dots, y_k$ ,

$$\hat{x}_{k/k} = \sum_{i=1}^k K_{k,i} y_i,$$

y usando la ecuación de Wiener-Hopf para dicho estimador, es claro que  $\hat{x}_{k/k} - K_{k,k} y_k$  es una combinación lineal de las observaciones  $y_1, \dots, y_{k-1}$  que satisface

$$E [(\hat{x}_{k/k} - K_{k,k} y_k) y_s^T] = E [(x_k - K_{k,k} y_k) y_s^T], \quad \forall s = 1, \dots, k-1.$$

Se deduce, por lo tanto, que  $\hat{x}_{k/k} - K_{k,k} y_k$  es el estimador lineal de menor error cuadrático medio de  $x_k - K_{k,k} y_k$  basado en  $y_1, \dots, y_{k-1}$ .

Por otra parte, debido a la linealidad de las proyecciones ortogonales, el estimador lineal de menor error cuadrático medio de  $x_k - K_{k,k} y_k$  basado en las observaciones

$y_1, \dots, y_{k-1}$  es  $\hat{x}_{k/k-1} - K_{k,k}\hat{y}_{k/k-1}$ , de donde se deduce la siguiente expresión para obtener el filtro en función del predictor:

$$\hat{x}_{k/k} = \hat{x}_{k/k-1} + K_k \tilde{y}_{k/k-1}, \quad k \geq 1,$$

donde  $\tilde{y}_{k/k-1} = y_k - \hat{y}_{k/k-1}$  constituye la innovación en el instante  $k$ , y  $K_k = K_{k,k}$  denota la matriz de ganancia del filtro.

Seguidamente deducimos la expresión de la matriz de covarianzas del error de filtrado en términos de la correspondiente del error de predicción.

Notando  $\tilde{x}_{k/k} = x_k - \hat{x}_{k/k}$  y  $\tilde{x}_{k/k-1} = x_k - \hat{x}_{k/k-1}$  a los errores de filtrado y predicción, respectivamente, la expresión del filtro en función del predictor nos conduce a la siguiente relación:

$$\tilde{x}_{k/k} = \tilde{x}_{k/k-1} - K_k \tilde{y}_{k/k-1}, \quad k \geq 1,$$

y notando  $P_{k/j}^x = E[\tilde{x}_{k/j}\tilde{x}_{k/j}^T]$ ,  $j = k-1, k$  a las matrices de covarianzas de los errores de predicción y filtrado, respectivamente, se tiene:

$$P_{k/k}^x = P_{k/k-1}^x + K_k E \left[ \tilde{y}_{k/k-1} \tilde{y}_{k/k-1}^T \right] K_k^T - E \left[ \tilde{x}_{k/k-1} \tilde{y}_{k/k-1}^T \right] K_k^T - K_k E \left[ \tilde{y}_{k/k-1} \tilde{x}_{k/k-1}^T \right], \quad k \geq 1.$$

Teniendo en cuenta que el error  $\tilde{x}_{k/k-1}$  es incorrelado con  $\hat{y}_{k/k-1}$ , combinación lineal de las observaciones  $y_1, \dots, y_{k-1}$ , se tiene que  $E \left[ \tilde{x}_{k/k-1} \tilde{y}_{k/k-1}^T \right] = E \left[ \tilde{x}_{k/k-1} y_k^T \right]$ , y expresando ahora  $\tilde{x}_{k/k-1}$  en términos de  $\tilde{x}_{k/k}$ :

$$E \left[ \tilde{x}_{k/k-1} y_k^T \right] = E \left[ \tilde{x}_{k/k} y_k^T \right] + K_k E \left[ \tilde{y}_{k/k-1} y_k^T \right] = K_k E \left[ \tilde{y}_{k/k-1} \tilde{y}_{k/k-1}^T \right],$$

donde, por una parte, se ha usado la incorrelación entre el error  $\tilde{x}_{k/k}$  y la observación  $y_k$  y, por otra, expresando  $y_k = \hat{y}_{k/k-1} + \tilde{y}_{k/k-1}$ , la incorrelación entre el error  $\tilde{y}_{k/k-1}$  y el predictor  $\hat{y}_{k/k-1}$ .

Este desarrollo nos conduce a la siguiente expresión de la matriz de covarianzas del error de filtrado en función de la matriz de covarianzas del error de predicción:

$$P_{k/k}^x = P_{k/k-1}^x - K_k \Pi_k K_k^T, \quad k \geq 1,$$

donde  $\Pi_k = E \left[ \tilde{y}_{k/k-1} \tilde{y}_{k/k-1}^T \right]$  es la matriz de covarianzas de la innovación en el instante  $k$ , de la que se obtendrá su expresión posteriormente.

A continuación, deducimos la expresión del predictor en términos del filtro en el instante anterior.

### Predictor en función del filtro

Consideramos las ecuaciones de Wiener-Hopf correspondientes al predictor,  $\hat{x}_{k/k-1}$  y al filtro en el instante anterior,  $\hat{x}_{k-1/k-1}$ :

$$E \left[ x_k y_i^T \right] = E \left[ \hat{x}_{k/k-1} y_i^T \right], \quad \forall i = 1, \dots, k-1$$

$$E \left[ x_{k-1} y_i^T \right] = E \left[ \hat{x}_{k-1/k-1} y_i^T \right], \quad \forall i = 1, \dots, k-1,$$

y teniendo en cuenta que  $w_{k-1}$  es centrado e incorrelado con  $y_i$ ,  $i < k$ , se tiene:

$$E \left[ x_k y_i^T \right] = E \left[ (F_{k-1} x_{k-1} + w_{k-1}) y_i^T \right] = F_{k-1} E \left[ \hat{x}_{k-1/k-1} y_i^T \right], \quad \forall i = 1, \dots, k-1,$$

con lo que queda probado que

$$\hat{x}_{k/k-1} = F_{k-1} \hat{x}_{k-1/k-1}, \quad k \geq 1.$$

Para calcular la matriz de covarianzas del error de predicción en términos de la matriz de covarianzas del error de filtrado, expresamos el error  $\tilde{x}_{k/k-1}$  en función del error  $\tilde{x}_{k-1/k-1}$ , teniendo en cuenta la ecuación de la señal, y la expresión anteriormente obtenida para el predictor en términos del filtro:

$$\tilde{x}_{k/k-1} = F_{k-1} \tilde{x}_{k-1/k-1} + w_{k-1}, \quad k \geq 1.$$



Así, teniendo en cuenta que  $\{w_k; k \geq 0\}$  es un ruido blanco incorrelado con  $x_0$  y, por tanto,  $w_{k-1}$  y  $\tilde{x}_{k-1/k-1}$  son incorrelados, se tiene:

$$\begin{aligned} P_{k/k-1}^x &= E [\tilde{x}_{k/k-1} \tilde{x}_{k/k-1}^T] = F_{k-1} E [\tilde{x}_{k-1/k-1} \tilde{x}_{k-1/k-1}^T] F_{k-1}^T + E [w_{k-1} w_{k-1}^T] \\ &= F_{k-1} P_{k-1/k-1}^x F_{k-1}^T + Q_{k-1}, \quad k \geq 1. \end{aligned}$$

Las ecuaciones que relacionan el filtro y el predictor proporcionan un algoritmo de filtrado recursivo, partiendo del filtro en el instante inicial,  $\hat{x}_{0/0} = 0$ , que, en cada etapa de actualización, requiere el cálculo de la innovación y de la matriz de ganancia del filtro. Por tanto, el siguiente paso para la deducción del algoritmo es calcular la expresión del proceso innovación.

### Proceso innovación

Con objeto de determinar la innovación,  $\tilde{y}_k = y_k - \hat{y}_{k/k-1}$ , consideramos la ecuación de Wiener-Hopf para el predictor  $\hat{y}_{k/k-1}$ :

$$E [y_k y_i^T] = E [\hat{y}_{k/k-1} y_i^T], \quad i = 1, \dots, k-1,$$

A partir la hipótesis  $v$ ), se deduce que  $\gamma_k$  es independiente de  $(x_k, y_1, \dots, y_{k-1})$  y, por tanto:

$$E [y_k y_i^T] = E [(\gamma_k H_k x_k + v_k) y_i^T] = p_k H_k E [\hat{x}_{k/k-1} y_i^T] + E [\hat{v}_{k/k-1} y_i^T], \quad i = 1, \dots, k-1,$$

de donde se deduce que

$$\hat{y}_{k/k-1} = p_k H_k \hat{x}_{k/k-1} + \hat{v}_{k/k-1}, \quad k \geq 1,$$

siendo  $\hat{v}_{k/k-1}$  el predictor en una etapa de  $v_k$ .

Así, la innovación en el instante  $k$ ,  $\tilde{y}_{k/k-1} = y_k - \hat{y}_{k/k-1}$ , viene dada por

$$\begin{aligned} \tilde{y}_{k/k-1} &= y_k - p_k H_k \hat{x}_{k/k-1} - \hat{v}_{k/k-1} \\ &= (\gamma_k - p_k) H_k x_k + p_k H_k \tilde{x}_{k/k-1} + \tilde{v}_{k/k-1}, \quad k \geq 1, \end{aligned}$$

siendo  $\tilde{v}_{k/k-1} = v_k - \hat{v}_{k/k-1}$  el error asociado al predictor del ruido.

Es claro que los vectores innovación tienen medias cero y sus matrices de covarianzas,  $\Pi_k = E \left[ \tilde{y}_{k/k-1} \tilde{y}_{k/k-1}^T \right]$ , vienen dadas por

$$\Pi_k = p_k(1 - p_k)H_k P_k H_k^T + p_k^2 H_k P_{k/k-1}^x H_k^T + P_{k,k/k-1}^v + p_k H_k P_{k,k/k-1}^{xv} + p_k P_{k,k/k-1}^{vx} H_k^T,$$

donde  $P_{k,k/k-1}^v = E[\tilde{v}_{k/k-1} \tilde{v}_{k/k-1}^T]$  es la matriz de covarianzas del error de predicción de  $v_k$ ,  $P_{k,k/k-1}^{xv} = E[\tilde{x}_{k/k-1} \tilde{v}_{k/k-1}^T]$  es la matriz de covarianzas cruzadas del error de predicción de la señal y el ruido de observación, y  $P_{k,k} = E[x_k x_k^T]$  la matriz de covarianzas de la señal que, teniendo en cuenta la ecuación de la misma, claramente verifica

$$P_{k,k} = F_{k-1} P_{k-1,k-1} F_{k-1}^T + Q_{k-1}, \quad k \geq 1; \quad P_{0,0} = P_0.$$

### Matriz de ganancia del filtro

Partiendo de la ecuación que relaciona los errores de filtrado y predicción,

$$\tilde{x}_{k/k} = \tilde{x}_{k/k-1} - K_k \tilde{y}_{k/k-1}, \quad k \geq 1,$$

se tiene:

$$E \left[ \tilde{x}_{k/k} \tilde{y}_{k/k-1}^T \right] = E \left[ \tilde{x}_{k/k-1} \tilde{y}_{k/k-1}^T \right] - K_k E \left[ \tilde{y}_{k/k-1} \tilde{y}_{k/k-1}^T \right].$$

- $E \left[ \tilde{x}_{k/k} \tilde{y}_{k/k-1}^T \right] = 0$ , ya que la innovación  $\tilde{y}_{k/k-1}^T$  es una combinación lineal de  $y_1, \dots, y_k$  y, por tanto, incorrelada con el error de filtrado.
- Sustituyendo la expresión de la innovación en  $E \left[ \tilde{x}_{k/k-1} \tilde{y}_{k/k-1}^T \right]$  se tiene:

$$E \left[ \tilde{x}_{k/k-1} \tilde{y}_{k/k-1}^T \right] = p_k P_{k/k-1}^x H_k^T + P_{k,k/k-1}^{xv},$$

Por lo tanto, la matriz de ganancia viene dada por la siguiente expresión:

$$K_k = \left[ p_k P_{k/k-1}^x H_k^T + P_{k,k/k-1}^{xv} \right] \Pi_k^{-1}.$$

A la vista de las expresiones obtenidas para la innovación y la matriz de ganancia del filtro, necesitamos conocer el predictor en una etapa del ruido de la observación y la correspondiente matriz de covarianzas del error, así como la matriz de covarianzas cruzadas de los errores de predicción de la señal y de dicho ruido. Procedemos a continuación a calcular estos estadísticos.

### Predictor del ruido de la observación

Con un razonamiento similar al realizado para obtener el filtro de la señal en función del predictor, se deduce que el predictor en una etapa del ruido,  $\hat{v}_{k/l-1}$ , en cualquier instante  $k$ , puede obtenerse mediante la siguiente expresión recursiva, partiendo de la condición inicial  $\hat{v}_{k/0} = E[v_k] = 0, \forall k \geq 1$ :

$$\hat{v}_{k/l} = \hat{v}_{k/l-1} + G_{k,l} \tilde{y}_{l/l-1}, \quad l < k, \quad k > 1,$$

siendo  $G_{k,l}$  la matriz de ganancia, que se obtiene mediante un razonamiento similar al usado para obtener la matriz de ganancia del filtro, como mostramos a continuación.

De la expresión recursiva para  $\hat{v}_{k/l}$  es inmediato deducir que el error  $\tilde{v}_{k/l} = v_k - \hat{v}_{k/l}$  verifica la siguiente:

$$\tilde{v}_{k/l} = \tilde{v}_{k/l-1} - G_{k,l} \tilde{y}_{l/l-1}, \quad l < k, \quad k > 1,$$

y, por tanto,

$$E [\tilde{v}_{k/l} \tilde{y}_{l/l-1}^T] = E [\tilde{v}_{k/l-1} \tilde{y}_{l/l-1}^T] - G_{k,l} E [\tilde{y}_{l/l-1} \tilde{y}_{l/l-1}^T],$$

de donde se deduce que

$$G_{k,l} = [p_l P_{k,l/l-1}^{vx} H_l^T + P_{k,l/l-1}^v] \Pi_l^{-1}.$$

Las matrices de covarianzas de los errores asociados a las sucesivas etapas de predicción del ruido,  $P_{k,l/l-1}^v$ , se obtienen recursivamente a partir de las siguientes relaciones:

$$P_{k,l/j}^v = E [\tilde{v}_{k/j} \tilde{v}_{l/j}^T] = P_{k,l/j-1}^v - G_{k,j} \Pi_j G_{l,j}^T, \quad k, l > 1, \quad j < \min(k, l),$$

$$P_{k,l/0}^v = R_{k,l}, \quad k, l \geq 1,$$

que, a su vez, se deducen de la expresión recursiva de los errores  $\tilde{v}_{k/j}$  y  $\tilde{v}_{l/j}$ , usando argumentos similares a los de la obtención de  $P_{k/k}^x$  a partir de  $P_{k/k-1}^x$ .

Para completar el algoritmo, calculamos seguidamente las matrices de covarianzas cruzadas  $P_{k,l/l-1}^{vx}$ , que se obtienen teniendo en cuenta de nuevo la expresiones de  $\tilde{v}_{k/l-1}$  en función de  $\tilde{v}_{k/l-2}$ , y la de  $\tilde{x}_{l/l-1}$  en función de  $\tilde{x}_{l-1/l-2}$ :

$$P_{k,l/l-1}^{vx} = E [\tilde{v}_{k/l-1} \tilde{x}_{l/l-1}^T] = [P_{k,l-1/l-2}^{vx} - G_{k,l-1} \Pi_{l-1} K_{l-1}^T] F_{l-1}^T, \quad l \leq k.$$

Notemos que, en particular, las matrices  $P_{k,k/k-1}^v$  y  $P_{k,k/k-1}^{vx}$  son también requeridas para el cálculo de la innovación.

Todas las expresiones anteriores se resumen en el siguiente algoritmo:

### Filtro lineal a partir de observaciones inciertas con ruido aditivo no blanco

$$\hat{x}_{k/k} = \hat{x}_{k/k-1} + K_k \tilde{y}_{k/k-1}, \quad k \geq 1; \quad \hat{x}_{0/0} = 0$$

$$\hat{x}_{k/k-1} = F_{k-1} \hat{x}_{k-1/k-1}, \quad k \geq 1$$

$$K_k = [p_k P_{k/k-1}^x H_k^T + P_{k,k/k-1}^{xv}] \Pi_k^{-1}, \quad k \geq 1$$

$$\tilde{y}_{k/k-1} = y_k - p_k H_k \hat{x}_{k/k-1} - \hat{v}_{k/k-1}, \quad k \geq 1,$$

$$\Pi_k = p_k (1 - p_k) H_k P_{k,k} H_k^T + p_k^2 H_k P_{k/k-1}^x H_k^T + P_{k,k/k-1}^v + p_k H_k P_{k,k/k-1}^{xv} + p_k P_{k,k/k-1}^{vx} H_k^T, \quad k \geq 1,$$

$$\begin{aligned}
P_{k,k} &= F_{k-1}P_{k-1,k-1}F_{k-1}^T + Q_{k-1}, \quad k \geq 1; \quad P_{0,0} = P_0 \\
P_{k/k}^x &= P_{k/k-1}^x - K_k \Pi_k K_k^T, \quad k \geq 1; \quad P_{0/0}^x = P_0 \\
P_{k/k-1}^x &= F_{k-1}P_{k-1/k-1}^x F_{k-1}^T + Q_{k-1}, \quad k \geq 1 \\
\hat{v}_{k/l} &= \hat{v}_{k/l-1} + G_{k,l} \tilde{y}_{l/l-1}, \quad l < k, \quad k > 1; \quad \hat{v}_{k/0} = 0 \quad k \geq 1 \\
G_{k,l} &= [p_l P_{k,l/l-1}^{vx} H_l^T + P_{k,l/l-1}^v] \Pi_l^{-1}, \quad l < k, \quad k \geq 1 \\
P_{k,l/j}^v &= P_{k,l/j-1}^v - G_{k,j} \Pi_j G_{l,j}^T, \quad k, l \geq 1, \quad j < \min(k, l); \quad P_{k,l/0}^v = R_{k,l}, \quad k, l \geq 1 \\
P_{k,l/l-1}^{vx} &= [P_{k,l-1/l-2}^{vx} - G_{k,l-1} \Pi_{l-1} K_{l-1}^T] F_{l-1}^T, \quad l \leq k, \quad k \geq 1; \quad P_{k,1/0}^{vx} = 0.
\end{aligned}$$

### 3.4.2. Algoritmo de filtrado lineal en el sistema aumentado

Consideramos el sistema aumentado (3.3),

$$\begin{aligned}
\bar{X}_k &= \bar{F}_{k-1} \bar{X}_{k-1} + \bar{W}_{k-1}, \quad k \geq 1 \\
\bar{Y}_k &= \gamma_k \bar{H}_k \bar{X}_k + \bar{V}_k, \quad k \geq 1.
\end{aligned}$$

Las propiedades de los procesos que intervienen en este sistema, analizadas en la Sección 3.3.2, permiten aplicar el algoritmo que acabamos de desarrollar en la Sección 3.4.1 y, por tanto, el estimador lineal de la señal aumentada,  $\bar{X}_k$ , obtenido a partir del conjunto de observaciones  $\{\bar{Y}_1, \dots, \bar{Y}_k\}$ , viene dado por las siguientes ecuaciones:

$$\begin{aligned}
\hat{\bar{X}}_{k/k} &= \hat{\bar{X}}_{k/k-1} + \bar{K}_k \tilde{\bar{Y}}_{k/k-1}, \quad k \geq 1 \\
\hat{\bar{X}}_{k/k-1} &= \bar{F}_{k-1} \hat{\bar{X}}_{k-1/k-1}, \quad k \geq 1 \\
\bar{K}_k &= [p_k P_{k/k-1}^{\bar{X}} \bar{H}_k^T + P_{k,k/k-1}^{\bar{X}\bar{V}}] \bar{\Pi}_k^{-1}, \quad k \geq 1 \\
\tilde{\bar{Y}}_{k/k-1} &= \bar{Y}_k - p_k \bar{H}_k \hat{\bar{X}}_{k/k-1} - \hat{\bar{V}}_{k/k-1}, \quad k \geq 1 \\
\bar{\Pi}_k &= p_k (1 - p_k) \bar{H}_k P_{k,k}^{\bar{X}} \bar{H}_k^T + p_k^2 \bar{H}_k P_{k/k-1}^{\bar{X}} \bar{H}_k^T + P_{k,k/k-1}^{\bar{V}} + p_k \bar{H}_k P_{k,k/k-1}^{\bar{X}\bar{V}} + p_k P_{k,k/k-1}^{\bar{V}\bar{X}} \bar{H}_k^T, \quad k \geq 1 \\
P_{k,k}^{\bar{X}} &= \bar{F}_{k-1} P_{k-1,k-1}^{\bar{X}} \bar{F}_{k-1}^T + \bar{Q}_{k-1}, \quad k \geq 1; \quad P_{0,0}^{\bar{X}} = P_{0/0}^{\bar{X}},
\end{aligned}$$

$$P_{k/k}^{\bar{X}} = P_{k/k-1}^{\bar{X}} - \bar{K}_k \bar{\Pi}_k \bar{K}_k^T, \quad k \geq 1$$

$$P_{k/k-1}^{\bar{X}} = \bar{F}_{k-1} P_{k-1/k-1}^{\bar{X}} \bar{F}_{k-1}^T + \bar{Q}_{k-1}, \quad k \geq 1$$

$$\hat{V}_{k/j} = \hat{V}_{k/j-1} + \bar{G}_{k,j} \tilde{Y}_{j/j-1}, \quad j < k, \quad k > 1, \quad \hat{V}_{k/0} = 0 \quad k \geq 1$$

$$\bar{G}_{k,j} = \left[ p_j P_{k,j/j-1}^{\bar{V}\bar{X}} \bar{H}_j^T + P_{k,j/j-1}^{\bar{V}} \right] \bar{\Pi}_j^{-1}, \quad j < k, \quad k \geq 1$$

$$P_{k,j/l}^{\bar{V}} = P_{k,j/l-1}^{\bar{V}} - \bar{G}_{k,l} \bar{\Pi}_l \bar{G}_{j,l}^T, \quad k, j \geq 1, \quad l < \min(k, j); \quad P_{k,j/0}^{\bar{V}} = \mathcal{R}_{k,j}, \quad k, j \geq 1$$

$$P_{k,j/j-1}^{\bar{V}\bar{X}} = \left[ P_{k,j-1/j-2}^{\bar{V}\bar{X}} - \bar{G}_{k,j-1} \bar{\Pi}_{j-1} \bar{K}_{j-1}^T \right] \bar{F}_{j-1}^T, \quad j \leq k, \quad k \geq 1,$$

cuyas condiciones iniciales son

$$\hat{X}_{0/0} = 0, \quad P_{0/0}^{\bar{X}} = \begin{pmatrix} P_0 & P_0^{(3)} \\ P_0^{(3)T} & P_0^{(4)} \end{pmatrix}.$$

Finalmente, como en el caso de sistemas no lineales con certidumbre, el estimador cuadrático de  $x_k$  se obtiene a partir del vector constituido por las primeras  $n$  componentes del estimador lineal de la señal aumentada. Teniendo en cuenta que la señal original,  $x_k$ , ha sido centrada previamente, el estimador requerido se obtiene sumando a dicho vector la media de  $x_k$ , que se obtiene recursivamente mediante la relación:

$$E[x_k] = F_{k-1} E[x_{k-1}] + u_{k-1}, \quad k \geq 1.$$

### 3.5. Filtro ensemble

El objetivo de esta sección es extender la metodología expuesta en la Sección 1.5 para la obtención del filtro de Kalman ensemble a sistemas con observaciones inciertas en los que la incertidumbre está modelizada por una sucesión de variables aleatorias de Bernoulli independientes, como venimos considerando en este capítulo.

Como se expuso en la Sección 1.5, la metodología del filtro de Kalman ensemble está basada en la simulación de la señal en el instante inicial, y sucesivas simulaciones

de los ruidos que perturban la evolución de la señal y las observaciones. En el caso de observaciones afectada por incertidumbre, la ecuación que las describe está perturbada, además de por ruido aditivo, por un ruido multiplicativo, lo que implica que la adaptación de la metodología de filtrado ensemble a sistemas no lineales con observaciones inciertas debe conllevar la simulación de dicho ruido en cada instante de tiempo.

Consideremos, por tanto, un sistema no lineal con observaciones inciertas, descrito por las ecuaciones

$$\begin{aligned} x_k &= f_{k-1}(x_{k-1}) + w_{k-1}, \quad k \geq 1 \\ y_k &= \gamma_k h_k(x_k) + v_k, \quad k \geq 1, \end{aligned} \tag{3.4}$$

satisfaciendo las siguientes hipótesis:

- i)* La señal en el instante inicial,  $x_0$ , es un vector aleatorio  $n$ -dimensional, con media cero y matriz de covarianzas conocida,  $P_0$ .
- ii)* El proceso  $\{w_k; k \geq 0\}$  es un ruido blanco  $n$ -dimensional, centrado, con función de covarianzas  $E[w_k w_k^T] = Q_k$ .
- iii)* El proceso  $\{v_k; k \geq 1\}$  es un ruido blanco  $m$ -dimensional, centrado, con función de covarianzas  $E[v_k v_k^T] = R_k$ .
- iv)*  $\{\gamma_k; k \geq 1\}$  es una sucesión de variables aleatorias de Bernoulli independientes, con  $P(\gamma_k = 1) = p_k$ .
- v)*  $f_k : \mathbb{R}^n \rightarrow \mathbb{R}^n$  y  $h_k : \mathbb{R}^n \rightarrow \mathbb{R}^m$  son funciones no lineales.
- vi)* La señal inicial,  $x_0$ , y los ruidos  $\{w_k; k \geq 0\}$ ,  $\{v_k; k \geq 1\}$  y  $\{\gamma_k; k \geq 1\}$  son mutuamente independientes.

Para la adaptación del filtro ensemble a este tipo de sistemas hemos de tener en cuenta que el ruido multiplicativo que interviene en el sistema no lineal bajo estudio sólo afecta a la ecuación de observación, por lo que las modificaciones sobre el algoritmo establecido en la Sección 1.5, sólo deben hacerse en la etapa de actualización. A continuación hacemos una descripción completa de ambas etapas, la de predicción y la de actualización.

### Etapa de predicción

Como ya hemos indicado, el procedimiento a seguir en esta etapa es totalmente similar al realizado en el caso de certidumbre en las observaciones, ya que en ella sólo interviene la ecuación de la señal. Por tanto:

- En el instante  $k = 1$ , se parte de un conjunto de puntos muestrales, que notaremos  $\hat{x}_{0/0}^1, \dots, \hat{x}_{0/0}^q$ , simulados de la distribución de la señal en el instante inicial,  $x_0$ . Para  $k > 1$ , se parte de un conjunto de  $q$  estimadores de la señal,  $\hat{x}_{k-1/k-1}^1, \dots, \hat{x}_{k-1/k-1}^q$ , que se habrán obtenido en la etapa anterior del algoritmo.
- En cada instante  $k \geq 1$ , se realizan  $q$  simulaciones de la distribución del ruido  $w_{k-1}$ , que notamos  $w_{k-1}^1, \dots, w_{k-1}^q$ , y cada uno de los estimadores de filtrado de partida se propaga a través de la ecuación de la señal, considerando los valores simulados del ruido; con ello se obtiene el siguiente conjunto de estimadores de  $x_k$ , basados en las observaciones  $y_1, \dots, y_{k-1}$ :

$$\hat{x}_{k/k-1}^i = f_{k-1}(\hat{x}_{k-1/k-1}^i) + w_{k-1}^i, \quad i = 1, \dots, q,$$

- Una vez obtenidos estos estimadores, el estimador de menor error cuadrático medio de  $x_k$  basado en  $y_1, \dots, y_{k-1}$  y la correspondiente matriz de covarianzas del error son aproximados por la media y por la cuasicovarianza muestral de dichos estimadores,



respectivamente; esto es, notando, como es usual,  $\hat{x}_{k/k-1}$  al predictor en una etapa de  $x_k$  y  $P_{k/k-1}$  a la matriz de covarianzas del error, se tiene:

$$\hat{x}_{k/k-1} = \frac{1}{q} \sum_{i=1}^q \hat{x}_{k/k-1}^i, \quad k \geq 1$$

$$P_{k/k-1} = \frac{1}{q-1} \sum_{i=1}^q (\hat{x}_{k/k-1}^i - \hat{x}_{k/k-1})(\hat{x}_{k/k-1}^i - \hat{x}_{k/k-1})^T, \quad k \geq 1.$$

A continuación, estos estadísticos son actualizados con la información proporcionada por la observación en el instante  $k$  como sigue.

### Etapa de actualización

El primer paso para la etapa de actualización de los estimadores de  $x_k$  basados en  $y_1, \dots, y_{k-1}$  es obtener predictores de  $y_k$  a partir de los predictores disponibles de la señal,  $\hat{x}_{k/k-1}^i$ ,  $i = 1, \dots, k$ , lo que se consigue propagando cada uno de ellos por la ecuación de observación.

Para la propagación hemos de tener en cuenta que la ecuación de observación está ahora perturbada, además de por el ruido aditivo, por el ruido multiplicativo que describe la incertidumbre; por lo tanto, siguiendo la metodología del filtro de Kalman ensemble, ahora realizaremos  $q$  simulaciones del ruido aditivo  $v_k$ , que serán denotadas  $v_k^1, \dots, v_k^q$  y  $q$  simulaciones del ruido multiplicativo  $\gamma_k$ , que serán denotadas  $\gamma_k^1, \dots, \gamma_k^q$ ; así, propagando el conjunto de predictores  $\hat{x}_{k/k-1}^1, \dots, \hat{x}_{k/k-1}^q$  a través de la ecuación de observación en estos valores simulados obtenemos  $q$  predictores de  $y_k$ :

$$\hat{y}_{k/k-1}^i = \gamma_k^i h_k(\hat{x}_{k/k-1}^i) + v_k^i, \quad i = 1, \dots, q.$$

Una vez obtenidos los puntos transformados  $\hat{y}_{k/k-1}^1, \dots, \hat{y}_{k/k-1}^q$ , los pasos a seguir son los mismos que en el caso de certidumbre en las observaciones, que especificamos a continuación.

- Se aproxima el predictor de la observación,  $\hat{y}_{k/k-1}$ , y la correspondiente matriz de covarianzas del error,  $P_{k/k-1}^y$ , por la media y cuasicovarianza muestral de los puntos transformados,  $\hat{y}_{k/k-1}^1, \dots, \hat{y}_{k/k-1}^q$ :

$$\hat{y}_{k/k-1} = \frac{1}{q} \sum_{i=1}^q \hat{y}_{k/k-1}^i, \quad k \geq 1$$

$$P_{k/k-1}^y = \frac{1}{q-1} \sum_{i=1}^q (\hat{y}_{k/k-1}^i - \hat{y}_{k/k-1})(\hat{y}_{k/k-1}^i - \hat{y}_{k/k-1})^T, \quad k \geq 1.$$

- Se aproxima la matriz de covarianzas cruzadas de los errores de predicción de la señal y de la observación,  $P_{k/k-1}^{xy}$ , por la cuasicovarianza de los estimadores de predicción  $\hat{x}_{k/k-1}^1, \dots, \hat{x}_{k/k-1}^q$  e  $\hat{y}_{k/k-1}^1, \dots, \hat{y}_{k/k-1}^q$ :

$$P_{k/k-1}^{xy} = \frac{1}{q-1} \sum_{i=1}^q (\hat{x}_{k/k-1}^i - \hat{x}_{k/k-1})(\hat{y}_{k/k-1}^i - \hat{y}_{k/k-1})^T, \quad k \geq 1.$$

- Finalmente, el filtro de la señal,  $\hat{x}_{k/k}$ , y la matriz de covarianzas del error de filtrado,  $P_{k/k}$ , se calculan mediante las siguientes expresiones:

$$\hat{x}_{k/k} = \frac{1}{q} \sum_{i=1}^q \hat{x}_{k/k}^i, \quad k \geq 1$$

$$P_{k/k} = \frac{1}{q-1} \sum_{i=1}^q (\hat{x}_{k/k}^i - \hat{x}_{k/k})(\hat{x}_{k/k}^i - \hat{x}_{k/k})^T, \quad k \geq 1,$$

donde los filtros  $\hat{x}_{k/k}^i$ , se obtienen aplicando la fórmula del filtro de Kalman en su etapa de actualización:

$$\hat{x}_{k/k}^i = \hat{x}_{k/k-1}^i + P_{k/k-1}^{xy} (P_{k/k-1}^y)^{-1} [y_k - \hat{y}_{k/k-1}^i], \quad i = 1, \dots, q.$$

### 3.6. Filtro ensemble cuadrático

En la sección anterior se ha propuesto una generalización del filtro del Kalman ensemble para el caso de sistemas no lineales con observaciones inciertas. Siguiendo la línea de desarrollo de nuestro trabajo, el objetivo en esta sección es generalizar este algoritmo considerando estimadores basados en las observaciones y sus potencias de segundo grado, definidas, como viene siendo usual, mediante el producto de Kronecker.

Al igual que en los algoritmos de filtrado cuadrático propuestos en el Capítulo 2 para sistemas con certidumbre, y en este capítulo para sistemas con observaciones inciertas, el procedimiento para la obtención de un filtro cuadrático es construir un nuevo sistema en el que los vectores señal y observación son los aumentados de los correspondientes originales con sus potencias Kronecker de segundo grado.

En los algoritmos propuestos en las secciones anteriores, basados en la aproximación de los sistemas originales por sistemas lineales, se definían los sistemas aumentados correspondientes a los sistemas linealizados; ahora, para seguir la metodología de filtrado ensemble, aumentaremos directamente el sistema original, considerando las potencias de segundo grado de la señal y la observación.

Concretamente, vamos a considerar el sistema (3.4):

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1}, \quad k \geq 1$$

$$y_k = \gamma_k h_k(x_k) + v_k, \quad k \geq 1,$$

y los vectores aumentados,

$$X_k = \begin{pmatrix} x_k \\ x_k^{[2]} \end{pmatrix} \in \mathbb{R}^{n+n^2}, \quad Y_k = \begin{pmatrix} y_k \\ y_k^{[2]} \end{pmatrix} \in \mathbb{R}^{m+m^2}.$$

Para obtener la dinámica de los vectores  $X_k$  e  $Y_k$  comenzamos calculando las potencias Kronecker de segundo grado de los vectores señal y observación originales,  $x_k^{[2]}$  e  $y_k^{[2]}$ .

### Potencias de segundo grado de la señal

Teniendo en cuenta la ecuación de la señal del sistema (3.4) se obtiene la siguiente expresión:

$$x_k^{[2]} = f_{k-1}^{[2]}(x_{k-1}) + \Phi_{k-1}, \quad k \geq 1,$$

donde

$$\Phi_k = (I_{n^2} + K_{n^2})[f_k(x_k) \otimes w_k] + w_k^{[2]}, \quad k \geq 0.$$

### Potencias de segundo grado de la observación

Procediendo de forma análoga con la ecuación de observación del sistema (3.4), y dado que  $\gamma_k$  sólo toma los valores 0 y 1, se obtiene:

$$y_k^{[2]} = \gamma_k h_k^{[2]}(x_k) + \Psi_k, \quad k \geq 1,$$

siendo

$$\Psi_k = (I_{m^2} + K_{m^2})[\gamma_k h_k(x_k) \otimes v_k] + v_k^{[2]}, \quad k \geq 1.$$

Una vez obtenidas estas expresiones, si definimos

$$F_k(X_k) = \begin{pmatrix} f_k(x_k) \\ f_k^{[2]}(x_k) \end{pmatrix}, \quad H_k(X_k) = \begin{pmatrix} h_k(x_k) \\ h_k^{[2]}(x_k) \end{pmatrix}, \quad W_k = \begin{pmatrix} w_k \\ \Phi_k \end{pmatrix}, \quad V_k = \begin{pmatrix} v_k \\ \Psi_k \end{pmatrix},$$

obtenemos las siguientes ecuaciones que modelizan la evolución de la señal aumentada y las observaciones aumentadas:

$$\begin{aligned} X_k &= F_{k-1}(X_{k-1}) + W_{k-1}, \quad k \geq 1, \\ Y_k &= \gamma_k H_k(X_k) + V_k, \quad k \geq 1. \end{aligned} \tag{3.5}$$

El sistema aumentado (3.5) es, naturalmente, un sistema no lineal, en el que las observaciones están perturbadas por el mismo ruido multiplicativo que el sistema original,  $\{\gamma_k; k \geq 1\}$ ; por tanto, podemos aplicar el algoritmo de filtrado ensemble desarrollado en

la sección anterior, que particularizado a este sistema actúa como indicamos a continuación.

### Etapa de predicción

- En el instante  $k = 1$ , se parte de un conjunto de puntos muestrales,  $\hat{X}_{0/0}^1, \dots, \hat{X}_{0/0}^q$ , definidos como

$$\hat{X}_{0/0}^i = \begin{pmatrix} \hat{x}_{0/0}^i \\ \hat{x}_{0/0}^{i[2]} \end{pmatrix}, \quad i = 1, \dots, q,$$

donde los valores  $\hat{x}_{0/0}^i$  son simulados de la distribución de la señal inicial,  $x_0$ , y  $\hat{x}_{0/0}^{i[2]}$  son sus potencias de Kronecker de segundo grado.

Para  $k > 1$ , el punto de partida será el conjunto de estimadores de la señal aumentada,  $\hat{X}_{k-1/k-1}^1, \dots, \hat{X}_{k-1/k-1}^q$ , que se habrá obtenido en la etapa anterior del algoritmo.

- Propagando dichos estimadores a través de la ecuación de la señal aumentada del sistema (3.5), obtenemos un conjunto de  $q$  predictores,

$$\hat{X}_{k/k-1}^i = F_{k-1}(\hat{X}_{k-1/k-1}^i) + W_{k-1}^i, \quad i = 1, \dots, q$$

donde  $W_{k-1}^i = \begin{pmatrix} w_{k-1}^i \\ \Phi_{k-1}^i \end{pmatrix}$ , siendo  $w_{k-1}^i$  valores simulados de la distribución de  $w_{k-1}$  y  $\Phi_{k-1}^i$  calculados, según la expresión de  $\Phi_k$ , mediante la siguiente:

$$\Phi_{k-1}^i = (I_{n^2} + K_{n^2})[f_{k-1}(\hat{x}_{k-1/k-1}^i) \otimes w_{k-1}^i] + w_{k-1}^{i[2]},$$

donde  $\hat{x}_{k-1/k-1}^i$  es el vector constituido por las  $n$  primeras componentes de  $\hat{X}_{k-1/k-1}^i$ .

- Una vez obtenidos estos estimadores, aproximamos el predictor de la señal aumentada y la matriz de covarianzas del error de predicción por la media y cuasicovarianza

muestral de éstos:

$$\hat{X}_{k/k-1} = \frac{1}{q} \sum_{i=1}^q \hat{X}_{k/k-1}^i, \quad k \geq 1$$

$$P_{k/k-1} = \frac{1}{q-1} \sum_{i=1}^q (\hat{X}_{k/k-1}^i - \hat{X}_{k/k-1})(\hat{X}_{k/k-1}^i - \hat{X}_{k/k-1})^T, \quad k \geq 1.$$

### Etapas de actualización

En esta etapa, el punto de inicio es el conjunto de predictores  $\hat{X}_{k/k-1}^1, \dots, \hat{X}_{k/k-1}^q$ , obtenidos en la etapa anterior del algoritmo, que se propagan a través de la ecuación de observación del sistema (3.5):

$$\hat{Y}_{k/k-1}^i = \gamma_k^i H_k(\hat{X}_{k/k-1}^i) + V_k^i, \quad i = 1, \dots, q,$$

siendo  $V_k^i = \begin{pmatrix} v_k^i \\ \Psi_k^i \end{pmatrix}$ . Los ruidos  $v_k^i$  y  $\gamma_k^i$  son simulados a partir de las distribuciones de  $v_k$  y  $\gamma_k$ , respectivamente, y  $\Psi_k^i$  se obtiene mediante la siguiente expresión:

$$\Psi_k^i = (I_{m^2} + K_{m^2})[\gamma_k^i h_k(\hat{x}_{k/k-1}^i) \otimes v_k^i] + v_k^{i[2]}, \quad k \geq 1,$$

donde  $\hat{x}_{k/k-1}^i$  es el vector constituido por las  $n$  primeras componentes de  $\hat{X}_{k/k-1}^i$ .

A partir de los puntos transformados,  $\hat{Y}_{k/k-1}^1, \dots, \hat{Y}_{k/k-1}^q$ , se aproxima el predictor de la observación,  $\hat{Y}_{k/k-1}$ , y la correspondiente matriz de covarianzas del error,  $P_{k/k-1}^Y$ , por la media y cuasicovarianza de los mismos:

$$\hat{Y}_{k/k-1} = \frac{1}{q} \sum_{i=1}^q \hat{Y}_{k/k-1}^i, \quad k \geq 1$$

$$P_{k/k-1}^Y = \frac{1}{q-1} \sum_{i=1}^q (\hat{Y}_{k/k-1}^i - \hat{Y}_{k/k-1})(\hat{Y}_{k/k-1}^i - \hat{Y}_{k/k-1})^T, \quad k \geq 1.$$

Finalmente, la matriz de covarianzas cruzadas del error del predictor de la señal y de la observación,  $P_{k/k-1}^{XY}$ , se aproxima por la cuasicovarianza de los predictores de la señal y

de la observación,  $\hat{X}_{k/k-1}^1, \dots, \hat{X}_{k/k-1}^q$  e  $\hat{Y}_{k/k-1}^1, \dots, \hat{Y}_{k/k-1}^q$ :

$$P_{k/k-1}^{XY} = \frac{1}{q-1} \sum_{i=1}^q (\hat{X}_{k/k-1}^i - \hat{X}_{k/k-1})(\hat{Y}_{k/k-1}^i - \hat{Y}_{k/k-1})^T, \quad k \geq 1.$$

Una vez aproximados los anteriores estadísticos de primer y segundo orden, se actualiza el conjunto de predictores de la señal con la nueva observación  $Y_k$ , a partir de las siguientes ecuaciones:

$$\hat{X}_{k/k}^i = \hat{X}_{k/k-1}^i + P_{k/k-1}^{XY} (P_{k/k-1}^Y)^{-1} [Y_k - \hat{Y}_{k/k-1}^i], \quad i = 1, \dots, q,$$

y, finalmente, se calcula el filtro de la señal aumentada,  $\hat{X}_{k/k}$ , y la matriz de covarianza del error de filtrado,  $P_{k/k}$ , como la media y cuasicovarianza de dichos puntos:

$$\hat{X}_{k/k} = \frac{1}{q} \sum_{i=1}^q \hat{X}_{k/k}^i, \quad k \geq 1$$

$$P_{k/k} = \frac{1}{q-1} \sum_{i=1}^q (\hat{X}_{k/k}^i - \hat{X}_{k/k})(\hat{X}_{k/k}^i - \hat{X}_{k/k})^T, \quad k \geq 1.$$

El proceso de estimación finaliza con la obtención del filtro de la señal original,  $\hat{x}_{k/k}$ , que se realiza extrayendo las  $n$  primeras componentes de  $\hat{X}_{k/k}$ .

### 3.7. Ejemplo numérico

En esta sección mostramos un ejemplo de simulación numérica para ilustrar los resultados obtenidos en este capítulo. Para ello, hemos considerado el mismo modelo no lineal para el que se han analizado los resultados del Capítulo 2, contemplando ahora la posibilidad de que las observaciones, de manera aleatoria, no contengan información sobre la señal, sino que consistan sólo en ruido. Así, consideramos el siguiente sistema no lineal con observaciones inciertas:

$$x_k = \frac{1}{x_{k-1}^2 + 3} + w_{k-1}, \quad k \geq 1$$

$$y_k = \gamma_k(x_k^2 + \exp(x_k)) + v_k, \quad k \geq 1,$$

sobre el que se suponen las siguientes hipótesis:

- i)* La señal inicial,  $x_0$ , es una variable aleatoria con media cero, varianza uno,  $P_0 = 1$ , y momentos de tercer y cuarto orden dados por  $P_0^{(3)} = 0$  y  $P_0^{(4)} = 3 - P_0^2$ , respectivamente.
- ii)*  $\{w_k; k \geq 0\}$  es un ruido blanco, centrado, con varianza constante,  $Q_k = \frac{19}{3}$ , y momentos de tercer y cuarto orden dados por  $Q_k^{(3)} = -\frac{128}{3}$  y  $Q_k^{(4)} = \frac{1123}{3} - Q_k^2$ , respectivamente.
- iii)*  $\{v_k; k \geq 1\}$  es un ruido blanco, centrado, con varianzas  $R_k = \frac{19}{3}$ , y momentos de tercer y cuarto orden dados por  $R_k^{(3)} = -\frac{128}{3}$  y  $R_k^{(4)} = \frac{1123}{3} - R_k^2$ , respectivamente.
- iv)*  $\{\gamma_k; k \geq 1\}$  es una sucesión de variables aleatorias de Bernoulli independientes con  $P(\gamma_k = 1) = p$ .
- v)* La señal inicial,  $x_0$ , y los ruidos  $\{w_k; k \geq 0\}$ ,  $\{v_k; k \geq 1\}$  y  $\{\gamma_k; k \geq 1\}$  son mutuamente independientes.

En este ejemplo se muestra la actuación de los algoritmos de filtrado extendido cuadrático y de filtrado ensemble, la comparación entre los mismos, así como la mejora que producen ambos con respecto al filtro linealizado mixturas propuesto por Hermoso y Linares. [25].

Para ello, se han implementado en Matlab los diferentes algoritmos, aplicándolos a la estimación de la señal en el modelo no lineal anterior; para cada algoritmo se han realizado 1000 simulaciones, y 50 iteraciones para cada simulación. Como en los ejemplos considerados en los capítulos anteriores, el comportamiento de los estimadores propuestos se compara mediante los errores cuadráticos medios correspondientes a las distintas



simulaciones de la señal y sus estimaciones:

$$ECM_k = \frac{1}{1000} \sum_{s=1}^{1000} (x_k^s - \hat{x}_{k/k}^s)^2, \quad k = 1, \dots, 50,$$

donde  $x_k^s$  denota el valor de la señal generado en la simulación  $s$  y la iteración  $k$ , y  $\hat{x}_{k/k}^s$  es el filtro calculado con el correspondiente algoritmo en dicha simulación e iteración.

### 3.7.1. Actuación del filtro extendido cuadrático

La actuación del filtro extendido cuadrático, así como su comparación con el filtro linealizado mixturas se ha llevado a cabo utilizando distintas probabilidades de falsa alarma, y suponiendo en cada caso que ésta es constante en todas las iteraciones. En todos los casos analizados, se ha apreciado una evidente mejora del filtro extendido cuadrático propuesto en este trabajo frente al linealizado mixturas de [25].

A título de ejemplo, en la Figura 3.1 presentamos los resultados obtenidos considerando  $P(\gamma_k = 1) = p = 0.5$ . Esta figura muestra la evolución de los errores cuadráticos medios de ambos filtros, reflejando, además de un mejor comportamiento del filtro extendido cuadrático, manifestado por errores sensiblemente inferiores a los obtenidos con el filtro linealizado mixturas, una menor dispersión en dichos errores a lo largo de las diferentes iteraciones, lo que reafirma la efectividad del algoritmo propuesto.

Por otra parte, hemos analizado también el comportamiento de las estimaciones obtenidas mediante el filtro extendido cuadrático frente a la probabilidad de falsa alarma en las observaciones. Para una mejor visualización de los resultados, éstos son mostrados en dos figuras; en la Figura 3.2 se representan los errores cuadráticos medios correspondientes a los valores  $p = 0.1$ ,  $p = 0.2$ ,  $p = 0.3$ ,  $p = 0.4$  y  $p = 0.5$ , y en la Figura 3.3 los correspondientes a  $p = 0.5$ ,  $p = 0.6$ ,  $p = 0.7$ ,  $p = 0.8$  y  $p = 0.9$ .

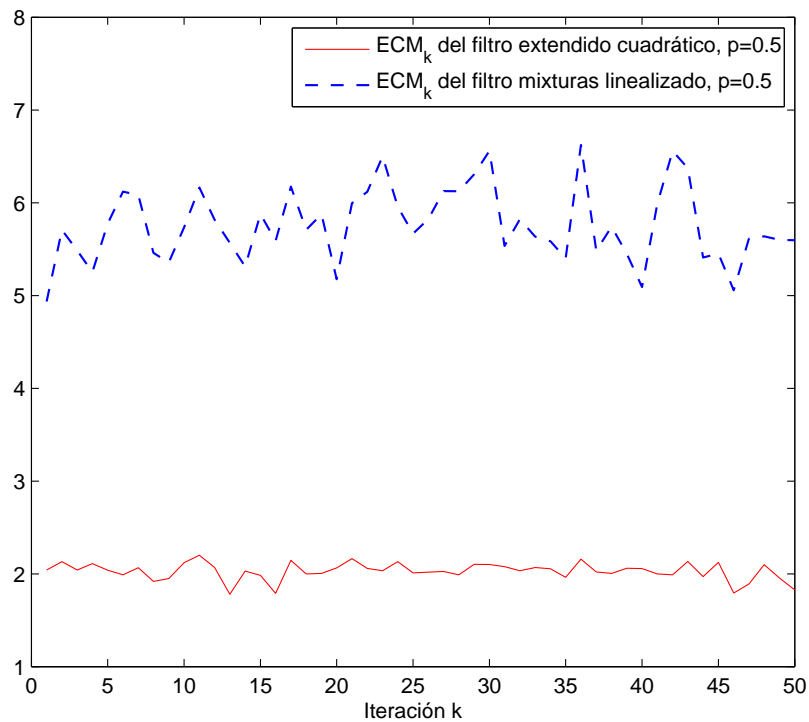


Figura 3.1: Error cuadrático medio del filtro mixturas linealizado y del filtro extendido cuadrático.

Como era de esperar, estas figuras muestran que, a medida que aumenta el valor de  $p$ , y por tanto disminuye la probabilidad de que las observaciones utilizadas en la estimación carezcan de información sobre la señal, los errores cuadráticos medios del filtro extendido cuadrático van disminuyendo de manera significativa.

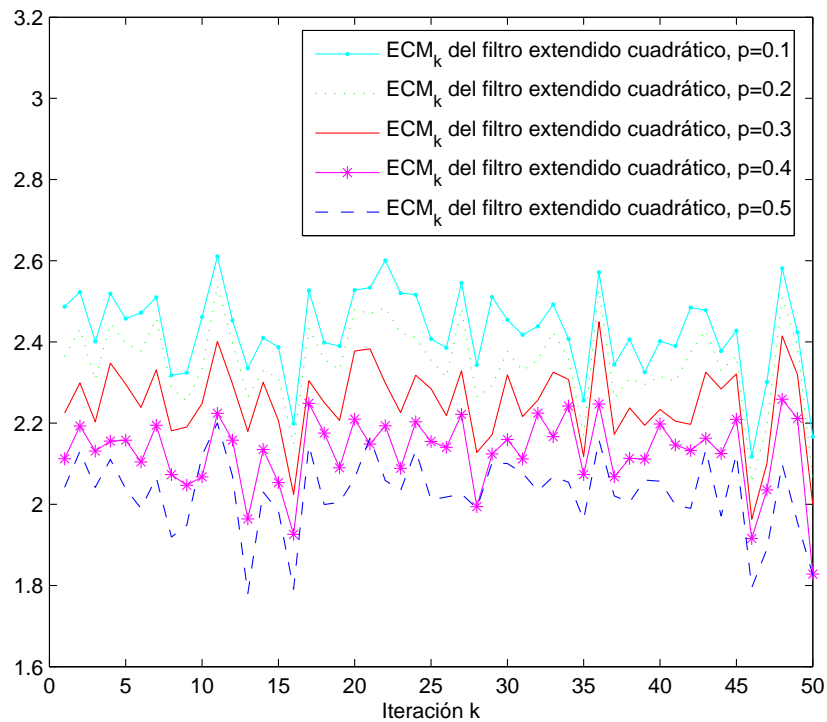


Figura 3.2: Error cuadrático medio del filtro extendido cuadrático para  $p = 0.1, 0.2, 0.3, 0.4, 0.5$

Finalmente, comparando estos resultados con los mostrados en la Figura 3.1 relativos a la aplicación del filtro linealizado mixturas, podemos también observar que, incluso para el caso más desfavorable en el sentido de mayores errores cuadráticos medios, correspondiente a  $p = 0.1$ , los resultados obtenidos con el filtro extendido cuadrático mejoran a los del filtro mixturas, aun para una mayor probabilidad de falsa alarma. Esto vuelve a reafirmar la mayor eficacia del filtro propuesto en este capítulo frente al propuesto en [25].

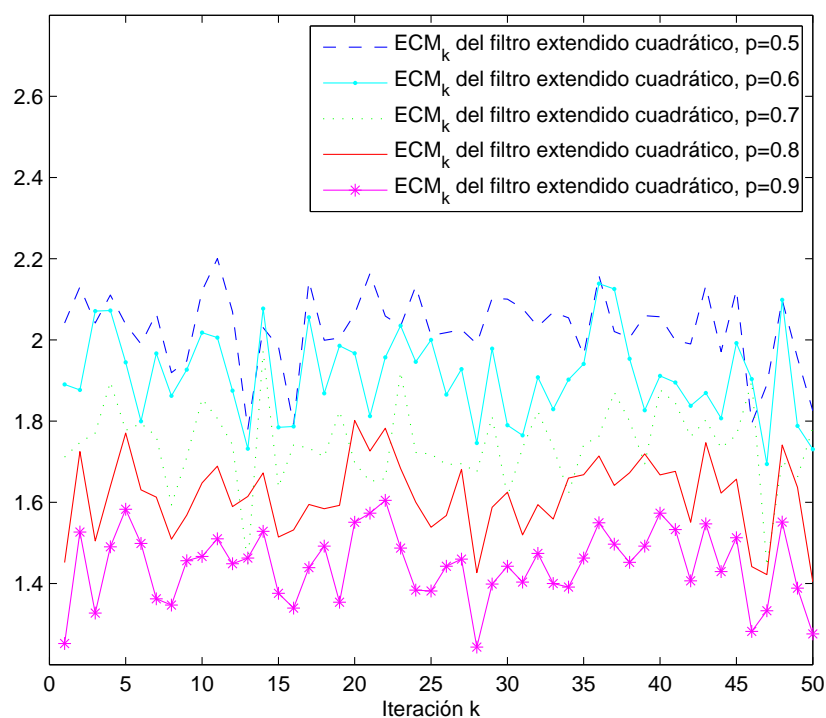


Figura 3.3: Error cuadrático medio del filtro extendido cuadrático para  $p = 0.5, 0.6, 0.7, 0.8, 0.9$ .

### 3.7.2. Actuación de los filtros ensemble

La implementación de los algoritmos de filtrado ensemble y su aplicación al sistema no lineal considerado en este ejemplo han dado lugar a los resultados que mostramos y comentamos a continuación. Los algoritmos se han implementado considerando  $q = 500$  simulaciones la señal en el instante inicial, y  $q = 500$  simulaciones de los ruidos pertinentes en cada iteración, tanto en el filtro ensemble lineal como en el cuadrático.

En la Figura 3.4, se muestran los errores cuadráticos medios asociados a 1000 simulaciones del filtro ensemble desarrollado en la Sección 3.5 y del filtro mixturas linealizado,

considerando una probabilidad de  $p = 0.5$  de que la señal en cada instante esté presente en las observaciones. Debemos indicar que la comparación se ha realizado para distintos valores de la probabilidad de falsa alarma, obteniéndose para todos ellos resultados similares, lo que nos induce a concluir una clara superioridad en la efectividad del filtro ensemble propuesto frente al linealizado.

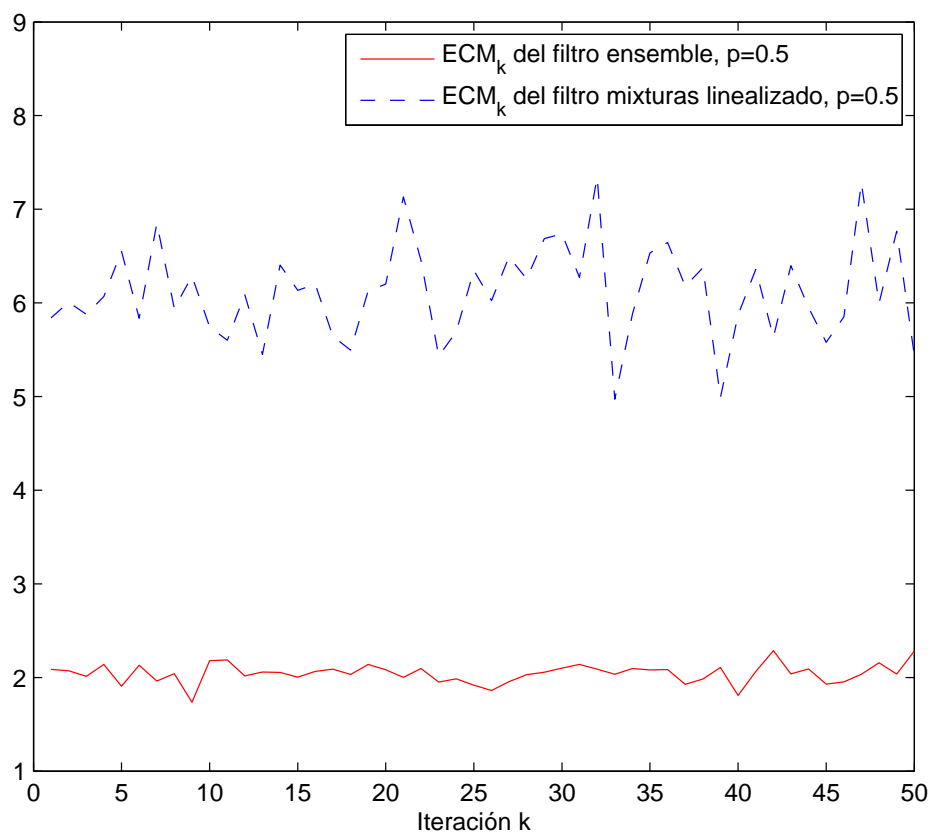


Figura 3.4: Error cuadrático medio del filtro ensemble y del filtro mixturas linealizado.

En la Figura 3.5 se representan los errores cuadráticos medios asociados a las estimaciones de la señal proporcionadas por el filtro ensemble para distintos valores de  $p$ ; concretamente,  $p = 0.1$ ,  $p = 0.3$ ,  $p = 0.5$ ,  $p = 0.7$  y  $p = 0.9$ . En esta figura se muestra el hecho lógico de que los errores cuadráticos medios de las estimaciones disminuyen de manera global a medida que aumenta  $p$  o, equivalentemente, a medida que disminuye la probabilidad de que las observaciones sean sólo ruido.

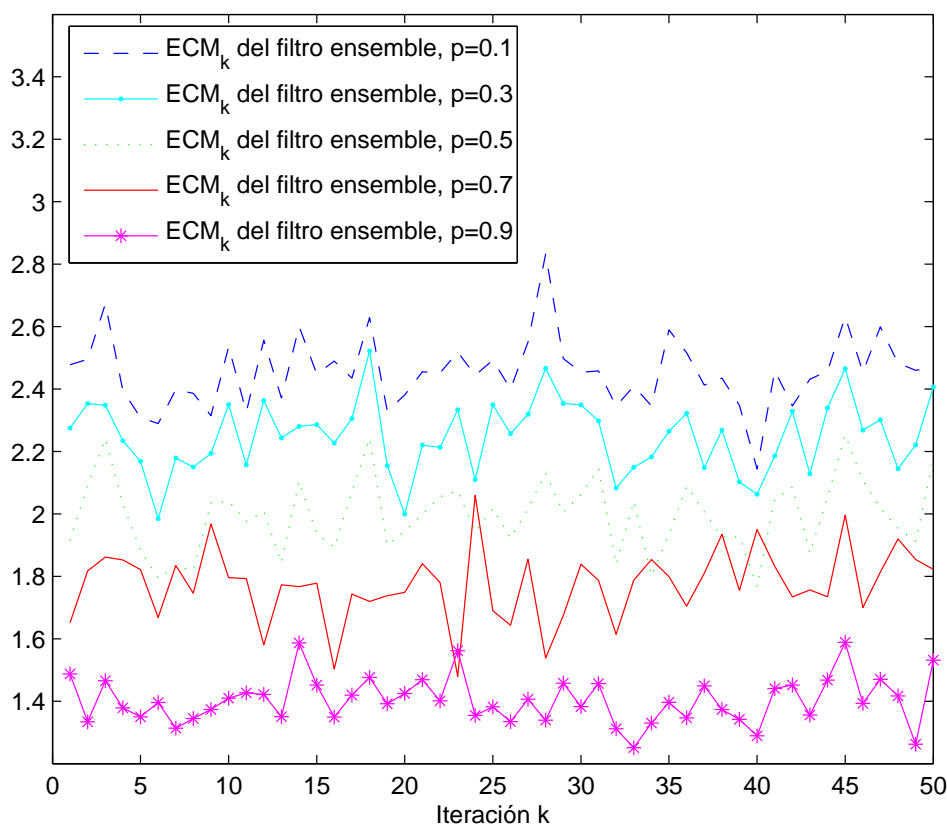


Figura 3.5: Error cuadrático medio del filtro ensemble para  $p = 0.1, 0.3, 0.5, 0.7, 0.9$ .

Seguidamente, en la Figura 3.6, se muestran los resultados obtenidos al comparar el filtro ensemble lineal con el filtro ensemble cuadrático considerando  $p = 0.5$ .

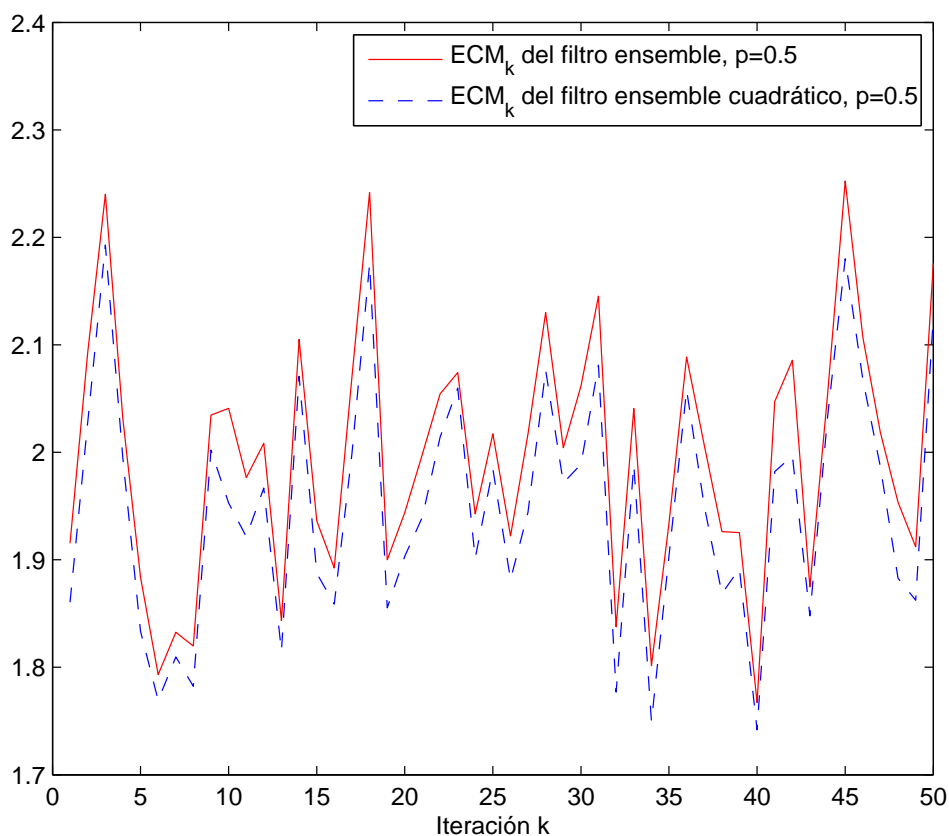


Figura 3.6: Error cuadrático medio del filtro ensemble y del filtro ensemble cuadrático.

En este caso, la comparación muestra, como era de esperar, una disminución en los errores cuadráticos medios asociados a las estimaciones del filtro cuadrático pero, sin embargo, la disminución es menos importante que la observada en otras comparaciones entre filtros lineales y cuadráticos (por ejemplo, en la Figura 2.1, en la que se muestra la comparación entre los filtros extendido lineal y cuadrático en un sistema con certidumbre, y la 3.1, en la se comparan el filtro linealizado mixturas con el filtro extendido cuadrático

considerando incertidumbre en las observaciones). Aun así, es evidente que la precisión en las estimaciones de la señal mediante la metodología de filtrado ensemble aumenta si estimamos a partir de las observaciones disponibles y de sus potencias Kronecker de segundo grado.

En la Figura 3.7 se muestran los errores cuadráticos medios asociados al filtro ensemble cuadrático para distintos valores de  $p$ , comprobándose de nuevo una mayor precisión de las estimaciones a medida que disminuye la incertidumbre en las observaciones.

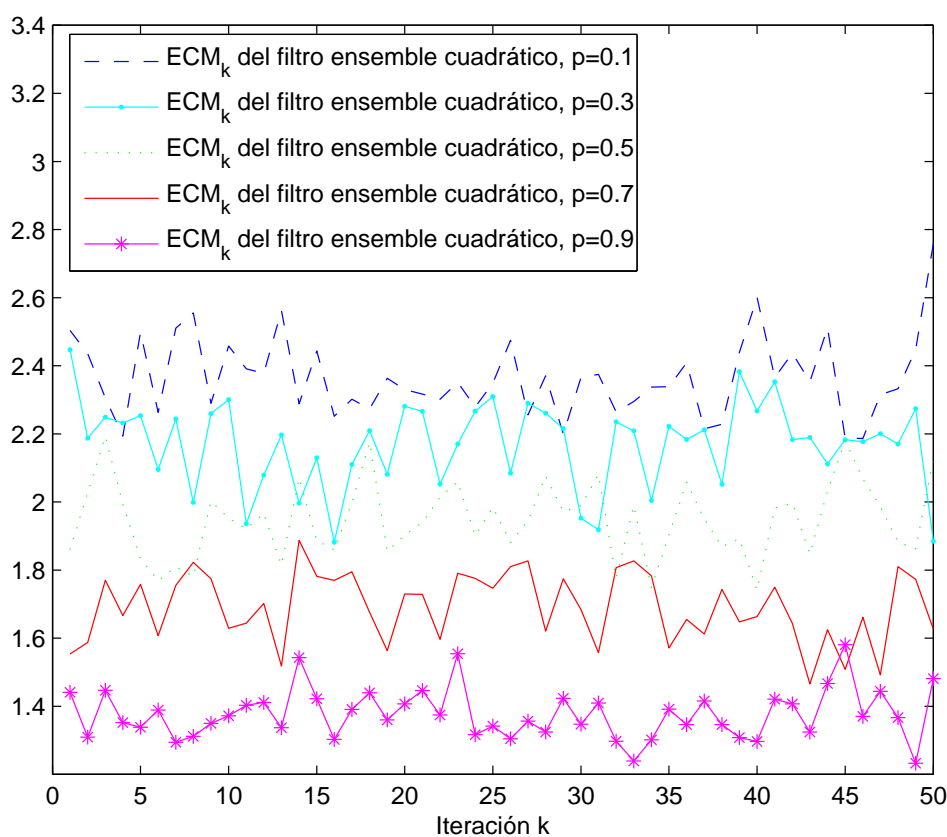


Figura 3.7: Error cuadrático medio del filtro ensemble cuadrático para  $p = 0.1, 0.3, 0.5, 0.7, 0.9$ .



Para finalizar, hemos realizado una comparación entre el filtro extendido cuadrático y el filtro ensemble cuadrático considerando, como en otras comparaciones, diferentes valores de  $p$ . Los resultados obtenidos son similares, por lo que nos limitamos a presentar, como representativo de todos los demás, los correspondientes a  $p = 0.5$ , que se muestran en la Figura 3.8. Como puede observarse en esta figura, los errores cuadráticos medios del filtro ensemble cuadrático son, globalmente, menores que los del filtro extendido cuadrático, demostrando así la mayor efectividad del primero frente al segundo.

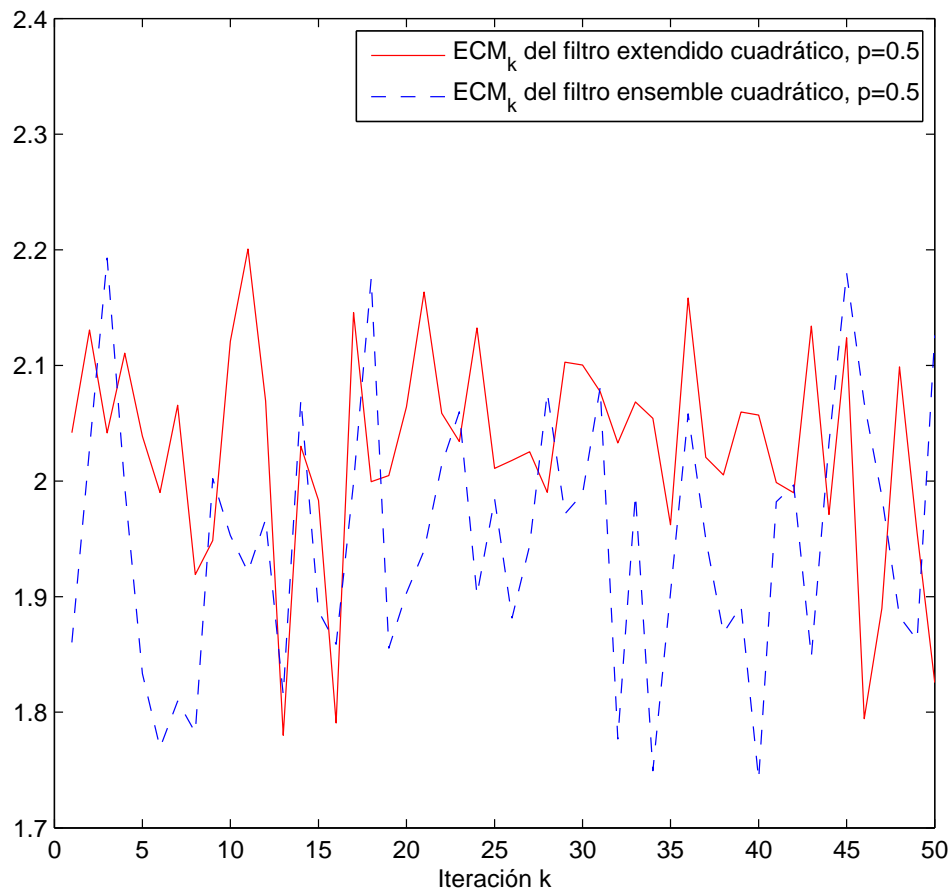


Figura 3.8: Error cuadrático medio del filtro extendido cuadrático y del filtro ensemble cuadrático.

Debemos indicar, finalmente, que, como puede comprobarse en los códigos fuente que se adjuntan, el filtro ensemble cuadrático, además de mejorar las estimaciones obtenidas con el filtro extendido cuadrático, presenta la ventaja adicional de ser computacionalmente menos complejo, disminuyendo sensiblemente el coste computacional del extendido.



# Conclusiones y Futuras Líneas de Investigación

El objetivo de este trabajo ha sido avanzar en el estudio del problema de estimación de señales ruidosas modelizadas por una dinámica no lineal a partir de la información proporcionada por la observación de funciones de la misma, también no lineales, y también afectadas por ruido.

El estudio se ha enfocado fundamentalmente al problema de estimación a partir de *observaciones inciertas*, en las que existe una probabilidad positiva de que carezcan de información sobre la señal y, por tanto, sólo aportan ruido a la estimación. Con este objetivo, los métodos de estimación abordados en nuestro estudio se han desarrollado en primer lugar en sistemas con certidumbre en las observaciones (Capítulo 2), para después adaptarlos al caso de incertidumbre (Capítulo 3).

En líneas generales, nuestro estudio ha ido especialmente dirigido a la obtención, mediante diversas metodologías, de algoritmos que proporcionan de manera recursiva estimadores basados en funciones polinomiales de segundo grado de las observaciones, definidas mediante productos de Kronecker, lo que denominamos *estimadores cuadráticos*. En general, el procedimiento usado para la obtención de tales estimadores es aumentar los vectores señal y observación originales con sus potencias de segundo orden, obteniendo así un nuevo sistema en el que se aplican algoritmos de filtrado lineal para obtener los

algoritmos de filtrado cuadrático en los sistemas originales.

El criterio de optimalidad subyacente en todo nuestro estudio es el minimizar el error cuadrático medio y, en consecuencia, la eficacia de los algoritmos propuestos se compara en términos de los errores cuadráticos medios asociados a las estimaciones obtenidas con los mismos.

Seguidamente hacemos un breve resumen del desarrollo del trabajo y de las principales aportaciones, para finalizar exponiendo algunas líneas abiertas para nuestra futura investigación.

En el **Capítulo 1** se presentan diversas metodologías que se aplican de manera usual para abordar el problema de estimación en sistemas no lineales, la del *filtro de Kalman extendido* y derivados, basadas en la aproximación de los sistemas originales mediante sistemas lineales, y la del *filtro de Kalman ensemble*, basada en la simulación de los procesos que generan la señal y las observaciones. Estas metodologías constituyen la base para el trabajo que desarrollamos en los siguientes capítulos.

El estudio realizado en el **Capítulo 2** se refiere al problema de estimación en sistemas no lineales en los que la señal que se desea estimar está siempre presente en las observaciones, *sistemas no lineales con certidumbre en las observaciones*. Basándonos en la metodología del filtro de Kalman extendido, y con objeto de mejorar las estimaciones proporcionadas por el mismo, en este capítulo proponemos el uso de estimadores cuadráticos, desarrollando un *filtro extendido cuadrático* que, como mostramos en un ejemplo de simulación, puede mejorar considerablemente las estimaciones basadas en funciones lineales de las observaciones.

Las estimaciones proporcionadas por el filtro extendido cuadrático pueden, a su vez, mejorarse si se realizan sucesivas iteraciones en la etapa de actualización de los estimadores

y, con esta idea, hemos desarrollado un *filtro extendido cuadrático iterado*. Finalmente, hemos deducido también un *filtro extendido cuadrático de segundo orden*, que mejora al extendido cuadrático al considerar una mejor aproximación lineal del sistema no lineal original.

En el **Capítulo 3** se han considerado *sistemas no lineales con observaciones inciertas*, en los que la incertidumbre está modelizada por un ruido multiplicativo en la ecuación de observación, descrito por variables de Bernoulli independientes. En este capítulo proponemos tres algoritmos recursivos para abordar el problema de estimación en estos sistemas, que comentamos a continuación.

En primer lugar, proponemos un *filtro extendido cuadrático*, como extensión del correspondiente en el caso de certidumbre. Hemos de indicar que la obtención de este algoritmo ha requerido desarrollar un filtro lineal para sistemas con observaciones inciertas en los que el ruido de la observación no es blanco, como en el filtro convencional de Nahi, lo que constituye por sí mismo una nueva aportación al problema de estimación lineal a partir de observaciones inciertas.

Seguidamente, aplicando la metodología de filtrado ensemble a sistemas con observaciones inciertas, proponemos el *filtro ensemble* y el *filtro ensemble cuadrático*. La presencia del ruido multiplicativo en la ecuación de la observación, además del ruido aditivo, hace necesaria la adaptación del filtro ensemble convencional para su aplicación a sistemas con observaciones inciertas. Por tanto, en una primera etapa, hemos realizado esta adaptación, obteniendo un filtro ensemble basado en las observaciones disponibles para la estimación, filtro que posteriormente hemos generalizado, con la misma metodología, pero basando la estimación en las observaciones y sus potencias de segundo grado.

Los distintos algoritmos desarrollados en esta investigación han sido implementados

en MATLAB considerando un modelo no lineal unidimensional concreto, con el fin de comparar el comportamiento de los estimadores propuestos. Como hemos indicado, para realizar estas comparaciones se han utilizado los errores cuadráticos medios asociados a las estimaciones obtenidas en diferentes simulaciones.

En el caso de certidumbre en las observaciones se concluye que el filtro extendido cuadrático mejora las estimaciones de la señal obtenidas por los filtros extendidos lineales y ensemble presentados en el primer capítulo. Por otro lado, como cabía esperar, el filtro extendido cuadrático iterado y el filtro extendido cuadrático de segundo orden mejoran el comportamiento del filtro extendido cuadrático, siendo la mejora del primero la más evidente.

En el caso de sistemas con observaciones inciertas, se han comparado los estimadores propuestos con el filtro mixturas linealizado propuesto en [25], y todos presentan mayor eficacia que éste. Además, los errores cuadráticos medios asociados a las estimaciones proporcionadas por el filtro ensemble cuadrático son menores que los del filtro ensemble, demostrando así que la precisión en las estimaciones de la señal aumenta si consideramos las potencias segundo orden de las observaciones disponibles. Por último, se han comparado los estimadores cuadráticos propuestos para este tipo de sistemas, concluyendo que el filtro ensemble cuadrático presenta un mejor comportamiento.

El trabajo desarrollado nos induce a plantearnos algunas extensiones, más o menos inmediatas, de los algoritmos propuestos en este trabajo.

- *Filtro extendido cuadrático iterado y filtro extendido cuadrático de segundo orden en sistemas con observaciones inciertas.*

Al igual que en el caso de certidumbre en las observaciones, es de esperar que

las estimaciones obtenidas mediante el filtro extendido cuadrático en sistemas con observaciones inciertas puedan mejorarse aplicando la metodología del filtro iterado, así como, por otra parte, considerando aproximaciones lineales más precisas de los sistemas no lineales bajo consideración. En este sentido, un objetivo inmediato en nuestro futuro estudio es la obtención de un *filtro extendido cuadrático iterado* y de un *filtro extendido cuadrático de segundo orden* para sistemas con observaciones inciertas.

- *Filtro ensemble mixtura para sistemas con observaciones inciertas.*

La extensión de la metodología de filtrado ensemble a sistemas con observaciones inciertas requiere, además de la simulación de los ruidos aditivos de las ecuaciones del estado y de la observación, la simulación del ruido multiplicativo que modeliza la incertidumbre. Sin embargo, esto podría evitarse si se tiene en cuenta que, según la forma mixtura de la función de densidad de las observaciones cuando éstas están afectadas por ruido multiplicativo, sus momentos de primer y segundo orden pueden expresarse como:

$$E[y_k] = p_k E[h_k(x_k)], \quad k \geq 1$$

$$Cov(y_k) = p_k Cov(h_k(x_k)) + p_k (1 - p_k) E[h_k(x_k)] E[h_k^T(x_k)] + R_k, \quad k \geq 1$$

$$Cov(x_k, y_k) = p_k Cov(x_k, h_k(x_k)), \quad k \geq 1.$$

Por tanto, una forma alternativa de proceder sería aproximar los momentos de primer y segundo orden de  $h_k(x_k)$  siguiendo la metodología del filtro ensemble y, a partir de ellos, aproximar los de la observación. Presumimos que al evitar la simulación del ruido multiplicativo, la disminución del número de simulaciones en cada etapa del algoritmo conlleve una mayor precisión de los estimadores obtenidos.



- *Filtros sigma-puntos para sistemas con observaciones inciertas.*

El filtro ensemble y el filtro ensemble cuadrático son métodos de estimación de señales en sistemas no lineales que utilizan técnicas basadas en la simulación de valores de la señal inicial y los procesos ruido que intervienen en el sistema. Un método alternativo interesante es sustituir la generación aleatoria de tales valores por una determinística, de manera que los puntos muestrales, denominados *sigma-puntos*, capturan los momentos de primer y segundo orden de las distribuciones correspondientes. Esta metodología ha sido aplicada con éxito en diversas situaciones [12], considerando estimadores basados en las observaciones disponibles, por lo que pensamos que podría ser interesante abordar la posibilidad de obtener algoritmos de filtrado cuadrático basados en la misma. El punto de partida sería, como es natural, readaptar el conjunto de puntos muestrales utilizado, de manera que se consiga capturar los momentos hasta el cuarto orden, necesarios para abordar el problema de estimación cuadrática.

Debemos indicar, finalmente, que los algoritmos desarrollados en todo nuestro trabajo se aplican en sistemas no lineales, bajo las hipótesis básicas de independencia de la señal inicial y de los ruidos que los definen. Obviamente, no son aplicables a sistemas en los que tales hipótesis no sean satisfechas, pero constituyen el punto de partida para el tratamiento del problema de estimación cuadrática en sistemas no lineales más generales.

# Bibliografía

- [1] Anderson, B. y Moore, J. (1979), *Optimal filtering*. Prentice Hall, Englewood Cliffs, New Jersey.
- [2] Bertino, L., Evensen, G., y Wackernagel, H. (2003), *Sequential data assimilation techniques in oceanography*. International Statistical Review, 71 , 223–241.
- [3] Brammer, K. y Siffing, G. (1989), *Kalman-Bucy Filters*. Artech House, Norwood.
- [4] Caballero, R, Hermoso, A. y Linares, J. (2003), *Polynomial filtering with uncertain observations in stochastic linear systems*. International Journal of Modelling and Simulation, 23, 22-28.
- [5] Caines, P.E. (1988), *Linear Stochastic Systems*. John Wiley and Sons, New York.
- [6] Carravetta, F., Germani, A. y Raimondi, N. (1996), *Polynomial filtering for linear discrete time non-gaussian systems*. SIAM Journal on Control and Optimization, 34 (5), 1666-1690.
- [7] Carravetta, F., Germani, A. y Raimondi, N. (1997), *Polynomial filtering of discrete-time stochastic linear systems with multiplicative state noise*. IEEE Transactions on Automatic Control AC-42 (8), 1106-1126.

- 
- [8] Catlin, D.E. (1989), *Estimation, Control and the Discrete Kalman Filter*. Springer-Verlag, New York.
- [9] Chow, B.S. y Birkemeier, W.P. (1989), *A new structure of recursive estimator*. IEEE Transactions on Automatic Control, AC- 34, 5, 568-572.
- [10] Chow, B.S. y Birkemeier, W.P. (1990), *A new recursive filter for systems with multiplicative noise*. IEEE Transactions on Automatic Control, AC-36, 6, 1430-1435.
- [11] Dalla Mora, M., Germani, A. y Raimondi, M. (2001), *Restoration of images corrupted by additive non-Gaussian noise*. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 48, 859-875.
- [12] Date, P., Mamon, R. y Jalen, L. (2008), *A new algorithm for latent state estimation in non-linear time series models*. Applied Mathematics and Computation, 203, 224-232.
- [13] Evensen, G. (1994a), *Sequential data assimilation with nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics*. J. Geophys Res 99(C5): 10 143-10 162.
- [14] Evensen, G. (1994b), *Inverse methods and data assimilation in nonlinear ocean models*. Physica D, 77, 108-129.
- [15] Evensen, G. (2003), *The Ensemble Kalman Filter: theoretical formulation and practical implementation*. Ocean Dynamics 53: 343-367.
- [16] Evensen, G. (2007), *Data Assimilation. The Ensemble Kalman Filter*. Springer-Verlag Berlin Heidelberg.
- [17] Evensen, G. y van Leeuwen, P.J. (1996), *Data assimilation and inverse methods in terms of a probabilistic formulation*. Mon. Weather Rev., 124, 2898-2913.

- 
- [18] Evensen, G. y van Leeuwen, P.J. (2000), *An ensemble Kalman smoother for nonlinear dynamics*. Mon. Weather Rev., 128, 1852-1867.
- [19] García-Ligero, M.J., Hermoso, A. y Linares, J. (1997), *Second order polynomial filtering for discrete systems with uncertain observation*. Proceedings of VIII International Symposium on Applied Stochastic Models and Data Analysis, 157-162.
- [20] Grewal, M.S. y Andrews, A.P. (1993), *Kalman Filtering: Theory and Practice*. Prentice Hall, Englewood Cliffs, New Jersey.
- [21] Grewal, M.S. y Andrews, A.P. (2001), *Kalman Filtering: Theory and Practice Using MATLAB*. John Wiley and Sons.
- [22] Hadidi, M.T. y Schwartz, S.C. (1979), *Linear recursive state estimators under uncertain observations*. IEEE Transactions on Automatic Control, AC-24, 6, 944-948.
- [23] Hermoso, A. y Linares, J. (1994), *Linear Estimation for discrete-time systems in the presence of time-correlated disturbances and uncertain observations*. IEEE Transactions on Automatic Control, AC-39, 8, 1636-1638.
- [24] Hermoso, A. y Linares, J. (1995), *Linear smoothing for discrete-time systems in the presence of correlated disturbances and uncertain observations*. IEEE Transactions on Automatic Control, 40, 1486-1488.
- [25] Hermoso, A. y Linares, J. (2007), *Different approaches for state filtering in nonlinear systems with uncertain observations*. Applied Mathematics and Computation, 187, 708-724.
- [26] Hermoso-Carazo, A., Linares-Pérez, J. y Valenzuela-Ruiz, S. (2009), *Estimación de segundo grado en sistemas no lineales discretos*. Actas del XXXI Congreso Nacio-

- nal de Estadística e Investigación Operativa y V Jornadas de Estadísticas Públicas. Murcia. 130-(1-6CD).
- [27] Hermoso-Carazo, A., Linares-Pérez, J. y Valenzuela-Ruiz, S. (2010), *Filtro extendido de segundo orden en sistemas no lineales con observaciones inciertas*. Programa Comunicaciones del XXXII Congreso Nacional de Estadística e Investigación Operativa y VI Jornadas de Estadísticas Públicas. A Coruña. ISBN: 978-84-693-6152-8, 161.
- [28] Jaffer, A.G. y Gupta, S.C. (1971), *Recursive bayesian estimation with uncertain observation*. IEEE Transaction on Information Theory, IT-17, 614-616.
- [29] Jazwinski, A. (1969), *Adaptative filtering*. Automatica, Vol.5, 475-485.
- [30] Jazwinski, A. (1970), *Stochastic Processes and Filtering Theory*. Academic Press.
- [31] Kailath, T. (1968), *An Innovations Approach to Least-Squares Estimation. PART I: Linear Filtering in Additive Noise*. IEEE Transactions on Automatic Control, Vol. AC-13 (6), 646-655.
- [32] Kailath, T. (1974), *A view of three decades of linear filtering theory*. IEEE Transactions on Information Theory, Vol. IT-20 (2), 146-181.
- [33] Kailath, T. y Frost, P. (1968), *An innovation approach to least squares estimation. PART II: Linear smoothing in additive white noise*. IEEE Transactions on Automatic Control, AC-13, 655-660.
- [34] Kalman, R.E. (1960), *A new approach to llinear filtering and prediction problems*, *Transactions of the ASME*. Journal of Basic Engineering, D-82, 35-45.
- [35] Korbicz, J. y Bidyuk, P. (1993), *State and Parameter Estimation. Digital and Optimal Filtering, Applications*. Higher College of Engineering Press, Poland.

- 
- [36] Li J. y Xiu D. (2008) *On numerical properties of the ensemble Kalman filter for data assimilation*. Comput. Methods Appl. Mech. Engrg. 197: 3574-3583.
- [37] Magnus J.R., Neudecker H. (1999), *Matrix Differential Calculus with Applications in Statistics and Econometrics*. Wiley and Sons, New York.
- [38] Meditch, J.S. (1967), *Orthogonal projection and discrete optimal linear smoothing*. SIAM Journal on Control, 5 (1), 74-89.
- [39] Meditch, J.S. (1969), *Stochastic Optimal Linear Estimation and Control*. McGraw-Hill, New York.
- [40] Monzingo, R.A. (1975), *Discrete optimal linear smoothing for systems with uncertain observations*. IEEE Transactions on Information Theory. IT-21 (3), 271-275.
- [41] Monzingo, R.A. (1981), *Discrete linear recursive smoothing for systems with uncertain observations*. IEEE Transactions on Automatic Control, AC-26 (3), 754-757.
- [42] Nahi, N.E. (1969), *Optimal Recursive Estimation with Uncertain Observation*. IEEE Transaction on Information Theory, Vol. IT-15, 3, pp.462-468.
- [43] NaNakara, W. y Yaz, E.E. (1997), *Recursive estimator for linear and nonlinear systems with uncertain observations*. Signal Processing, 62, 215-228.
- [44] Nakamori, S., Caballero-Águila, R., Hermoso-Carazo, A. y Linares-Pérez, J. (2003), *Second-order polynomial estimator from uncertain observations using covariance information*. Applied Mathematics and Computation 143, 319-338.
- [45] Nakamori, S., Caballero-Águila, R., Hermoso-Carazo, A. y Linares-Pérez, J. (2004), *Quadratic estimation from uncertain observations with white plus coloured noises using covariance information*. Applied Mathematics and Computation 155, 65-79.

- 
- [46] Nakamori, S., Hermoso-Carazo, A. y Linares-Pérez, J. (2008), *Design of Quadratic Estimators Using Covariance Information in Linear Discrete-Time Stochastic Systems*. Journal of Time Series Analysis, Vol. 29, (3), pp. 501-512.
- [47] Nakamori, S., Caballero-Águila, R., Hermoso-Carazo, A., Jiménez-Lopez, J.D. y Linares-Pérez, J. (2009), *Signal estimation with nonlinear uncertain observations using covariance information*. Journal of Statistical Computation and Simulation 79 (1), 55-66.
- [48] Rajasekaran, P.K., Satyanarayana, N. y Srinath, M.D. (1971), *Optimum linear estimation of stochastic signal in the presence of multiplicative noise*. IEEE Transactions on Aerospace and Electronic Systems, AES-7, 3, 462-468.
- [49] De Santis, A., Germani, A. y Raimondi, M. (1995), *Optimal Quadratic Filtering of Linear Discrete Time Non-Gaussian Systems*. IEEE Transactions on Automatic Control. Vol. 40 (7), 1274-1278.
- [50] Schmidt, S.F. (1970), *Computational techniques in Kalman filtering*. AGARDograph 139, NATO Advisory Group for Aerospace Research and Development.
- [51] Simon, D. (2006), *Optimal state estimation: Kalman,  $H_\infty$  and Approaches*. John and Sons.
- [52] Sorenson, H.W. (1970), *Least squares estimation: from Gauss to Kalman*. IEEE Spectrum, 7, 63-68.
- [53] Tanizaki, H. (1996), *Nonlinear filters: estimation and applications*. Springer.
- [54] Tugnait, J.K. (1981), *Asymptotic stability of the MMSE linear filter for systems with uncertain observations*. IEEE Transactions on Information Theory, IT-27, 2, 247-250.

- 
- [55] Tugnait, J.K. (1981), *Stability of optimum linear estimators of stochastic signals in white multiplicative noise*. IEEE Transactions on Automatic Control, AC-26, 757-761.
- [56] Wang, X. (1984), *Recursive algorithms for linear LMSE estimators under uncertain observations*. IEEE Transactions on Automatic Control, AC-29, 9, 853-854.
- [57] Uppala, S.V. y Sahr, J.D. (1997) *On the design of quadratic filters with applications to image processing*. IEEE Transactions on Image Processing 6, 608-614.
- [58] Wishner, R.P., Tabaczynski, J.A. y Athams, M. (1969), *A comparison of three Non-Linear filters*. Automatica, Vol. 5, 487-496.
- [59] Zheng, D.Q., Leung, J.K.C y Lee, B.Y (2010), *An ensemble Kalman filter for atmospheric data assimilation: Application to wind tunnel data*. Atmospheric Environment 1-7.





# Códigos fuente de MATLAB



## CAPÍTULO 1: Figuras 1.1, 1.2, 1.3 y 1.4

```

%Modelo de la señal:      x(k+1)=1/(x(k)^2+3)+w(k)   donde w(k)-->N(0,1)
%                          x0-->N(0,1)
%                          f(x)=1/(x^2+3)

%Observaciones:         y(k)=x(k)^2+exp{x(k)}+v(k)   donde v(k)-->N(0,1)
%                          h(x)=x^2+exp{x}

n_simul=1000; %número de simulaciones
n_iter=50;    %número de iteraciones

for s=1:n_simul

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%% Generación de la señal x(k) %%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    P0=1; %varianza de la condición inicial
    Q=1;  %varianza del ruido de la ecuación de la señal

    %% Condiciones iniciales %%

    x0(s)=P0^(1/2)*randn(1,1);
    w0(s)=Q^(1/2)*randn(1,1);

    x(s,1)=1/(x0(s)^2+3)+w0(s);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    for k=1:n_iter
        w(s,k)=Q^(1/2)*randn(1,1);
        x(s,k+1)=1/(x(s,k)^2+3)+w(s,k);
    end

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%% Generación de las observaciones y(k) %%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    R=1; %varianza del ruido de la ecuación observación

    for k=1:n_iter
        v(s,k)=R^(1/2)*randn(1,1);
        y(s,k)=x(s,k)^2+exp(x(s,k))+v(s,k);
    end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO DE KALMAN EXTENDIDO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fill=filtro de Kalman extendido (FKE)
%pred1=predicador de Kalman extendido (PKE)
%Cfill=varianza del FKE
%Cpred1=varianza del PKE
%H1=derivada de la función h en el PKE
%F1=derivada de la función f en el FKE
%K1=ganancia del FKE
%DC_fill=cuadrado de la diferencia entre la señal y el FKE

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%% Condiciones iniciales para el FKE %%%%%

    fill0(s)=0;
    Cfill0(s)=P0;

    pred1(s,1)=1/(fill0(s)^2+3);
    F10(s)=((-2)*fill0(s))/((fill0(s)^2+3)^2);
    Cpred1(s,1)=F10(s)*Cfill0(s)*F10(s)'+Q;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    for k=1:n_iter

        H1(s,k)=2*pred1(s,k)+exp(pred1(s,k));
        K1(s,k)=Cpred1(s,k)*H1(s,k)'*(H1(s,k)*Cpred1(s,k)*H1(s,k)'+R)^(-1);
        fill(s,k)=pred1(s,k)+K1(s,k)*(y(s,k)-...
            (pred1(s,k)^2+exp(pred1(s,k))));
        Cfill(s,k)=(1-K1(s,k)*H1(s,k))*Cpred1(s,k);

        pred1(s,k+1)=1/(fill(s,k)^2+3);
        F1(s,k)=((-2)*fill(s,k))/((fill(s,k)^2+3)^2);
        Cpred1(s,k+1)=F1(s,k)*Cfill(s,k)*F1(s,k)'+Q;

        DC_fill(s,k)=(x(s,k)-fill(s,k))^2;

    end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILTRO DE KALMAN EXTENDIDO ITERADO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fil2=filtro de Kalman extendido iterado (FKEI)
%pred2=predicador de Kalman extendido iterado (PKEI)
%Cfil2=varianza del FKEI
%Cpred2=varianza del PKEI
%F2=derivada de la función f en el FKEI
%H2=derivada de la función h en el PKEI
%K2=ganancia del FKEI
%DC_fil2=cuadrado de la diferencia entre la señal y el FKEI

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Condiciones iniciales para el FKEI %%%%%%%%%

    fil20(s)=0;
    Cfil20(s)=P0;
    pred2(s,1)=1/(fil20(s)^2+3);
    F20(s)=((-2)*fil20(s))/((fil20(s)^2+3)^2);
    Cpred2(s,1)=F20(s)*Cfil20(s)*F20(s)'+Q;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bucle de iteraciones para el cálculo del FKEI %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%filk(i)= FKEI obtenido en la iteración i del bucle
%Cfilk(i)= varianza del filk(i)
%Hk(i)=derivada de la función h en filk(i)
%Kk(i)=ganancia en la iteración i del bucle

N=20; %número de iteraciones del bucle que calcula el FKEI

for k=1:n_iter

    H2(s,k)= 2*pred2(s,k)+exp(pred2(s,k));
    K2(s,k)=Cpred2(s,k)*H2(s,k)'*(H2(s,k)*Cpred2(s,k)*H2(s,k)'+R)^(-1);

    %% Condiciones iniciales del bucle para el cálculo del FKEI %%

    filk0(s,k)=pred2(s,k)+K2(s,k)*(y(s,k)-...
        (pred2(s,k)^2+exp(pred2(s,k))));
    Cfilk0(s,k)=(1-K2(s,k)*H2(s,k))*Cpred2(s,k);

    Hk0(s,k)=2*filk0(s,k)+exp(filk0(s,k));
    Kk0(s,k)=Cpred2(s,k)*Hk0(s,k)'+...
        (Hk0(s,k)*Cpred2(s,k)*Hk0(s,k)'+R)^(-1);

```

```

filk(s,k,1)=pred2(s,k)+Kk0(s,k)*(y(s,k)-...
    (filk0(s,k)^2+exp(filk0(s,k)))-...
    Hk0(s,k)*(pred2(s,k)-filk0(s,k)));
Cfilk(s,k,1)=(1-Kk0(s,k)*Hk0(s,k))*Cpred2(s,k);

for i=1:N   %%% Bucle que calcula el FKEI %%%

    Hk(s,k,i)=2*filk(s,k,i)+exp(filk(s,k,i));
    Kk(s,k,i)=Cpred2(s,k)*Hk(s,k,i)'+...
        (Hk(s,k,i)*Cpred2(s,k)*Hk(s,k,i)'+R)^(-1);
    filk(s,k,i+1)=pred2(s,k)+Kk(s,k,i)*(y(s,k)-...
        (filk(s,k,i)^2+exp(filk(s,k,i)))-...
        Hk(s,k,i)*(pred2(s,k)-filk(s,k,i)));
    Cfilk(s,k,i+1)=(1-Kk(s,k,i)*Hk(s,k,i))*Cpred2(s,k);

end

fil2(s,k)=filk(s,k,N+1);
Cfil2(s,k)=Cfilk(s,k,N+1);
DC_fil2(s,k)=(x(s,k)-fil2(s,k))^2;

F2(s,k)=((-2)*fil2(s,k))/(fil2(s,k)^2+3)^2;
pred2(s,k+1)=1/(fil2(s,k)^2+3);
Cpred2(s,k+1)=F2(s,k)*Cfil2(s,k)*F2(s,k)'+Q;

end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILTRO DE KALMAN EXTENDIDO DE 2º ORDEN %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fil3=filtro de Kalman extendido de segundo orden (FKE20)
%pred3=predicador de Kalman extendido de segundo orden (PKE20)
%Cfil3=varianza del FKE20
%Cpred3=varianza del PKE20
%F3=derivada de la función f en el FKE20
%H3=derivada de la función h en el PKE20
%K3=ganancia del FKE20
%DC_fil3=cuadrado de la diferencia entre la señal y el FKE20

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Condiciones iniciales para el FKE20 %

fil30(s)=0;
Cfil30(s)=P0;
pred3(s,1)=1/(fil30(s)^2+3)+(1/2)*(((8*fil30(s)^2)/(fil30(s)^2+3)^2)-...
    (2/(fil30(s)^2+3)^2))*Cfil30(s);
F30=(-2)*fil30(s)/(fil30(s)^2+3)^2;
Cpred3(s,1)=F30*Cfil30(s)*F30'+Q;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    H3(s,k)=2*pred3(s,k)+exp(pred3(s,k));
    K3(s,k)=Cpred3(s,k)*H3(s,k)'+...
        (H3(s,k)*Cpred3(s,k)*H3(s,k)'+R)^(-1);
    fil3(s,k)=pred3(s,k)+K3(s,k)*(y(s,k)-pred3(s,k)^2-...
        exp(pred3(s,k))-((1/2)*(2+exp(pred3(s,k))))*...
        Cpred3(s,k));
    Cfil3(s,k)=(1-K3(s,k)*H3(s,k))*Cpred3(s,k);

    pred3(s,k+1)=1/(fil3(s,k)^2+3)+(1/2)*...
        (((8*fil3(s,k)^2)/(fil3(s,k)^2+3)^2)-...
        (2/(fil3(s,k)^2+3)^2))*Cfil3(s,k);
    F3(s,k)=((-2)*fil3(s,k))/(fil3(s,k)^2+3)^2;
    Cpred3(s,k+1)=F3(s,k)*Cfil3(s,k)*F3(s,k)'+Q;
    DC_fil3(s,k)=(x(s,k)-fil3(s,k))^2;

end %iter

```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO DE KALMAN ENSEMBLE %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fil4=filtro de Kalman ensemble (FKEn)
%pred4=predicador de Kalman ensemble (PKEn)
%Cfil4=cuasivarianza del FKEn
%Cpred4=cuasivarianza del PKEn
%K4=ganancia del FKEn
%DC_fil4=cuadrado de la diferencia entre la señal y el FKEn

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Generación del ensemble de filtrado y predicción %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%efil=conjunto de filtros
%epred=conjunto de predictores
%ew=conjunto de valores simulados a partir de la distribución de w
%ev=conjunto de valores simulados a partir de la distribución de v
%obs=conjunto de predictores de las observaciones
%yest=media de obs
%Cyy=cuasivarianza de obs
%Cxy=cuasicovarianza del PKEn y obs

q=100; %tamaño del conjunto de valores simulados

% Simulación del ensemble de filtrado y predicción inicial %

for j=1:q

    efil0(s,j)=P0^(1/2)*randn(1,1);
    ew0(s,j)=Q^(1/2)*randn(1,1);
    epred(s,1,j)= 1/(efil0(s,j)^2+3)+ew0(s,j);

end

% Cálculo del PKEn y su cuasivarianza para k=1 %%

pred4(s,1)=mean(epred(s,1,:));
Cpred4(s,1)=var(epred(s,1,:)); %función que calcula la cuasivarianza

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del FKEn y su cuasivarianza para k %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    %% Cálculo de los predictores de las observaciones en k %%

    for j=1:q

        ev(s,k,j)=R^(1/2)*randn(1,1);
        obs(s,k,j)=epred(s,k,j)^2+exp(epred(s,k,j))+ev(s,k,j);

    end

    yest(s,k)=mean(obs(s,k,:));
    Cyy(s,k)=var(obs(s,k,:));

    %% Cálculo de la cuasivarianza del PKEn y obs en k%%

    restapred(s,k,1)=epred(s,k,1)-pred4(s,k);
    restayy(s,k,1)=obs(s,k,1)-yest(s,k);
    Cxysum(s,k,1)=restapred(s,k,1)*restayy(s,k,1);

    for j=2:q

        restapred(s,k,j)=epred(s,k,j)-pred4(s,k);
        restayy(s,k,j)=obs(s,k,j)-yest(s,k);
        Cxysum(s,k,j)=Cxysum(s,k,j-1)+...
            (restapred(s,k,j)*restayy(s,k,j));

    end

    Cxy(s,k)=Cxysum(s,k,q)/(q-1);

    %% Cálculo del FKEn y su cuasivarianza en k %%

    K4(s,k)=Cxy(s,k)*Cyy(s,k)^(-1);

    for j=1:q
        efil(s,k,j)=epred(s,k,j)+K4(s,k)*(y(s,k)-obs(s,k,j));
    end

    fil4(s,k)=mean(efil(s,k,:));
    Cfil4(s,k)=var(efil(s,k,:));
    DC_fil4(s,k)=(x(s,k)-fil4(s,k))^2;

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Cálculo del PKE y su cuasivarianza para k+1 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

    for j=1:q

        ew(s,k,j)=Q^(1/2)*randn(1,1);
        epred(s,k+1,j)= 1/(efil(s,k,j)^2+3)+ew(s,k,j);

    end

    pred4(s,k+1)=mean(epred(s,k+1,:));
    Cpred4(s,k+1)=var(epred(s,k+1,:));
end %iter

end %simul

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Errores cuadráticos medios del FKE, FKEI, FKE20 y FKEN %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

for k=1:n_iter
    MSE_fil1(1,k)=DC_fil1(1,k);
    MSE_fil2(1,k)=DC_fil2(1,k);
    MSE_fil3(1,k)=DC_fil3(1,k);
    MSE_fil4(1,k)=DC_fil4(1,k);
    for s=2:n_simul
        MSE_fil1(s,k)=MSE_fil1(s-1,k)+DC_fil1(s,k);
        MSE_fil2(s,k)=MSE_fil2(s-1,k)+DC_fil2(s,k);
        MSE_fil3(s,k)=MSE_fil3(s-1,k)+DC_fil3(s,k);
        MSE_fil4(s,k)=MSE_fil4(s-1,k)+DC_fil4(s,k);
    end
    RMSE_fil1(k)=(MSE_fil1(n_simul,k)/n_simul)^(1/2);
    RMSE_fil2(k)=(MSE_fil2(n_simul,k)/n_simul)^(1/2);
    RMSE_fil3(k)=(MSE_fil3(n_simul,k)/n_simul)^(1/2);
    RMSE_fil4(k)=(MSE_fil4(n_simul,k)/n_simul)^(1/2);
end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Gráficas de los Errores cuadráticos medios %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

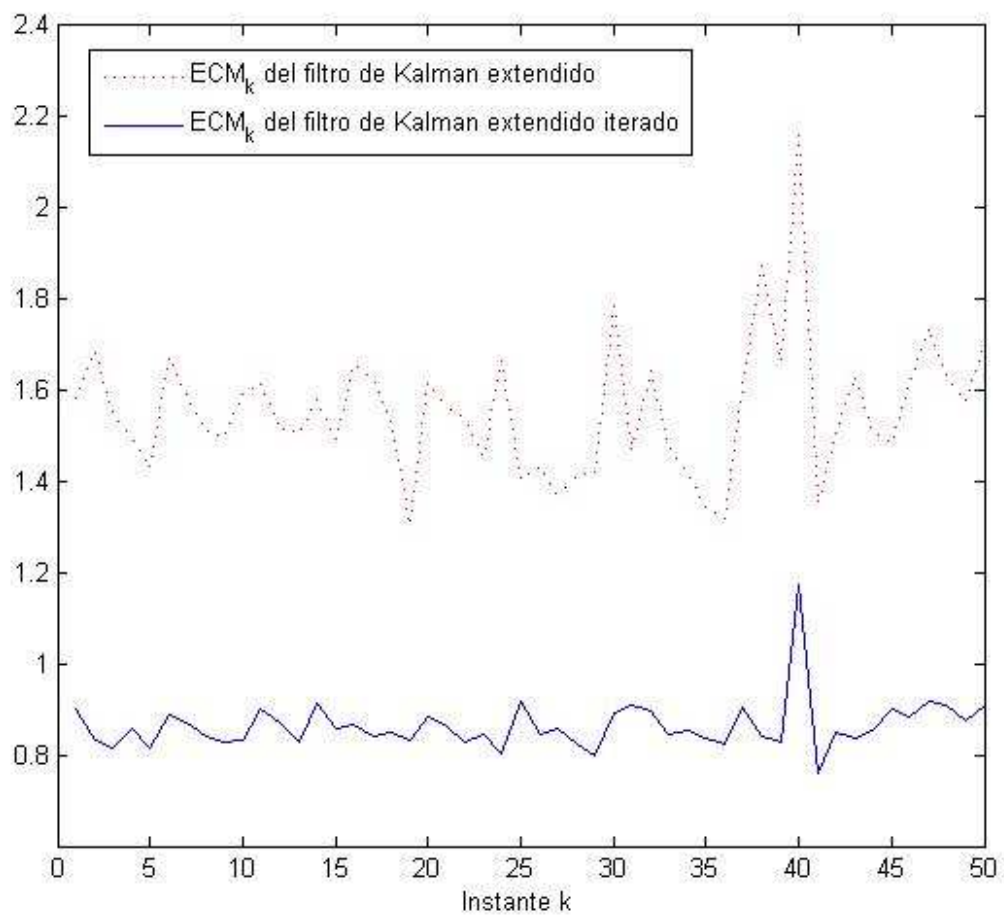
```

for k=1:n_iter
    time(k)=k;
    pRMSE_fil1(k)=RMSE_fil1(k);
    pRMSE_fil2(k)=RMSE_fil2(k);
    pRMSE_fil3(k)=RMSE_fil3(k);
    pRMSE_fil4(k)=RMSE_fil4(k);
end

```

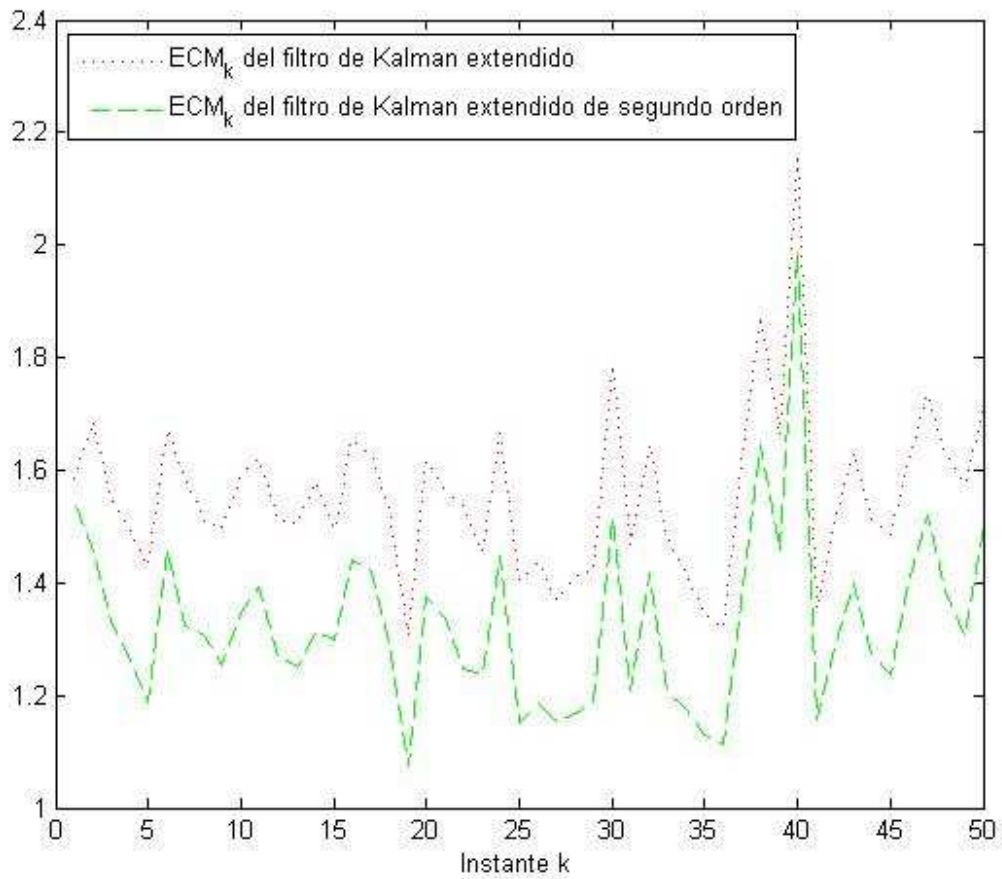
```
figure(1.1) %Error cuadrático medio del FKE y FKEI
grafic1=plot(time,pRMSE_fill,'r:',time,pRMSE_fil2,'b-');
xlabel('Instante k')
legend('ECM_k del filtro de Kalman extendido',...
       'ECM_k del filtro de Kalman extendido iterado')
saveas(gcf, 'Fig1_1', 'fig')

waitforbuttonpress;
```



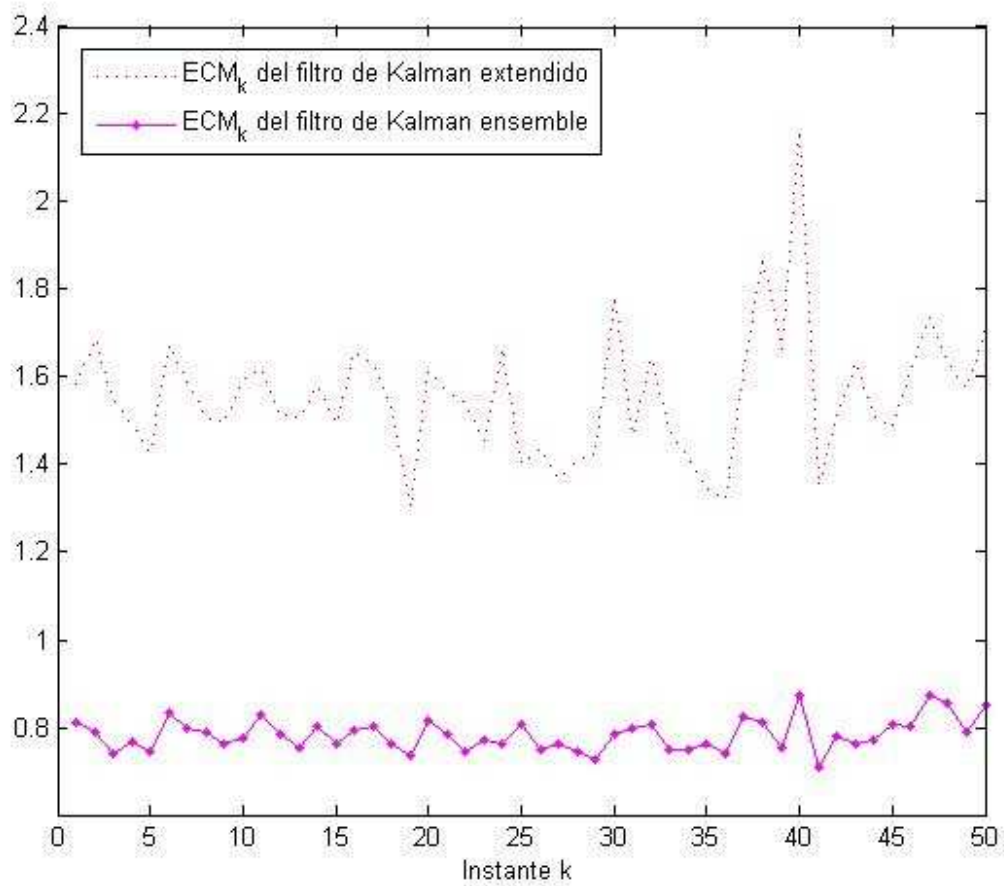
```
figure(1.2) %Error cuadrático medio del FKE y FKE20
grafic2=plot(time,pRMSE_fil1,'r:',time,pRMSE_fil3,'g--');
xlabel('Instante k')
legend('ECM_k del filtro de Kalman extendido',...
       'ECM_k del filtro de Kalman extendido de segundo orden')
saveas(gcf, 'Fig1_2', 'fig')

waitforbuttonpress;
```

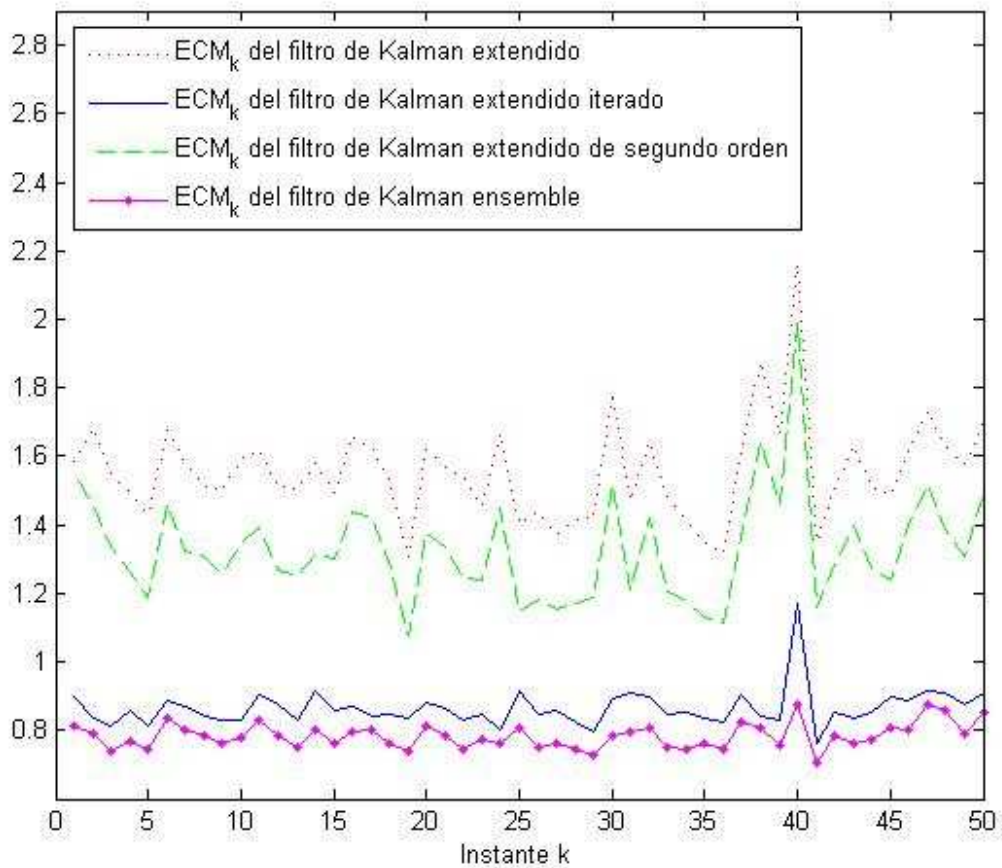


```
figure(1.3) %Error cuadrático medio del FKE y FKEn
grafic3=plot(time,pRMSE_fill,'r:',time,pRMSE_fil4,'m.-');
xlabel('Instante k')
legend('ECM_k del filtro de Kalman extendido',...
       'ECM_k del filtro de Kalman ensemble')
saveas(gcf, 'Fig1_3', 'fig')

waitforbuttonpress;
```



```
figure(1.4) %Errores cuadráticos medios del FKE, FKEI, FKE2O y FKEn
grafic4=plot(time,pRMSE_fil1,'r:',time,pRMSE_fil2,'b-',...
             time,pRMSE_fil3,'g--',time,pRMSE_fil4,'m.-');
xlabel('Instante k')
legend('ECMk del filtro de Kalman extendido',...
      'ECMk del filtro de Kalman extendido iterado',...
      'ECMk del filtro de Kalman extendido de segundo orden',...
      'ECMk del filtro de Kalman ensemble')
saveas(gcf, 'Fig1_4', 'fig')
```



## CAPÍTULO 2: Figura 2.1

```

%Modelo de la señal:      x(k+1)=1/(x(k)^2+3)+w(k)   donde w(k)-->N(0,1)
%                          x0-->N(0,1)
%                          f(x)=1/(x^2+3)

%Observaciones:         y(k)= x(k)^2+exp{x(k)}+v(k)   donde
%                          h(x)= x^2+exp{x}

%   P[v(k)=1]=15/18      P[v(k)=-3]=2/18      P[v(k)=-9]=1/18

n_simul=1000;%número de simulaciones
n_iter=50;   %número de iteraciones
N=20;       %número de iteraciones del bucle de filtrado del FKEI
q=100;      %tamaño del ensemble

for s=1:n_simul

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%% Generación de la señal x(k) %%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    P0=1;          %varianza de la señal inicial
    P30=0;         %momento centrado de tercer orden de la señal inicial
    P40=3*P0^2-P0^2; %momento centrado de cuarto orden de la señal inicial

    Q=19/3;        %varianza del ruido de la ecuación de la señal w(k)
    Q3=-128/3;     %momento centrado de tercer orden del ruido w(k)
    Q4=1123/3-Q^2; %momento centrado de cuarto orden del ruido w(k)

    %% Condiciones iniciales %%

    a=rand(1,1);
    if a <= 1/18
        w0(s)=-9;
    elseif a <= 3/18
        w0(s)=-3;
    else w0(s)=1;
    end

    x0(s)=P0^(1/2)*randn(1,1);
    x(s,1)=1/(x0(s)^2+3)+w0(s);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```



```
for k=1:n_iter

    b=rand(1,1);
    if b <= 1/18
        w(s,k)=-9;
    elseif b <= 3/18
        w(s,k)=-3;
    else w(s,k)=1;
    end
    x(s,k+1)=1/(x(s,k)^2+3)+w(s,k);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Generación de las observaciones y(k) %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

R=19/3;           %varianza del ruido de la ecuación observación v(k)
R3=-128/3;       %momento centrado de tercer orden del ruido v(k)
R4=1123/3-R^2;   %momento centrado de cuarto orden del ruido v(k)

for k=1:n_iter

    c=rand(1,1);
    if c <= 1/18
        v(s,k)=-9;
    elseif c <= 3/18
        v(s,k)=-3;
    else v(s,k)=1;
    end
    y(s,k)=x(s,k)^2+exp(x(s,k))+v(s,k);

end
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO DE KALMAN EXTENDIDO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fill=filtro de Kalman extendido (FKE)
%pred1=predicador de Kalman extendido (PKE)
%Cfill=varianza del FKE
%Cpred1=varianza del PKE
%H1=derivada de la función h en el PKE
%F1=derivada de la función f en el FKE
%K1=ganancia del FKE
%DC_fill=cuadrado de la diferencia entre la señal y el FKE

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%% Condiciones iniciales para el FKE %%%%%

    fill0(s)=0;
    Cfill0(s)=P0;

    pred1(s,1)=1/(fill0(s)^2+3);
    F10(s)=((-2)*fill0(s))/((fill0(s)^2+3)^2);
    Cpred1(s,1)=F10(s)*Cfill0(s)*F10(s)'+Q;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    for k=1:n_iter

        H1(s,k)=2*pred1(s,k)+exp(pred1(s,k));
        K1(s,k)=Cpred1(s,k)*H1(s,k)'*(H1(s,k)*Cpred1(s,k)*H1(s,k)'+R)^(-1);
        fill(s,k)=pred1(s,k)+K1(s,k)*(y(s,k)-...
            (pred1(s,k)^2+exp(pred1(s,k))));
        Cfill(s,k)=(1-K1(s,k)*H1(s,k))*Cpred1(s,k);

        pred1(s,k+1)=1/(fill(s,k)^2+3);
        F1(s,k)=((-2)*fill(s,k))/((fill(s,k)^2+3)^2);
        Cpred1(s,k+1)=F1(s,k)*Cfill(s,k)*F1(s,k)'+Q;
        DC_fill(s,k)=(x(s,k)-fill(s,k))^2;

    end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILTRO DE KALMAN EXTENDIDO ITERADO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fil2=filtro de Kalman extendido iterado (FKEI)
%pred2=predicador de Kalman extendido iterado (PKEI)
%Cfil2=varianza del FKEI
%Cpred2=varianza del PKEI
%F2=derivada de la función f en el FKEI
%H2=derivada de la función h en el PKEI
%K2=ganancia del FKEI
%DC_fil2=cuadrado de la diferencia entre la señal y el FKEI

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Condiciones iniciales para el FKEI %Condiciones

    fil20(s)=0;
    Cfil20(s)=P0;

    pred2(s,1)=1/(fil20(s)^2+3);
    F20(s)=((-2)*fil20(s))/((fil20(s)^2+3)^2);
    Cpred2(s,1)=F20(s)*Cfil20(s)*F20(s)'+Q;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bucle de iteraciones para el cálculo del FKEI %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%filk(i)= FKEI obtenido en la iteración i del bucle
%%Cfilk(i)= varianza del filk(i)
%%Hk(i)=derivada de la función h en filk(i)
%%Kk(i)=ganancia en la iteración i del bucle

N=20; %número de iteraciones del bucle que calcula el FKEI

for k=1:n_iter

    H2(s,k)= 2*pred2(s,k)+exp(pred2(s,k));
    K2(s,k)=Cpred2(s,k)*H2(s,k)'*(H2(s,k)*Cpred2(s,k)*H2(s,k)'+R)^(-1);

    %% Condiciones iniciales del bucle para el cálculo del FKEI %%

    filk0(s,k)=pred2(s,k)+K2(s,k)*(y(s,k)-...
        (pred2(s,k)^2+exp(pred2(s,k))));
    Cfilk0(s,k)=(1-K2(s,k)*H2(s,k))*Cpred2(s,k);

```

```

Hk0(s,k)=2*filk0(s,k)+exp(filk0(s,k));
Kk0(s,k)=Cpred2(s,k)*Hk0(s,k)'+...
        (Hk0(s,k)*Cpred2(s,k)*Hk0(s,k)'+R)^(-1);

filk(s,k,1)=pred2(s,k)+Kk0(s,k)*(y(s,k)-...
        (filk0(s,k)^2+exp(filk0(s,k)))-...
        Hk0(s,k)*(pred2(s,k)-filk0(s,k)));
Cfilk(s,k,1)=(1-Kk0(s,k)*Hk0(s,k))*Cpred2(s,k);

for i=1:N   %% Bucle que calcula el FKEI %%
    Hk(s,k,i)=2*filk(s,k,i)+exp(filk(s,k,i));
    Kk(s,k,i)=Cpred2(s,k)*Hk(s,k,i)'+...
        (Hk(s,k,i)*Cpred2(s,k)*Hk(s,k,i)'+R)^(-1);
    filk(s,k,i+1)=pred2(s,k)+Kk(s,k,i)*(y(s,k)-...
        (filk(s,k,i)^2+exp(filk(s,k,i)))-...
        Hk(s,k,i)*(pred2(s,k)-filk(s,k,i)));
    Cfilk(s,k,i+1)=(1-Kk(s,k,i)*Hk(s,k,i))*Cpred2(s,k);

end

fil2(s,k)=filk(s,k,N+1);
Cfil2(s,k)=Cfilk(s,k,N+1);
DC_fil2(s,k)=(x(s,k)-fil2(s,k))^2;

F2(s,k)=((-2)*fil2(s,k))/(fil2(s,k)^2+3)^2;
pred2(s,k+1)=1/(fil2(s,k)^2+3);
Cpred2(s,k+1)=F2(s,k)*Cfil2(s,k)*F2(s,k)'+Q;

end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILTRO DE KALMAN EXTENDIDO DE 2º ORDEN %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fil3=filtro de Kalman extendido de 2º orden (FKE20)
%pred3=predicador de Kalman extendido de 2º orden (PKE20)
%Cfil3=varianza del FKE20
%Cpred3=varianza del PKE20
%F3=derivada de la función f en el FKE20
%H3=derivada de la función h en el PKE20
%K3=ganancia del FKE20
%DC_fil3=cuadrado de la diferencia entre la señal y el FKE20

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Condiciones iniciales para el FKE20 %Condiciones

fil30(s)=0;
Cfil30(s)=P0;

pred3(s,1)=1/(fil30(s)^2+3)+(1/2)*(((8*fil30(s)^2)/(fil30(s)^2+3)^2)-...
    (2/(fil30(s)^2+3)^2))*Cfil30(s);
F30=(-2)*fil30(s)/(fil30(s)^2+3)^2;
Cpred3(s,1)=F30*Cfil30(s)*F30'+Q;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    H3(s,k)=2*pred3(s,k)+exp(pred3(s,k));
    K3(s,k)=Cpred3(s,k)*H3(s,k)'+...
        (H3(s,k)*Cpred3(s,k)*H3(s,k)'+R)^(-1);
    fil3(s,k)=pred3(s,k)+K3(s,k)*(y(s,k)-pred3(s,k)^2-...
        exp(pred3(s,k))-((1/2)*(2+exp(pred3(s,k))))*...
        Cpred3(s,k));
    Cfil3(s,k)=(1-K3(s,k)*H3(s,k))*Cpred3(s,k);

    pred3(s,k+1)=1/(fil3(s,k)^2+3)+(1/2)*...
        (((8*fil3(s,k)^2)/(fil3(s,k)^2+3)^2)-...
        (2/(fil3(s,k)^2+3)^2))*Cfil3(s,k);
    F3(s,k)=(-2)*fil3(s,k)/(fil3(s,k)^2+3)^2;
    Cpred3(s,k+1)=F3(s,k)*Cfil3(s,k)*F3(s,k)'+Q;
    DC_fil3(s,k)=(x(s,k)-fil3(s,k))^2;

end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO DE KALMAN ENSEMBLE %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fil4=filtro de Kalman ensemble (FKEn)
%pred4=predicador de Kalman ensemble (PKEn)
%Cfil4=cuasivarianza del FKEn
%Cpred4=cuasivarianza del PKEn
%K4=ganancia del FKEn
%DC_fil4=cuadrado de la diferencia entre la señal y el FKEn

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Generación del ensemble de filtrado y predicción %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %efil=conjunto de filtros
    %epred=conjunto de predictores
    %ew=conjunto de valores simulados a partir de la distribución de w
    %ev=conjunto de valores simulados a partir de la distribución de v
    %obs=conjunto de predictores de las observaciones
    %yest=media de obs
    %Cyy=cuasivarianza de obs
    %Cxy=cuasicovarianza del PKEn y obs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    q=100; %tamaño del conjunto de valores simulados

    %% Simulación del ensemble de filtrado y predicción inicial %%

    for j=1:q

        efil0(s,j)=P0^(1/2)*randn(1,1);
        ew0(s,j)=Q^(1/2)*randn(1,1);
        epred(s,1,j)= 1/(efil0(s,j)^2+3)+ew0(s,j);

    end

    %% Cálculo del PKEn y su cuasivarianza para k=1 %%

    pred4(s,1)=mean(epred(s,1,:));
    Cpred4(s,1)=var(epred(s,1,:)); %función que calcula la cuasivarianza

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del FKEn y su cuasivarianza para k %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    %% Cálculo de los predictores de las observaciones en k %%

    for j=1:q

        ev(s,k,j)=R^(1/2)*randn(1,1);
        obs(s,k,j)=epred(s,k,j)^2+exp(epred(s,k,j))+ev(s,k,j);

    end

    yest(s,k)=mean(obs(s,k,:));
    Cyy(s,k)=var(obs(s,k,:));

    %% Cálculo de la cuasicovarianza del PKEn y obs en k%%

    restapred(s,k,1)=epred(s,k,1)-pred4(s,k);
    restayy(s,k,1)=obs(s,k,1)-yest(s,k);
    Cxysum(s,k,1)=restapred(s,k,1)*restayy(s,k,1);

    for j=2:q

        restapred(s,k,j)=epred(s,k,j)-pred4(s,k);
        restayy(s,k,j)=obs(s,k,j)-yest(s,k);
        Cxysum(s,k,j)=Cxysum(s,k,j-1)+...
            (restapred(s,k,j)*restayy(s,k,j));

    end

    Cxy(s,k)=Cxysum(s,k,q)/(q-1);

    %% Cálculo del FKEn y su cuasivarianza en k %%

    K4(s,k)=Cxy(s,k)*Cyy(s,k)^(-1);

    for j=1:q
        efil(s,k,j)=epred(s,k,j)+K4(s,k)*(y(s,k)-obs(s,k,j));
    end

    fil4(s,k)=mean(efil(s,k,:));
    Cfil4(s,k)=var(efil(s,k,:));
    DC_fil4(s,k)=(x(s,k)-fil4(s,k))^2;

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del PKEn y su cuasivarianza para k+1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
  
    for j=1:q  
  
        ew(s,k,j)=Q^(1/2)*randn(1,1);  
        epred(s,k+1,j)= 1/(efil(s,k,j)^2+3)+ew(s,k,j);  
  
    end  
  
    pred4(s,k+1)=mean(epred(s,k+1,:));  
    Cpred4(s,k+1)=var(epred(s,k+1,:));  
  
end %iter
```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    FILTRO EXTENDIDO CUADRÁTICO
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Fil=filtro del sistema aumentado
%Pred=predicador del sistema aumentado
%CFil=matriz de covarianzas de Fil
%CPred=matriz de covarianzas de Pred
%filtro=filtro extendido cuadrático (FEC)
%predicador=predicador extendido cuadrático (PEC)
%FQ=derivada de la función f en el FEC
%FF= matriz de la ecuación de la señal aumentada
%HQ=derivada de la función h en el PEC
%HH=matriz de la ecuación observación aumentada
%d=esperanza de la señal linealizada
%D=varianza de la señal linealizada
%QQ=matriz de covarianzas del ruido de la ecuación de la señal aumentada
%Q_ij=componente ij de QQ
%RR=matriz de covarianzas del ruido de la ecuación observación aumentada
%R_ij=componente ij de RR
%Pi=matriz de covarianzas de la innovación
%KK=ganancia del sistema aumentado

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %%% Condiciones iniciales %%%

    filtro0(s)=0;
    Fil0(:,s)=[0;0];
    CFil0(:,s)=[P0 P30;P30' P40];
    FQ0(s)=(-2)*filtro0(s)/(filtro0(s)^2+3)^2;
    FF0(:,s)=[FQ0(s) 0;0 FQ0(s)^2];

    u0(s)=1/(filtro0(s)^2+3)-FQ0(s)*filtro0(s);
    d0(s)=0;
    D0(s)=P0;

    X0(:,s)=[x0(s); x0(s)^2]; %señal del sistema aumentado
    EX0(:,s)=[d0(s); D0(s)]; %esperanza de X0(:,s)
    XX0(:,s)=X0(:,s)-EX0(:,s); %señal del sistema aumentado centrada

    Q0_11=Q;
    Q0_12=Q3;
    Q0_21=Q0_12';
    Q0_22=4*FQ0(s)^2*D0(s)*Q+Q4;
    QQ0(:,s)=[Q0_11, Q0_12;Q0_21, Q0_22];

    Pred(:,s,1)=FF0(:,s)*Fil0(:,s);
    CPred(:,s,1)=FF0(:,s)*CFil0(:,s)*FF0(:,s)'+QQ0(:,s);

```

```

d(s,1)=FQ0(s)*d0(s)+u0(s);
D(s,1)=FQ0(s)*D0(s)*FQ0(s)'+Q;

X(:,s,1)=[x(s,1); x(s,1)^2];
EX(:,s,1)=[d(s,1);D(s,1)];
XX(:,s,1)=X(:,s,1)-EX(:,s,1);
predictor(s,1)= Pred(1,s,1)+d(s,1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    HQ(s,k)=2*predictor(s,k)+exp(predictor(s,k));
    HH(:,:,s,k)=[HQ(s,k) 0;0 HQ(s,k)^2];
    Psi(s,k)=2*HQ(s,k)*XX(1,s,k)*v(s,k)+v(s,k)^2;
    V(:,s,k)=[v(s,k); Psi(s,k)-R];

    z(s,k)=predictor(s,k)^2+exp(predictor(s,k))-...
            HQ(s,k)*predictor(s,k);
    dy(s,k)=HQ(s,k)*d(s,k)+z(s,k);
    Dy(s,k)=HQ(s,k)*D(s,k)*HQ(s,k)'+R;

    Y(:,s,k)=HH(:,:,s,k)*XX(:,s,k)+V(:,s,k);
    R_11(s,k)=R;
    R_12(s,k)= R3;
    R_21(s,k)=R_12(s,k)';
    R_22(s,k)=4*HQ(s,k)^2*D(s,k)*R+R4;
    RR(:,:,s,k)=[R_11(s,k) R_12(s,k);R_21(s,k) R_22(s,k)];
    Pi(:,:,s,k)=HH(:,:,s,k)*CPred(:,:,s,k)*HH(:,:,s,k)'+RR(:,:,s,k);
    KK(:,:,s,k)=CPred(:,:,s,k)*HH(:,:,s,k)'+Pi(:,:,s,k)^(-1);
    Fil(:,s,k)=Pred(:,s,k)+KK(:,:,s,k)*(Y(:,s,k)-...
            HH(:,:,s,k)*Pred(:,s,k));
    filtro(s,k)=Fil(1,s,k)+d(s,k);
    CFil(:,:,s,k)=(eye(2)-KK(:,:,s,k)*HH(:,:,s,k))*CPred(:,:,s,k);
    DC_Fil(s,k)=(x(s,k)-filtro(s,k))^2;

    FQ(s,k)=(-2)*filtro(s,k)/(filtro(s,k)^2+3)^2;
    FF(:,:,s,k)=[FQ(s,k) 0;0 FQ(s,k)^2];

    Q_11(s,k)=Q;
    Q_12(s,k)=Q3;
    Q_21(s,k)=Q_12(s,k)';
    Q_22(s,k)=4*FQ(s,k)*D(s,k)*FQ(s,k)'+Q+Q4;
    QQ(:,:,s,k)=[Q_11(s,k) Q_12(s,k);Q_21(s,k) Q_22(s,k)];

    Pred(:,s,k+1)=FF(:,:,s,k)*Fil(:,s,k);
    CPred(:,:,s,k+1)=FF(:,:,s,k)*CFil(:,:,s,k)*FF(:,:,s,k)'+...
            QQ(:,:,s,k);

```

```

    u(s,k)=1/(filtro(s,k)^2+3)-FQ(s,k)*filtro(s,k);
    d(s,k+1)= FQ(s,k)*d(s,k)+u(s,k);
    D(s,k+1)=FQ(s,k)^2*D(s,k)+Q;

    X(:,s,k+1)=[x(s,k+1); x(s,k+1)^2];
    EX(:,s,k+1)=[d(s,k+1); D(s,k+1)];
    XX(:,s,k+1)=X(:,s,k+1)-EX(:,s,k+1);

    predictor(s,k+1)=Pred(1,s,k+1)+d(s,k+1);
end %iter
end %simul

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Errores cuadráticos medios del FKE, FKEI, FKE20, FKE n y FEC %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter
    MSE_fil1(1,k)=DC_fil1(1,k);
    MSE_fil2(1,k)=DC_fil2(1,k);
    MSE_fil3(1,k)=DC_fil3(1,k);
    MSE_fil4(1,k)=DC_fil4(1,k);
    MSE_Fil(1,k)=DC_Fil(1,k);
    for s=2:n_simul
        MSE_fil1(s,k)=MSE_fil1(s-1,k)+DC_fil1(s,k);
        MSE_fil2(s,k)=MSE_fil2(s-1,k)+DC_fil2(s,k);
        MSE_fil3(s,k)=MSE_fil3(s-1,k)+DC_fil3(s,k);
        MSE_fil4(s,k)=MSE_fil4(s-1,k)+DC_fil4(s,k);
        MSE_Fil(s,k)=MSE_Fil(s-1,k)+DC_Fil(s,k);
    end
    RMSE_fil1(k)=(MSE_fil1(n_simul,k)/n_simul)^(1/2);
    RMSE_fil2(k)=(MSE_fil2(n_simul,k)/n_simul)^(1/2);
    RMSE_fil3(k)=(MSE_fil3(n_simul,k)/n_simul)^(1/2);
    RMSE_fil4(k)=(MSE_fil4(n_simul,k)/n_simul)^(1/2);
    RMSE_Fil(k)=(MSE_Fil(n_simul,k)/n_simul)^(1/2);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Gráfica de los errores cuadráticos medios %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

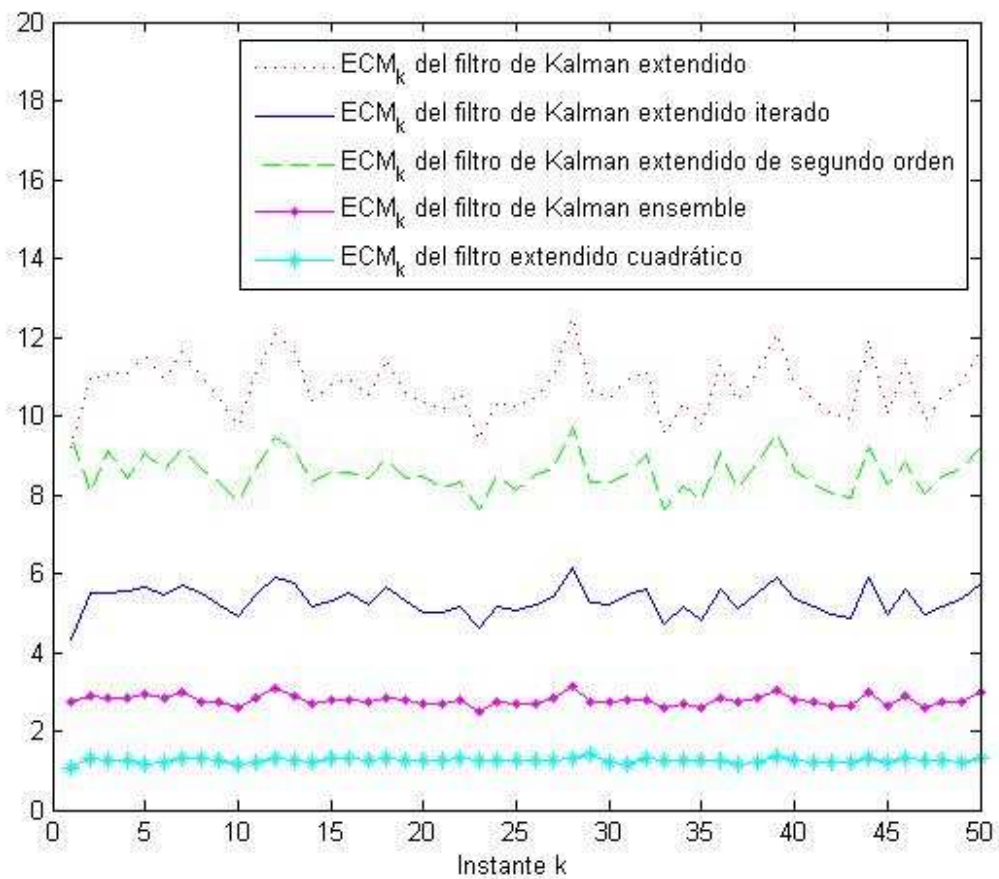
for k=1:n_iter
    time(k)=k;
    pRMSE_fil1(k)=RMSE_fil1(k);
    pRMSE_fil2(k)=RMSE_fil2(k);
    pRMSE_fil3(k)=RMSE_fil3(k);
    pRMSE_fil4(k)=RMSE_fil4(k);
    pRMSE_Fil(k)=RMSE_Fil(k);
end

```

```

figure(2.1)
grafic1=plot(time,pRMSE_fil1,'r:',time,pRMSE_fil2,'b-',...
             time,pRMSE_fil3,'g--',time,pRMSE_fil4,'m.-',...
             time,pRMSE_Fil,'c*-');
xlabel('Instante k')
legend('ECMk del filtro de Kalman extendido',...
      'ECMk del filtro de Kalman extendido iterado',...
      'ECMk del filtro de Kalman extendido de segundo orden',...
      'ECMk del filtro de Kalman ensemble',...
      'ECMk del filtro extendido cuadrático')
saveas(gcf, 'Fig2_1', 'fig')

```



## CAPÍTULO 2: Figura 2.2

```

%Modelo de la Señal:   x(k+1)=1/(x(k)^2+3)+w(k)   donde   x0-->N(0,1)
%                               f(x)=1/(x^2+3)

%Observaciones:      y(k)=x(k)^2+exp{x(k)}+v(k)   donde
%                               h(x)=x^2+exp{x}

%   P[w(k)=1]=15/18      P[w(k)=-3]=2/18      P[w(k)=-9]=1/18
%   P[v(k)=1]=15/18      P[v(k)=-3]=2/18      P[v(k)=-9]=1/18

n_simul=1000; %número de simulaciones
n_iter=50;    %número de iteraciones

for s=1:n_simul

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%% Generación de la señal x(k) %%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    P0=1;          %varianza de la señal inicial
    P30=0;         %momento centrado de tercer orden de la señal inicial
    P40=3*P0^2-P0^2; %momento centrado de cuarto orden de la señal inicial

    Q=19/3;        %varianza del ruido de la ecuación de la señal w(k)
    Q3=-128/3;     %momento centrado de tercer orden del ruido w(k)
    Q4=1123/3-Q^2; %momento centrado de cuarto orden del ruido w(k)

    %% Condiciones iniciales %%

    a=rand(1,1);
    if a <= 1/18
        w0(s)=-9;
    elseif a <= 3/18
        w0(s)=-3;
    else w0(s)=1;
    end

    x0(s)=P0^(1/2)*randn(1,1);
    x(s,1)=1/(x0(s)^2+3)+w0(s);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

for k=1:n_iter
    b=rand(1,1);
    if b <= 1/18
        w(s,k)=-9;
    elseif b <= 3/18
        w(s,k)=-3;
    else w(s,k)=1;
    end
    x(s,k+1)=1/(x(s,k)^2+3)+w(s,k);
end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Generación de las observaciones y(k) %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

R=19/3;           %varianza del ruido de la ecuación observación v(k)
R3=-128/3;       %momento centrado de tercer orden del ruido v(k)
R4=1123/3-R^2;   %momento centrado de cuarto orden del ruido v(k)

```

```

for k=1:n_iter
    c=rand(1,1);
    if c <= 1/18
        v(s,k)=-9;
    elseif c <= 3/18
        v(s,k)=-3;
    else v(s,k)=1;
    end
    y(s,k)=x(s,k)^2+exp(x(s,k))+v(s,k);
end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO EXTENDIDO CUADRÁTICO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Fil=filtro del sistema aumentado
%Pred=predicador del sistema aumentado
%CFil=matriz de covarianzas de Fil
%CPred=matriz de covarianzas de Pred
%filtro=filtro extendido cuadrático (FEC)
%predicador=predicador extendido cuadrático (PEC)
%FQ=derivada de la función f en el FEC
%FF= matriz de la ecuación de la señal aumentada
%HQ=derivada de la función h en el PEC
%HH=matriz de la ecuación observación aumentada
%d=esperanza de la señal linealizada
%D=varianza de la señal linealizada
%QQ=matriz de covarianzas del ruido de la ecuación de la señal aumentada
%Q_ij=componente ij de la matriz QQ
%RR=matriz de covarianzas del ruido de la ecuación observación aumentada
%R_ij=componente ij de la matriz RR
%Pi=matriz de covarianzas de la innovación
%KK=ganancia del sistema aumentado

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %%%% Condiciones iniciales %%%%

    filtro0(s)=0;
    Fil0(:,s)=[0;0];
    CFil0(:,s)=[P0 P30;P30' P40];
    FQ0(s)=(-2)*filtro0(s)/(filtro0(s)^2+3)^2;
    FF0(:,s)=[FQ0(s) 0;0 FQ0(s)^2];

    u0(s)=1/(filtro0(s)^2+3)-FQ0(s)*filtro0(s);
    d0(s)=0;
    D0(s)=P0;

    X0(:,s)=[x0(s); x0(s)^2]; %señal del sistema aumentado
    EX0(:,s)=[d0(s); D0(s)]; %esperanza de X0(:,s)
    XX0(:,s)=X0(:,s)-EX0(:,s); %señal del sistema aumentado centrada

    Q0_11=Q;
    Q0_12=Q3;
    Q0_21=Q0_12';
    Q0_22=4*FQ0(s)^2*D0(s)*Q+Q4;
    QQ0(:,s)=[Q0_11, Q0_12;Q0_21, Q0_22];

```

```

Pred(:,s,1)=FF0(:,s)*Fil0(:,s);
CPred(:,s,1)=FF0(:,s)*CFil0(:,s)*FF0(:,s)'+QQ0(:,s);

d(s,1)=FQ0(s)*d0(s)+u0(s);
D(s,1)=FQ0(s)*D0(s)*FQ0(s)'+Q;
EX(:,s,1)=[d(s,1);D(s,1)];
X(:,s,1)=[x(s,1); x(s,1)^2];
XX(:,s,1)=X(:,s,1)-EX(:,s,1);
predictor(s,1)= Pred(1,s,1)+d(s,1);

for k=1:n_iter

    HQ(s,k)=2*predictor(s,k)+exp(predictor(s,k));
    HH(:,s,k)=[HQ(s,k) 0;0 HQ(s,k)^2];
    Psi(s,k)=2*HQ(s,k)*XX(1,s,k)*v(s,k)+v(s,k)^2;
    V(:,s,k)=[v(s,k); Psi(s,k)-R];
    z(s,k)=predictor(s,k)^2+exp(predictor(s,k))-...
           HQ(s,k)*predictor(s,k);

    Y(:,s,k)=HH(:,s,k)*XX(:,s,k)+V(:,s,k);

    R_11(s,k)=R;
    R_12(s,k)= R3;
    R_21(s,k)=R_12(s,k)';
    R_22(s,k)=4*HQ(s,k)^2*D(s,k)*R+R4;
    RR(:,s,k)=[R_11(s,k) R_12(s,k);R_21(s,k) R_22(s,k)];
    Pi(:,s,k)=HH(:,s,k)*CPred(:,s,k)*HH(:,s,k)'+RR(:,s,k);
    KK(:,s,k)=CPred(:,s,k)*HH(:,s,k)'+Pi(:,s,k)^(-1);

    Fil(:,s,k)=Pred(:,s,k)+KK(:,s,k)*(Y(:,s,k)-...
           HH(:,s,k)*Pred(:,s,k));
    filtro(s,k)=Fil(1,s,k)+d(s,k);
    CFil(:,s,k)=(eye(2)-KK(:,s,k)*HH(:,s,k))*CPred(:,s,k);
    DC_Fil(s,k)=(x(s,k)-filtro(s,k))^2;

    FQ(s,k)=(-2)*filtro(s,k)/(filtro(s,k)^2+3)^2;
    FF(:,s,k)=[FQ(s,k) 0;0 FQ(s,k)^2];

    Q_11(s,k)=Q;
    Q_12(s,k)=Q3;
    Q_21(s,k)=Q_12(s,k)';
    Q_22(s,k)=4*FQ(s,k)*D(s,k)*FQ(s,k)'+Q+Q4;
    QQ(:,s,k)=[Q_11(s,k) Q_12(s,k);Q_21(s,k) Q_22(s,k)];

```



```
Pred(:,s,k+1)=FF(:,:,s,k)*Fil(:,s,k);
CPred(:,:,s,k+1)=FF(:,:,s,k)*CFil(:,:,s,k)*...
    FF(:,:,s,k)'+QQ(:,:,s,k);

u(s,k)=1/(filtro(s,k)^2+3)-FQ(s,k)*filtro(s,k);
d(s,k+1)= FQ(s,k)*d(s,k)+u(s,k);
D(s,k+1)=FQ(s,k)^2*D(s,k)+Q;

X(:,s,k+1)=[x(s,k+1); x(s,k+1)^2];
EX(:,s,k+1)=[d(s,k+1); D(s,k+1)];
XX(:,s,k+1)=X(:,s,k+1)-EX(:,s,k+1);

predictor(s,k+1)=Pred(1,s,k+1)+d(s,k+1);

end %iter
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILTRO EXTENDIDO CUADRÁTICO ITERADO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Fil2=filtro del sistema aumentado
%Pred2=predicador del sistema aumentado
%CFil2=matriz de covarianzas de Fil2
%CPred2=matriz de covarianzas de Pred2
%filtro2=filtro extendido cuadrático iterado (FECI)
%predictor2=predicador extendido cuadrático iterado (PECI)
%FQ2=derivada de la función f en el FECI
%FF2= matriz de la ecuación de la señal aumentada
%HQ2=derivada de la función h en el PECI
%HH2=matriz de la ecuación observación aumentada
%d2=esperanza de la señal linealizada
%D2=varianza de la señal linealizada
%QQ2=matriz de covarianzas del ruido de la ecuación de la señal aumentada
%Q2_ij=componente ij de la matriz QQ2
%RR2=matriz de covarianzas del ruido de la ecuación observación aumentada
%R2_ij=componente ij de la matriz RR2
%Pi2=matriz de covarianzas de la innovación
%KK2=ganancia del sistema aumentado

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Condiciones iniciales %%%%%%%%%

filtro02(s)=0;
Fil02(:,s)=[0;0];
CFil02(:,:,s)=[P0 P30;P30' P40];
FQ02(s)=(-2)*filtro02(s)/(filtro02(s)^2+3)^2;
FF02(:,:,s)=[FQ02(s) 0;0 FQ02(s)^2];

u02(s)=1/(filtro02(s)^2+3)-FQ02(s)*filtro02(s);
d02(s)=0;
D02(s)=P0;

X02(:,s)=[x0(s); x0(s)^2]; %señal del sistema aumentado
EX02(:,s)=[d02(s); D02(s)]; %esperanza de X02(:,s)
XX02(:,s)=X02(:,s)-EX02(:,s); %señal del sistema aumentado centrada

d2(s,1)=FQ02(s)*d02(s)+u02(s);
D2(s,1)=FQ02(s)*D02(s)*FQ02(s)'+Q;

X2(:,s,1)=[x(s,1); x(s,1)^2];
EX2(:,s,1)=[d2(s,1);D2(s,1)];
XX2(:,s,1)=X2(:,s,1)-EX2(:,s,1);

```

```

Q02_11=Q;
Q02_12=Q3;
Q02_21=Q02_12';
Q02_22=4*FQ02(s)*D02(s)*FQ02(s)'+Q+Q4;
QQ02(:,:,s)=[Q02_11, Q02_12;Q02_21, Q02_22];

Pred2(:,s,1)=FF02(:,:,s)*Fil02(:,s);
predictor2(s,1)= Pred2(1,s,1)+d2(s,1);
CPred2(:,:,s,1)=FF02(:,:,s)*CFil02(:,:,s)*FF02(:,:,s)'+QQ02(:,:,s);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Bucle de iteraciones para el cálculo del FECI %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Fil2k(i)=filtro del sistema aumentado en la iteración i
%CFil2k(i)=matriz de covarianzas de Fil2k(i)
%filtro2k(i)=filtro extendido cuadrático en la iteración i
%HQ2k(i)=derivada de la función h en el filtro2k(i)
%HH2k(i)=matriz de la ecuación observación aumentada en la iteración i
%V2k(i)=ruido de la ecuación observación aumentada en la iteración i
%RR2k(i)=matriz de covarianzas de V2k(i)
%R2k_ij=componente ij de la matriz RR2k(i)
%Pi2k(i)=matriz de covarianzas de la innovación en el aiteración i
%KK2k(i)=ganancia en la iteración i del bucle

N=20; %número de iteraciones del bucle que calcula el FECI

for k=1:n_iter

    HQ2(s,k)=2*predictor2(s,k)+exp(predictor2(s,k));
    HH2(:,:,s,k)=[HQ2(s,k) 0; 0 HQ2(s,k)^2];
    Psi2(s,k)=2*HQ2(s,k)*XX2(1,s,k)*v(s,k)+v(s,k)^2;
    V2(:,s,k)=[v(s,k); (Psi2(s,k)-R)];
    z2(s,k)=predictor2(s,k)^2+exp(predictor2(s,k))-...
        HQ2(s,k)*predictor2(s,k);

    Y2(:,s,k)=HH2(:,:,s,k)*XX2(:,s,k)+V2(:,s,k);

    R2_11(s,k)=R;
    R2_12(s,k)= R3;
    R2_21(s,k)=R2_12(s,k)';
    R2_22(s,k)=4*HQ2(s,k)*D2(s,k)*HQ2(s,k)'+R+R4;
    RR2(:,:,s,k)=[R2_11(s,k) R2_12(s,k);R2_21(s,k) R2_22(s,k)];

    Pi2(:,:,s,k)=HH2(:,:,s,k)*CPred2(:,:,s,k)*HH2(:,:,s,k)'+...
        RR2(:,:,s,k);
    KK2(:,:,s,k)=CPred2(:,:,s,k)*HH2(:,:,s,k)'\*Pi2(:,:,s,k)^(-1);

```

```

%%% Condiciones iniciales del bucle para el cálculo del FECI %%%

Fil2k0(:,s,k)=Pred2(:,s,k)+KK2(:,s,k)*(Y2(:,s,k)-...
    HH2(:,s,k)*Pred2(:,s,k));
filtro2k0(s,k)=Fil2k0(1,s,k)+d2(s,k); %FEC
CFil2k0(:,s,k)=(eye(2)-KK2(:,s,k)*HH2(:,s,k))*CPred2(:,s,k);

HQ2k0(s,k)=2*filtro2k0(s,k)+exp(filtro2k0(s,k));
HH2k0(:,s,k)=[HQ2k0(s,k) 0;0 HQ2k0(s,k)^2];
Psi2k0(s,k)=2*HQ2k0(s,k)*XX2(1,s,k)*v(s,k)+v(s,k)^2;
V2k0(:,s,k)=[v(s,k); (Psi2k0(s,k)-R)];

z2k0(s,k)=filtro2k0(s,k)^2+exp(filtro2k0(s,k))-...
    HQ2k0(s,k)*filtro2k0(s,k);

Y2k0(:,s,k)=HH2k0(:,s,k)*XX2(:,s,k)+V2k0(:,s,k);

R2k0_11(s,k)=R;
R2k0_12(s,k)= R3;
R2k0_21(s,k)=R2k0_12(s,k)';
R2k0_22(s,k)=4*HQ2k0(s,k)*D2(s,k)*HQ2k0(s,k)'+R+R4;
RR2k0(:,s,k)=[R2k0_11(s,k) R2k0_12(s,k);...
    R2k0_21(s,k) R2k0_22(s,k)];

Pi2k0(:,s,k)=HH2k0(:,s,k)*CPred2(:,s,k)*HH2k0(:,s,k)'+...
    RR2k0(:,s,k);
KK2k0(:,s,k)=CPred2(:,s,k)*HH2k0(:,s,k)'+Pi2k0(:,s,k)^(-1);

Fil2k(:,s,k,1)=Pred2(:,s,k)+KK2k0(:,s,k)*(Y2k0(:,s,k)-...
    HH2k0(:,s,k)*Pred2(:,s,k));

filtro2k(s,k,1)=Fil2k(1,s,k,1)+d2(s,k);
CFil2k(:,s,k,1)=(eye(2)-KK2k0(:,s,k)*HH2k0(:,s,k))*...
    CPred2(:,s,k);

for i=1:N

    HQ2k(s,k,i)=2*filtro2k(s,k,i)+exp(filtro2k(s,k,i));
    HH2k(:,s,k,i)=[HQ2k(s,k,i) 0;0 HQ2k(s,k,i)^2];
    Psi2k(s,k,i)=2*HQ2k(s,k,i)*XX2(1,s,k)*v(s,k)+v(s,k)^2;
    V2k(:,s,k,i)=[v(s,k); (Psi2k(s,k,i)-R)];

    z2k(s,k,i)=filtro2k(s,k,i)^2+exp(filtro2k(s,k,i))-...
        HQ2k(s,k,i)*filtro2k(s,k,i);

    Y2k(:,s,k,i)=HH2k(:,s,k,i)*XX2(:,s,k)+V2k(:,s,k,i);

```

```

R2k_11(s,k,i)=R;
R2k_12(s,k,i)=R3;
R2k_21(s,k,i)=R2k_12(s,k,i)';
R2k_22(s,k,i)=4*HQ2k(s,k,i)*D2(s,k)*HQ2k(s,k,i)*R+R4;

RR2k(:,:,s,k,i)=[R2k_11(s,k,i) R2k_12(s,k,i);...
                  R2k_21(s,k,i) R2k_22(s,k,i)];
Pi2k(:,:,s,k,i)=HH2k(:,:,s,k,i)*CPred2(:,:,s,k)*...
                HH2k(:,:,s,k,i)'+RR2k(:,:,s,k,i);
KK2k(:,:,s,k,i)=CPred2(:,:,s,k)*HH2k(:,:,s,k,i)'+...
                Pi2k(:,:,s,k,i)^(-1);

Fil2k(:,s,k,i+1)=Pred2(:,s,k)+KK2k(:,:,s,k,i)*(Y2k(:,s,k,i)-...
                HH2k(:,:,s,k,i)*Pred2(:,s,k));
filtro2k(s,k,i+1)=Fil2k(1,s,k,i+1)+d2(s,k);
CFil2k(:,:,s,k,i+1)=(eye(2)-KK2k(:,:,s,k,i)*HH2k(:,:,s,k,i))*...
                CPred2(:,:,s,k);

end

Fil2(:,s,k)=Fil2k(:,s,k,N+1);
CFil2(:,:,s,k)=CFil2k(:,:,s,k,N+1);
filtro2(s,k)=Fil2(1,s,k);
DC_Fil2(s,k)=(x(s,k)-filtro2(s,k))^2;

FQ2(s,k)=(-2)*filtro2(s,k)/(filtro2(s,k)^2+3)^2;
FF2(:,:,s,k)=[FQ2(s,k) 0;0 FQ2(s,k)^2];

Q2_11(s,k)=Q;
Q2_12(s,k)=Q3;
Q2_21(s,k)=Q2_12(s,k)';
Q2_22(s,k)=4*FQ2(s,k)*D2(s,k)*FQ2(s,k)'+Q+Q4;
QQ2(:,:,s,k)=[Q2_11(s,k) Q2_12(s,k);Q2_21(s,k) Q2_22(s,k)];

Pred2(:,s,k+1)=FF2(:,:,s,k)*Fil2(:,s,k);
CPred2(:,:,s,k+1)=FF2(:,:,s,k)*CFil2(:,:,s,k)*FF2(:,:,s,k)'+...
                QQ2(:,:,s,k);

u2(s,k)=1/(filtro2(s,k)^2+3)-FQ2(s,k)*filtro2(s,k);
d2(s,k+1)= FQ2(s,k)*d2(s,k)+u2(s,k);
D2(s,k+1)=FQ2(s,k)*D2(s,k)*FQ2(s,k)'+Q;

X2(:,s,k+1)=[x(s,k+1); x(s,k+1)^2];
EX2(:,s,k+1)=[d2(s,k+1); D2(s,k+1)];
XX2(:,s,k+1)=X2(:,s,k+1)-EX2(:,s,k+1);

predictor2(s,k+1)=Pred2(1,s,k+1)+d2(s,k+1);

end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%***** FILTRO EXTENDIDO CUADRÁTICO DE 2º ORDEN %*****
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Fil3=filtro del sistema aumentado
%Pred3=predicador del sistema aumentado
%CFil3=matriz de covarianzas de Fil3
%CPred3=matriz de covarianzas de Pred3
%filtro3=filtro extendido cuadrático de segundo orden (FEC20)
%predicador3=predicador extendido cuadrático de segundo orden (PEC20)
%FQ3=derivada de la función f en el FEC20
%FF3= matriz de la ecuación de la señal aumentada
%HQ3=derivada de la función h en el PEC20
%HH3=matriz de la ecuación observación aumentada
%d3=esperanza de la señal linealizada
%D3=varianza de la señal linealizada
%QQ3=matriz de covarianzas del ruido de la ecuación de la señal aumentada
%Q3_ij=componente ij de la matriz QQ3
%RR3=matriz de covarianzas del ruido de la ecuación observación aumentada
%R3_ij=componente ij de la matriz RR3
%Pi3=matriz de covarianzas de la innovación
%KK3=ganancia del sistema aumentado

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%***** Condiciones iniciales %*****

filtro03(s)=0;
Fil03(:,s)=[0;0];
CFil03(:,s)=[P0 P30;P30' P40];
FQ03(s)=(-2)*filtro0(s)/(filtro0(s)^2+3)^2;
FF03(:,s)=[FQ03(s) 0;0 FQ03(s)^2];

u03(s)=1/(filtro03(s)^2+3)-FQ03(s)*filtro03(s)+(1/2)*...
(8*filtro03(s)^2/(filtro03(s)^2+3)^3-...
2/(filtro03(s)^2+3)^2)*CFil03(1,1,s);
d03(s)=0;
D03(s)=P0;

X03(:,s)=[x0(s); x0(s)^2]; %señal del sistema aumentado
EX03(:,s)=[d03(s); D03(s)]; %esperanza X03(:,s)
XX03(:,s)=X03(:,s)-EX03(:,s); %señal del sistema aumentado centrada

Q03_11=Q;
Q03_12=Q3;
Q03_21=Q03_12';
Q03_22=4*FQ03(s)^2*D03(s)*Q+Q4;
QQ03(:,s)=[Q03_11, Q03_12;Q03_21, Q03_22];

```

```

Pred3(:,s,1)=FF03(:,s)*Fil03(:,s);
CPred3(:,s,1)=FF03(:,s)*CFil03(:,s)*FF03(:,s)'+QQ03(:,s);

d3(s,1)=FQ03(s)*d03(s)+u03(s);
D3(s,1)=FQ03(s)*D03(s)*FQ03(s)'+Q;
X3(:,s,1)=[x(s,1); x(s,1)^2];
EX3(:,s,1)=[d3(s,1);D3(s,1)];
XX3(:,s,1)=X3(:,s,1)-EX3(:,s,1);

predictor3(s,1)= Pred3(1,s,1)+d3(s,1);

for k=1:n_iter

    HQ3(s,k)=2*predictor3(s,k)+exp(predictor3(s,k));
    HH3(:,s,k)=[HQ3(s,k) 0;0 HQ3(s,k)^2];
    Psi3(s,k)=2*HQ3(s,k)*XX3(1,s,k)*v(s,k)+v(s,k)^2;
    V3(:,s,k)=[v(s,k); Psi3(s,k)-R];
    z3(s,k)=predictor3(s,k)^2+exp(predictor3(s,k))-...
            HQ3(s,k)*predictor3(s,k)+...
            (1/2)*(2+exp(predictor3(s,k)))*CPred3(1,1,s,k);

    Y3(:,s,k)=HH3(:,s,k)*XX3(:,s,k)+V3(:,s,k);

    R3_11(s,k)=R;
    R3_12(s,k)= R3;
    R3_21(s,k)=R3_12(s,k)';
    R3_22(s,k)=4*HQ3(s,k)^2*D3(s,k)*R+R4;
    RR3(:,s,k)=[R3_11(s,k) R3_12(s,k);R3_21(s,k) R3_22(s,k)];
    Pi3(:,s,k)=HH3(:,s,k)*CPred3(:,s,k)*HH3(:,s,k)'+...
            RR3(:,s,k);
    KK3(:,s,k)=CPred3(:,s,k)*HH3(:,s,k)'+Pi3(:,s,k)^(-1);

    Fil3(:,s,k)=Pred3(:,s,k)+KK3(:,s,k)*(Y3(:,s,k)-...
            HH3(:,s,k)*Pred3(:,s,k));
    filtro3(s,k)=Fil3(1,s,k)+d3(s,k);
    CFil3(:,s,k)=(eye(2)-KK3(:,s,k)*HH3(:,s,k))*...
            CPred3(:,s,k);
    DC_Fil3(s,k)=(x(s,k)-filtro3(s,k))^2;

    FQ3(s,k)=(-2)*filtro3(s,k)/(filtro3(s,k)^2+3)^2;
    FF3(:,s,k)=[FQ3(s,k) 0;0 FQ3(s,k)^2];

    Q3_11(s,k)=Q;
    Q3_12(s,k)=Q3;
    Q3_21(s,k)=Q3_12(s,k)';
    Q3_22(s,k)=4*FQ3(s,k)*D3(s,k)*FQ3(s,k)'+Q+Q4;

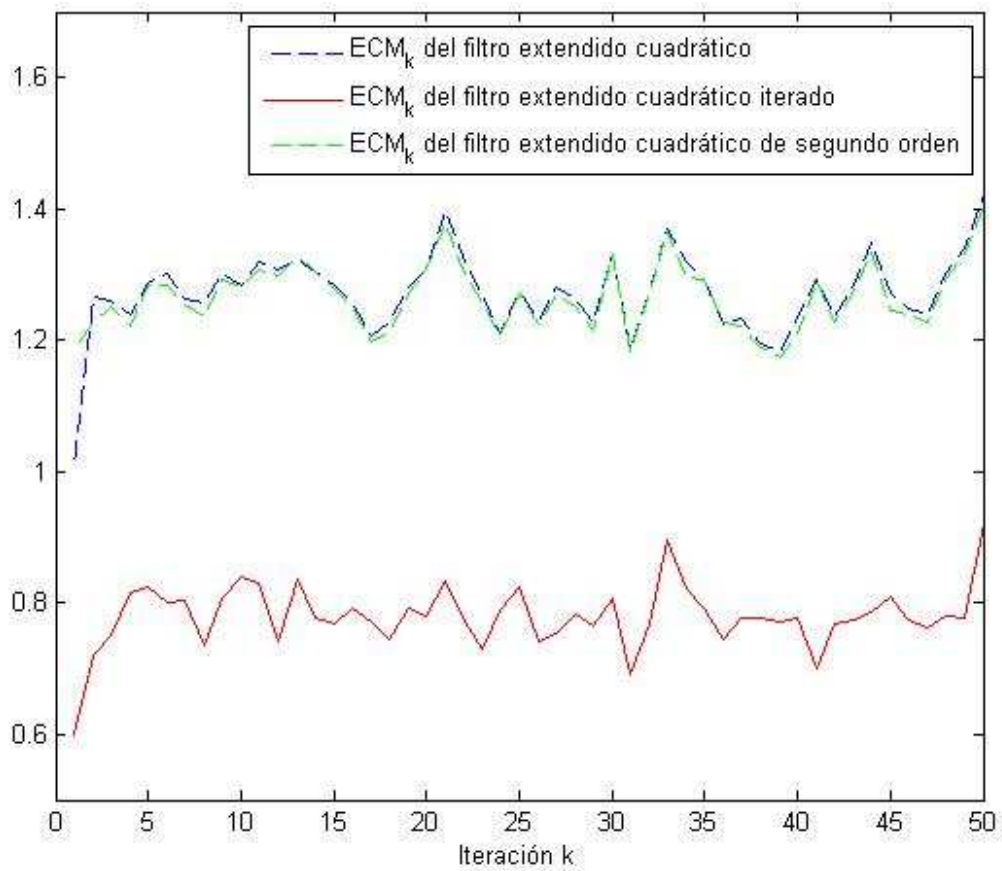
    QQ3(:,s,k)=[Q3_11(s,k) Q3_12(s,k);Q3_21(s,k) Q3_22(s,k)];

```





```
graf(2.2)=plot(time,pECM_Fil,'b--',time,pECM_Fil2,'r-',...  
              time,pECM_Fil3,'g--');  
xlabel('Iteración k')  
legend('ECM_k del filtro extendido cuadrático',...  
      'ECM_k del filtro extendido cuadrático iterado',...  
      'ECM_k del filtro extendido cuadrático de segundo orden')  
saveas(gcf, 'Fig2_2', 'fig')
```



### CAPÍTULO 3: Figura 3.1

```

%Modelo de la Señal:      x(k+1)=1/(x(k)^2+3)+w(k)   donde  x0-->N(0,1)
%                          f(x)=1/(x^2+3)

%Observaciones:         y(k)= gamma(k)*(x(k)^2+exp{x(k)})+v(k)  donde
%                          h(x)=x^2+exp{x}
%                          gamma(k)-->B(p)

%   P[w(k)=1]=15/18      P[w(k)=-3]=2/18      P[w(k)=-9]=1/18
%   P[v(k)=1]=15/18      P[v(k)=-3]=2/18      P[v(k)=-9]=1/18

n_simul=1000; %número de simulaciones
n_iter=50;    %número de iteraciones

for s=1:n_simul

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%% Generación de la señal x(k) %%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    P0=1;          %varianza de la señal inicial
    P30=0;         %momento centrado de tercer orden de la señal inicial
    P40=3*P0^2-P0^2; %momento centrado de cuarto orden de la señal inicial

    x0(s)=P0^(1/2)*randn(1,1);

    Q=19/3;        %varianza del ruido de la ecuación de la señal w(k)
    Q3=-128/3;    %momento centrado de tercer orden del ruido w(k)
    Q4=1123/3-Q^2; %momento centrado de cuarto orden del ruido w(k)

    %% Condiciones iniciales %%

    a=rand(1,1);
    if a <= 1/18
        w0(s)=-9;
    elseif a <= 3/18
        w0(s)=-3;
    else w0(s)=1;
    end

    x(s,1)=1/(x0(s)^2+3)+w0(s);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```
for k=1:n_iter

    b=rand(1,1);
    if b <= 1/18
        w(s,k)=-9;
    elseif b <= 3/18
        w(s,k)=-3;
    else w(s,k)=1;
    end
    x(s,k+1)=1/(x(s,k)^2+3)+w(s,k);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Generación de las observaciones y(k) %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

p=0.5; %parámetro de la distribución Bernoulli

R=19/3;          %varianza del ruido de la ecuación observación v(k)
R3=-128/3;      %momento centrado de tercer orden del ruido v(k)
R4=1123/3-R^2;  %momento centrado de cuarto orden del ruido v(k)

for k=1:n_iter

    gamma(s,k)=binornd(1,p);
    c=rand(1,1);
    if c <= 1/18
        v(s,k)=-9;
    elseif c <= 3/18
        v(s,k)=-3;
    else v(s,k)=1;
    end
    y(s,k)=gamma(s,k)*(x(s,k)^2+exp(x(s,k)))+v(s,k);

end
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO EXTENDIDO CUADRÁTICO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Fil=filtro del sistema aumentado
%Pred=predicador del sistema aumentado
%CFil=matriz de covarianzas de Fil
%CPred=matriz de covarianzas de Pred
%filtro=filtro extendido cuadrático (FEC)
%predicador=predicador extendido cuadrático (PEC)
%FQ=derivada de la función f en el FEC
%FF= matriz de la ecuación de la señal aumentada
%HQ=derivada de la función h en el PEC
%HH=matriz de la ecuación observación aumentada
%d=esperanza de la señal linealizada
%D=varianza de la señal linealizada
%vv=ruido del sistema linealizado
%r=varianza de vv
%r3=momento centrado de tercer orden de vv
%r4=momento centrado de cuarto orden de vv
%QQ=matriz de covarianzas del ruido de la ecuación de la señal
%  aumentada
%Q_ij=componente ij de la matriz QQ
%VV=ruido del sistema aumentado
%V=predicador de VV
%RR=matriz de covarianzas de VV
%R_ij=componente ij de la matriz RR
%innov=innovación
%Pi=matriz de covarianzas de la innovación
%KK=ganancia del sistema aumentado
%G=matriz de ganancia del predictor de VV
%Px=matriz de covarianzas de la señal aumentada
%Pv=matriz de covarianzas de V
%Pvx=matriz de covarianzas cruzadas de los errores de predicción de
%  la señal y el ruido
%DC_filtro=cuadrado de la diferencia entre la señal y el FEC

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%% Condiciones iniciales %%%%%%

%%% k=0

%filtro

Fil0(:,s)=[0;0];
CFil0(:,s)=[P0 P30;P30' P40];

```

```

d0(s)=0;      %esperanza de x_0
filtro0(s)=Fil0(1,s)+d0(s);

%predictor de la señal

FQ0(s)=(-2)*filtro0(s)/(filtro0(s)^2+3)^2;
FF0(:, :, s)=[FQ0(s) 0;0 FQ0(s)^2];
Pred(:, s, 1)=FF0(:, :, s)*Fil0(:, s);

D0(s)=P0;      %E[(x0-E[x0]) (x0-E[x0])^T]
Q0_22=4*FQ0(s)^2*D0(s)*Q+Q4;
QQ0(:, :, s)=[Q Q3;Q3' Q0_22];
CPred(:, :, s, 1)=FF0(:, :, s)*CFil0(:, :, s)*FF0(:, :, s)'+QQ0(:, :, s);

u0(s)=1/(filtro0(s)^2+3)-FQ0(s)*filtro0(s);
d(s, 1)=FQ0(s)*d0(s)+u0(s);      %E[x(1)]
predictor(s, 1)= Pred(1, s, 1)+d(s, 1);

%%% k=1

%observación aumentadas

D(s, 1)=FQ0(s)^2*D0(s)+Q; %E[(x(1)-E[x(1)])(x(1)-E[x(1)])']
X(:, s, 1)=[x(s, 1); x(s, 1)^2];
EX(:, s, 1)=[d(s, 1); D(s, 1)];
XX(:, s, 1)=X(:, s, 1)-EX(:, s, 1);

HQ(s, 1)=2*predictor(s, 1)+exp(predictor(s, 1));
HH(:, :, s, 1)=[HQ(s, 1) 0;0 HQ(s, 1)^2];

z(s, 1)=predictor(s, 1)^2+exp(predictor(s, 1))-HQ(s, 1)*predictor(s, 1);
c(s, 1)=HQ(s, 1)*d(s, 1)+z(s, 1);
vv(s, 1)=(gamma(s, 1)-p)*c(s, 1)+v(s, 1);
Psi(s, 1)=2*HQ(s, 1)*XX(1, s, 1)*gamma(s, 1)*vv(s, 1)+vv(s, 1)^2;

r(s, 1)=p*(1-p)*c(s, 1)^2+R;
r3(s, 1)=p*(1-p)*(1-2*p)*c(s, 1)^3+R3;
r4(s, 1)=(p*(1-p)*(3*p^2-3*p+1)-p^2*(1-p)^2)*c(s, 1)^4+...
        R4+4*p*(1-p)*c(s, 1)^2*R;

VV(:, s, 1)=[vv(s, 1); Psi(s, 1)-r(s, 1)]+...
            (gamma(s, 1)-p)*HH(:, :, s, 1)*EX(:, s, 1);

Y(:, s, 1)=gamma(s, 1)*HH(:, :, s, 1)*XX(:, s, 1)+VV(:, s, 1);

```

```

%innovación, filtro

Innov(:,s,1)=Y(:,s,1)-p*HH(:,s,1)*Pred(:,s,1);

Px(:,s,1)=FF0(:,s,1)*CFil0(:,s,1)*FF0(:,s,1)'+QQ0(:,s,1);

R_11(s,1,1)=r(s,1);
R_12(s,1,1)=r3(s,1)+p*(1-p)*c(s,1)*D(s,1)*HQ(s,1)^2;
R_22(s,1,1)=4*HQ(s,1)^2*D(s,1)*(p*(1-p)^2*c(s,1)^2+p*R)+...
            r4(s,1)+p*(1-p)*(1-2*p)*(c(s,1)^2*D(s,1)*HQ(s,1)^2+...
            HQ(s,1)^2*D(s,1)*c(s,1)^2)+p*(1-p)*HQ(s,1)^4*D(s,1)^2;
RR(:,s,1,1)=[R_11(s,1,1) R_12(s,1,1);R_12(s,1,1)' R_22(s,1,1)];

Pv0(:,s,1,1)=RR(:,s,1,1);

Pi(:,s,1)=p*(1-p)*HH(:,s,1)*Px(:,s,1)*HH(:,s,1)'+...
          p^2*HH(:,s,1)*CPred(:,s,1)*HH(:,s,1)'+...
          Pv0(:,s,1,1);

KK(:,s,1)=p*CPred(:,s,1)*HH(:,s,1)'^*Pi(:,s,1)^(-1);

Fil(:,s,1)=Pred(:,s,1)+KK(:,s,1)*Innov(:,s,1);
CFil(:,s,1)=CPred(:,s,1)-KK(:,s,1)*Pi(:,s,1)*KK(:,s,1)';
filtro(s,1)=Fil(1,s,1)+d(s,1);
DC_filtro(s,1)=(x(s,1)-filtro(s,1))^2;

%predicador de la señal

FQ(s,1)=(-2)*filtro(s,1)/(filtro(s,1)^2+3)^2;
FF(:,s,1)=[FQ(s,1) 0;0 FQ(s,1)^2];
Pred(:,s,2)=FF(:,s,1)*Fil(:,s,1);

Q_22(s,1)=4*FQ(s,1)^2*D(s,1)*Q+Q4;
QQ(:,s,1)=[Q Q3;Q3' Q_22(s,1)];
CPred(:,s,2)=FF(:,s,1)*CFil(:,s,1)*FF(:,s,1)'+QQ(:,s,1);

u(s,1)=1/(filtro(s,1)^2+3)-FQ(s,1)*filtro(s,1);
d(s,2)=FQ(s,1)*d(s,1)+u(s,1);
predicador(s,2)= Pred(1,s,2)+d(s,2);

```

```

%%% k=2

%observación aumentadas

D(s,2)=FQ(s,1)^2*D(s,1)+Q; %E[(x(2)-E[x(2)])(x(2)-E[x(2)])']
X(:,s,2)=[x(s,2); x(s,2)^2];
EX(:,s,2)=[d(s,2);D(s,2)];

XX(:,s,2)=X(:,s,2)-EX(:,s,2);

HQ(s,2)=2*predicador(s,2)+exp(predicador(s,2));
HH(:,s,2)=[HQ(s,2) 0;0 HQ(s,2)^2];

z(s,2)=predicador(s,2)^2+exp(predicador(s,2))-HQ(s,2)*predicador(s,2);
c(s,2)=HQ(s,2)*d(s,2)+z(s,2);
vv(s,2)=(gamma(s,2)-p)*c(s,2)+v(s,2);
Psi(s,2)=2*HQ(s,2)*XX(1,s,2)*gamma(s,2)*vv(s,2)+vv(s,2)^2;

r(s,2)=p*(1-p)*c(s,2)^2+R;
r3(s,2)=p*(1-p)*(1-2*p)*c(s,2)^3+R3;
r4(s,2)=(p*(1-p)*(3*p^2-3*p+1)-p^2*(1-p)^2)*c(s,2)^4+...
        R4+4*p*(1-p)*c(s,2)^2*R;

VV(:,s,2)=[vv(s,2); Psi(s,2)-r(s,2)]+...
        (gamma(s,2)-p)*HH(:,s,2)*EX(:,s,2);

Y(:,s,2)=gamma(s,2)*HH(:,s,2)*XX(:,s,2)+VV(:,s,2);

%predicador del ruido (algoritmo en j)

%condiciones iniciales j=0

V0(:,s,2)=[0;0];
Pvx0(:,s,2,1)=[0 0;0 0];

for i=1:2
    if i==2
        R_11(s,2,i)=p*(1-p)*c(s,2)^2+R;
        R_12(s,2,i)=r3(s,2)+p*(1-p)*c(s,2)*D(s,2)*HQ(s,2)^2;
        R_22(s,2,i)=4*HQ(s,2)^2*D(s,2)*(p*(1-p)^2*c(s,2)^2+p*R)+...
            r4(s,2)+p*(1-p)*(1-2*p)*(c(s,2)^2*D(s,2)*HQ(s,2)^2+...
            HQ(s,2)^2*D(s,2)*c(s,2)^2)+p*(1-p)*HQ(s,2)^4*D(s,2)^2;
    else
        R_11(s,2,i)=0;
        R_12(s,2,i)=0;
        R_22(s,2,i)=4*HQ(s,2)^2*D(s,2)*p^2*(1-p)^2*c(s,2)^2;
    end
end

```

```

RR(:,:,s,2,i)=[R_11(s,2,i) R_12(s,2,i);R_12(s,2,i)' R_22(s,2,i)];
Pv0(:,:,s,2,i)=RR(:,:,s,2,i);
end %i

```

---

```

G(:,:,s,2,1)=(p*Pvx0(:,:,s,2,1)*HH(:,:,s,1)'+...
Pv0(:,:,s,2,1))*Pi(:,:,s,1)^(-1);

```

---

```

V(:,s,2,1)=V0(:,s,2)+G(:,:,s,2,1)*Innov(:,s,1);

```

---

```

%innovación, filtro

```

```

Innov(:,s,2)=Y(:,s,2)-p*HH(:,:,s,2)*Pred(:,s,2)-V(:,s,2,1);

```

```

Px(:,:,s,2)=FF(:,:,s,1)*Px(:,:,s,1)*FF(:,:,s,1)'+QQ(:,:,s,1);

```

```

Pv(:,:,s,2,2,1)=Pv0(:,:,s,2,2,1)-G(:,:,s,2,1)*Pi(:,:,s,1)*...
G(:,:,s,2,1)';

```

```

Pvx(:,:,s,2,2,1)=(Pvx0(:,:,s,2,1)-...
G(:,:,s,2,1)*Pi(:,:,s,1)*KK(:,:,s,1)')*FF(:,:,s,1)';

```

```

Pi(:,:,s,2)=p*(1-p)*HH(:,:,s,2)*Px(:,:,s,2)*HH(:,:,s,2)'+...
p^2*HH(:,:,s,2)*CPred(:,:,s,2)*HH(:,:,s,2)'+...
Pv(:,:,s,2,2,1)+p*HH(:,:,s,2)*Pvx(:,:,s,2,2,1)'+...
p*Pvx(:,:,s,2,2,1)*HH(:,:,s,2)';

```

```

KK(:,:,s,2)=(p*CPred(:,:,s,2)*HH(:,:,s,2)'+...
Pvx(:,:,s,2,2,1)')*Pi(:,:,s,2)^(-1);

```

```

Fil(:,s,2)=Pred(:,s,2)+KK(:,:,s,2)*Innov(:,s,2);
CFil(:,:,s,2)=CPred(:,:,s,2)-KK(:,:,s,2)*Pi(:,:,s,2)*KK(:,:,s,2)';
filtro(s,2)=Fil(1,s,2)+d(s,2);
DC_filtro(s,2)=(x(s,2)-filtro(s,2))^2;

```

```

%predicador de la señal

```

```

FQ(s,2)=(-2)*filtro(s,2)/(filtro(s,2)^2+3)^2;
FF(:,:,s,2)=[FQ(s,2) 0;0 FQ(s,2)^2];
Pred(:,s,3)=FF(:,:,s,2)*Fil(:,s,2);

```



```

Q_22(s,2)=4*FQ(s,2)^2*D(s,2)*Q+Q4;
QQ(:,:,s,2)=[Q Q3;Q3' Q_22(s,2)];
CPred(:,:,s,3)=FF(:,:,s,2)*CFil(:,:,s,2)*FF(:,:,s,2)'+QQ(:,:,s,2);

u(s,2)=1/(filtro(s,2)^2+3)-FQ(s,2)*filtro(s,2);
d(s,3)=FQ(s,2)*d(s,2)+u(s,2);
predictor(s,3)= Pred(1,s,3)+d(s,3);

for k=3:n_iter

%observación aumentadas

D(s,k)=FQ(s,k-1)^2*D(s,k-1)+Q; %E[(x(k)-E[x(k)])(x(k)-E[x(k)])']
X(:,s,k)=[x(s,k); x(s,k)^2];
EX(:,s,k)=[d(s,k);D(s,k)];
XX(:,s,k)=X(:,s,k)-EX(:,s,k);

HQ(s,k)=2*predictor(s,k)+exp(predictor(s,k));
HH(:,:,s,k)=[HQ(s,k) 0;0 HQ(s,k)^2];

z(s,k)=predictor(s,k)^2+exp(predictor(s,k))-HQ(s,k)*predictor(s,k);
c(s,k)=HQ(s,k)*d(s,k)+z(s,k);
vv(s,k)=(gamma(s,k)-p)*c(s,k)+v(s,k);
Psi(s,k)=2*HQ(s,k)*XX(1,s,k)*gamma(s,k)*vv(s,k)+vv(s,k)^2;

r(s,k)=p*(1-p)*c(s,k)^2+R;
r3(s,k)=p*(1-p)*(1-2*p)*c(s,k)^3+R3;
r4(s,k)=(p*(1-p)*(3*p^2-3*p+1)-p^2*(1-p)^2)*c(s,k)^4+...
        R4+4*p*(1-p)*c(s,k)^2*R;

VV(:,s,k)=[vv(s,k); Psi(s,k)-r(s,k)]+...
        (gamma(s,k)-p)*HH(:,:,s,k)*EX(:,s,k);

Y(:,s,k)=gamma(s,k)*HH(:,:,s,k)*XX(:,s,k)+VV(:,s,k);

%predictor del ruido (algoritmo en j)

%condiciones iniciales j=0

V0(:,s,k)=[0;0];
Pvx0(:,:,s,k,1)=[0 0;0 0];

```

```

for i=1:k
    if i==k
        R_11(s,k,i)=p*(1-p)*c(s,k)^2+R;
        R_12(s,k,i)=r3(s,k)+p*(1-p)*c(s,k)*D(s,k)*HQ(s,k)^2;
        R_22(s,k,i)=4*HQ(s,k)^2*D(s,k)*(p*(1-p)^2*c(s,k)^2+p*R)+...
            r4(s,k)+p*(1-p)*(1-2*p)*(c(s,k)^2*D(s,k)*HQ(s,k)^2+...
            HQ(s,k)^2*D(s,k)*c(s,k)^2)+p*(1-p)*HQ(s,k)^4*D(s,k)^2;
    else
        R_11(s,k,i)=0;
        R_12(s,k,i)=0;
        R_22(s,k,i)=4*HQ(s,k)^2*D(s,k)*p^2*(1-p)^2*c(s,k)^2;
    end

    RR(:,:,s,k,i)=[R_11(s,k,i) R_12(s,k,i);R_12(s,k,i)' R_22(s,k,i)];
    Pv0(:,:,s,k,i)=RR(:,:,s,k,i);
end %i

%
G(:,:,s,k,1)=(p*Pvx0(:,:,s,k,1)*HH(:,:,s,1)'+...
    Pv0(:,:,s,k,1))*Pi(:,:,s,1)^(-1);

%
Pv(:,:,s,k,2,1)=Pv0(:,:,s,k,2)-...
    G(:,:,s,k,1)*Pi(:,:,s,1)*G(:,:,s,2,1)';
Pv(:,:,s,k,k,1)=Pv0(:,:,s,k,k)-...
    G(:,:,s,k,1)*Pi(:,:,s,1)*G(:,:,s,k,1)';
Pvx(:,:,s,k,2,1)=(Pvx0(:,:,s,k,1)-...
    G(:,:,s,k,1)*Pi(:,:,s,1)*KK(:,:,s,1)')*FF(:,:,s,1)';

for j=3:k-1
    Pv(:,:,s,k,j,1)=Pv0(:,:,s,k,j)-...
        G(:,:,s,k,1)*Pi(:,:,s,1)*G(:,:,s,j,1)';
    for l=2:j-1
        Pv(:,:,s,k,j,l)=Pv(:,:,s,k,j,l-1)-...
            G(:,:,s,k,1)*Pi(:,:,s,1)*G(:,:,s,j,l)';
    end %l
    Pvx(:,:,s,k,j,j-1)=(Pvx(:,:,s,k,j-1,j-2)-...
        G(:,:,s,k,j-1)*Pi(:,:,s,j-1)*KK(:,:,s,j-1)')*FF(:,:,s,j-1)';
end %j

%
for j=2:k-1
    G(:,:,s,k,j)=(p*Pvx(:,:,s,k,j,j-1)*HH(:,:,s,j)'+...
        Pv(:,:,s,k,j,j-1))*Pi(:,:,s,j)^(-1);
end

%

```

```

V(:,s,k,1)=V0(:,s,k)+G(:, :,s,k,1)*Innov(:,s,1);

for j=2:k-1
    V(:,s,k,j)=V(:,s,k,j-1)+G(:, :,s,k,j)*Innov(:,s,j);
end
%
%innovación, filtro

Innov(:,s,k)=Y(:,s,k)-p*HH(:, :,s,k)*Pred(:,s,k)-V(:,s,k,k-1);

Px(:, :,s,k)=FF(:, :,s,k-1)*Px(:, :,s,k-1)*FF(:, :,s,k-1)'+QQ(:, :,s,k-1);

for j=2:k-1
    Pv(:, :,s,k,k,j)=Pv(:, :,s,k,k,j-1)-...
        G(:, :,s,k,j)*Pi(:, :,s,j)*G(:, :,s,k,j)';
end

Pvx(:, :,s,k,k,k-1)=(Pvx(:, :,s,k,k-1,k-2)-...
    G(:, :,s,k,k-1)*Pi(:, :,s,k-1)*KK(:, :,s,k-1)')*FF(:, :,s,k-1)';

Pi(:, :,s,k)=p*(1-p)*HH(:, :,s,k)*Px(:, :,s,k)*HH(:, :,s,k)'+...
    p^2*HH(:, :,s,k)*CPred(:, :,s,k)*HH(:, :,s,k)'+...
    Pv(:, :,s,k,k,k-1)+p*HH(:, :,s,k)*Pvx(:, :,s,k,k,k-1)'+...
    p*Pvx(:, :,s,k,k,k-1)*HH(:, :,s,k)';

KK(:, :,s,k)=(p*CPred(:, :,s,k)*HH(:, :,s,k)'+...
    Pvx(:, :,s,k,k,k-1)')*Pi(:, :,s,k)^(-1);
Fil(:,s,k)=Pred(:,s,k)+KK(:, :,s,k)*Innov(:,s,k);
CFil(:, :,s,k)=CPred(:, :,s,k)-KK(:, :,s,k)*Pi(:, :,s,k)*KK(:, :,s,k)';
filtro(s,k)=Fil(1,s,k)+d(s,k);
DC_filtro(s,k)=(x(s,k)-filtro(s,k))^2;

%predictor de la señal

FQ(s,k)=(-2)*filtro(s,k)/(filtro(s,k)^2+3)^2;
FF(:, :,s,k)=[FQ(s,k) 0;0 FQ(s,k)^2];
Pred(:,s,k+1)=FF(:, :,s,k)*Fil(:,s,k);

Q_22(s,k)=4*FQ(s,k)^2*D(s,k)*Q+Q4;
QQ(:, :,s,k)=[Q Q3;Q3' Q_22(s,k)];
CPred(:, :,s,k+1)=FF(:, :,s,k)*CFil(:, :,s,k)*FF(:, :,s,k)'+QQ(:, :,s,k);

u(s,k)=1/(filtro(s,k)^2+3)-FQ(s,k)*filtro(s,k);
d(s,k+1)=FQ(s,k)*d(s,k)+u(s,k);
predictor(s,k+1)= Pred(1,s,k+1)+d(s,k+1);

end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILTRO MIXTURAS LINEALIZADO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fil2=filtro mixturas linealizado (FML)
%pred2=predictor mixturas linealizado(PML)
%Cfil2=varianza del FML
%Cpred2=varianza del PML
%F=derivada de la función f en el FML
%H=derivada de la función h en el PML
%K2=ganancia del FML
%DC_fil2=cuadrado de la diferencia entre la señal y el FML

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Condiciones iniciales %Condiciones

fil02(s)=0;
Cfil02(s)=P0;
pred2(s,1)=1/(fil02(s)^2+3);
F0(s)=(-2)*fil02(s)/(fil02(s)^2+3)^2;
Cpred2(s,1)=F0(s)*Cfil02(s)*F0(s)'+Q;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    H(s,k)=2*pred2(s,k)+exp(pred2(s,k));
    Cinnov(s,k)=p*H(s,k)*Cpred2(s,k)*H(s,k)'+...
        p*(1-p)*(pred2(s,k)^2+exp(pred2(s,k)))*(pred2(s,k)^2+...
        exp(pred2(s,k)))'+R;
    K2(s,k)=p*Cpred2(s,k)*H(s,k)'^Cinnov(s,k)^(-1);
    fil2(s,k)=pred2(s,k)+K2(s,k)*(y(s,k)- p*(pred2(s,k)^2+...
        exp(pred2(s,k))));
    Cfil2(s,k)=(1-p*K2(s,k)*H(s,k))*Cpred2(s,k);
    pred2(s,k+1)=1/(fil2(s,k)^2+3);
    F(s,k)=(-2)*fil2(s,k)/(fil2(s,k)^2+3)^2;
    Cpred2(s,k+1)=F(s,k)*Cfil2(s,k)*F(s,k)'+Q;
    DC_fil2(s,k)=(x(s,k)-fil2(s,k))^2;

end %iter

end %sim

```

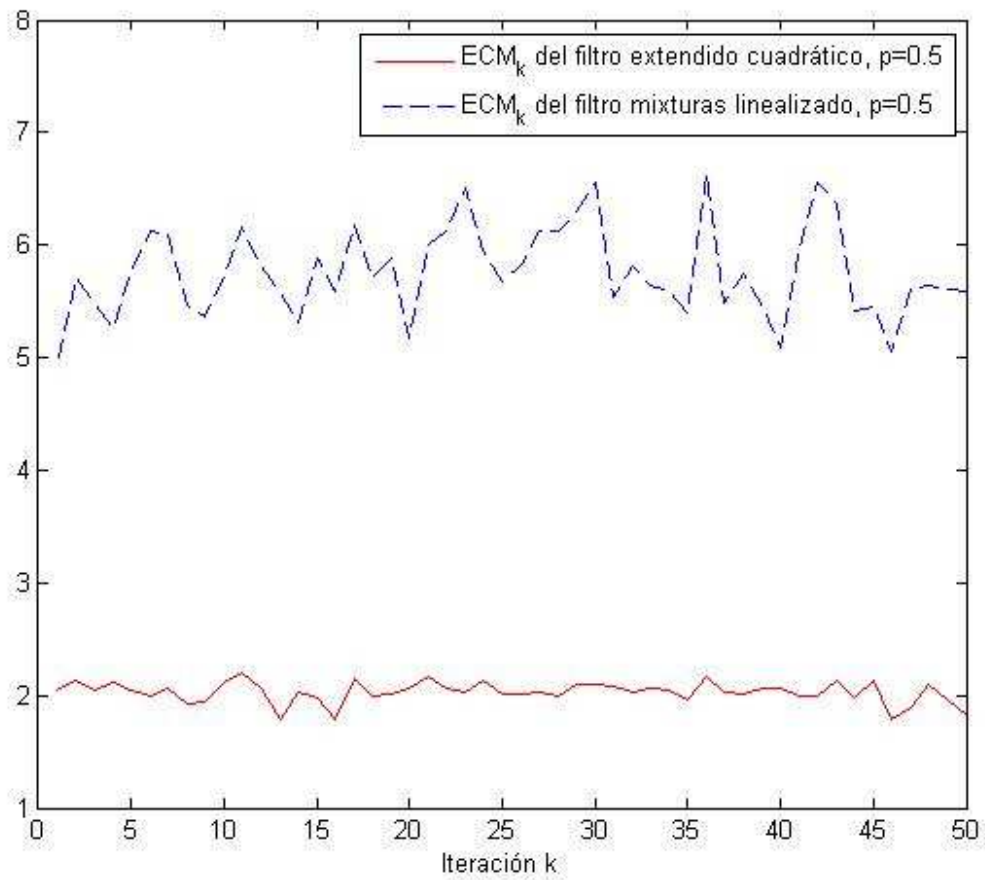
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Errores cuadráticos medios del FEC y FML %%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
for k=1:n_iter  
    SCE_filtro(1,k)=DC_filtro(1,k);  
    SCE_fil2(1,k)=DC_fil2(1,k);  
    for s=2:n_simul  
        SCE_filtro(s,k)=SCE_filtro(s-1,k)+DC_filtro(s,k);  
        SCE_fil2(s,k)=SCE_fil2(s-1,k)+DC_fil2(s,k);  
    end  
    ECM_filtro(k)=sqrt(SCE_filtro(n_simul,k)/n_simul);  
    ECM_fil2(k)=sqrt(SCE_fil2(n_simul,k)/n_simul);  
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Gráfica de los errores cuadráticos medios %%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
for k=1:n_iter  
    time(k)=k;  
    pECM_filtro(k)=ECM_filtro(k);  
    pECM_fil2(k)=ECM_fil2(k);  
end
```

```
graf(3.1)=plot(time,pECM_filtro,'r-',time,pECM_fil2,'b--');  
xlabel('Iteración k')  
legend('ECM_k del filtro cuadrático, p=0.5',...  
      'ECM_k del filtro mixturas linealizado, p=0.5')  
saveas(gcf, 'Fig3_1', 'fig')
```



### CAPÍTULO 3: Figura 3.3

```

%Modelo de la Señal:      x(k+1)=1/(x(k)^2+3)+w(k)   donde  x0-->N(0,1)
%                          f(x)=1/(x^2+3)

%Observaciones:          y(k)= gamma(k)*(x(k)^2+exp{x(k)})+v(k)  donde
%                          h(x)=x^2+exp{x}
%                          gamma(k)-->B(p)

%   P[w(k)=1]=15/18      P[w(k)=-3]=2/18      P[w(k)=-9]=1/18
%   P[v(k)=1]=15/18      P[v(k)=-3]=2/18      P[v(k)=-9]=1/18

n_simul=1000; %número de simulaciones
n_iter=50;    %número de iteraciones

for s=1:n_simul

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%% Generación de la señal x(k) %%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    P0=1;          %varianza de la señal inicial
    P3=0;          %momento centrado de tercer orden de la señal inicial
    P4=3*P0^2-P0^2; %momento centrado de cuarto orden de la señal inicial

    x0(s)=P0^(1/2)*randn(1,1);

    Q=19/3;        %varianza del ruido de la ecuación de la señal w(k)
    Q3=-128/3;    %momento centrado de tercer orden del ruido w(k)
    Q4=1123/3-Q^2; %momento centrado de cuarto orden del ruido w(k)

    %% Condiciones iniciales %%

    a=rand(1,1);
    if a <= 1/18
        w0(s)=-9;
    elseif a <= 3/18
        w0(s)=-3;
    else w0(s)=1;
    end

    x(s,1)=1/(x0(s)^2+3)+w0(s);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

for k=1:n_iter

    b=rand(1,1);
    if b <= 1/18
        w(s,k)=-9;
    elseif b <= 3/18
        w(s,k)=-3;
    else w(s,k)=1;
    end

    x(s,k+1)=1/(x(s,k)^2+3)+w(s,k);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Generación de las observaciones y(k) %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

p=0.5; %parámetro de la distribución Bernoulli

R=19/3;          %varianza del ruido de la ecuación observación v(k)
R3=-128/3;      %momento centrado de tercer orden del ruido v(k)
R4=1123/3-R^2; %momento centrado de cuarto orden del ruido v(k)

for k=1:n_iter

    gamma(s,k)=binornd(1,p);
    c=rand(1,1);
    if c <= 1/18
        v(s,k)=-9;
    elseif c <= 3/18
        v(s,k)=-3;
    else v(s,k)=1;
    end

    y(s,k)=gamma(s,k)*(x(s,k)^2+exp(x(s,k)))+v(s,k);

end

```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO ENSEMBLE %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%fil=filtro ensemble (FEn)
%pred=predictor ensemble (PEn)
%Cfil=cuasivarianza del FEn
%Cpred=cuasivarianza del PEn
%K=ganancia del FEn
%DC_fil=cuadrado de la diferencia entre la señal y el FEn

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Generación del ensemble de filtrado y predicción %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %efil=conjunto de filtros
    %epred=conjunto de predictores
    %ew=conjunto de valores simulados a partir de la distribución de w
    %ev=conjunto de valores simulados a partir de la distribución de v
    %egamma=conjunto de valores simulados a partir de la distribución del
    %      ruido multiplicativo de la ecuación de observación
    %obs=conjunto de predictores de las observaciones
    %yest=media de obs
    %Cyy=cuasivarianza de obs
    %Cxy=cuasivarianza del PEn y obs

q=500; %tamaño del conjunto de valores simulados

% Simulación del ensemble de filtrado y predicción inicial %
for j=1:q
    efil0(s,j)=P0^(1/2)*randn(1,1);

    d=rand(1,1);
    if d <= 1/18
        ew0(s,j)=-9;
    elseif d <= 3/18
        ew0(s,j)=-3;
    else ew0(s,j)=1;
    end

    epred(s,1,j)= 1/(efil0(s,j)^2+3)+ew0(s,j);

end

% Cálculo del PEn y su cuasivarianza para k=1 %%

pred(s,1)=mean(epred(s,1,:));
Cpred(s,1)=var(epred(s,1,:));

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del FEn y su cuasivarianza para k %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    %% Cálculo de los predictores de las observaciones en k %%

    for j=1:q

        egamma(s,k,j)=binornd(1,p);

        e=rand(1,1);
        if e <= 1/18
            ev(s,k,j)=-9;
        elseif e <= 3/18
            ev(s,k,j)=-3;
        else ev(s,k,j)=1;
        end

        obs(s,k,j)=egamma(s,k,j)*(epred(s,k,j)^2+...
            exp(epred(s,k,j)))+ev(s,k,j);

    end

    yest(s,k)=mean(obs(s,k,:));
    Cyy(s,k)=var(obs(s,k,:));

    %% Cálculo de la cuasicovarianza del PEn y obs en k %%

    restapred(s,k,1)=epred(s,k,1)-pred(s,k);
    restayy(s,k,1)=obs(s,k,1)-yest(s,k);
    Cxysum(s,k,1)=restapred(s,k,1)*restayy(s,k,1);

    for j=2:q
        restapred(s,k,j)=epred(s,k,j)-pred(s,k);
        restayy(s,k,j)=obs(s,k,j)-yest(s,k);
        Cxysum(s,k,j)=Cxysum(s,k,j-1)+...
            (restapred(s,k,j)*restayy(s,k,j));
    end

    Cxy(s,k)=Cxysum(s,k,q)/(q-1);

    %% Cálculo del FEn y su cuasivarianza en k %%

    K(s,k)=Cxy(s,k)*Cyy(s,k)^(-1);

```

```
for j=1:q
    efil(s,k,j)=epred(s,k,j)+K(s,k)*(y(s,k)-obs(s,k,j));
end

fil(s,k)=mean(efil(s,k,:));
Cfil(s,k)=var(efil(s,k,:));
DC_fil(s,k)=(x(s,k)-fil(s,k))^2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del PEn y su cuasivarianza para k+1 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for j=1:q

    n=rand(1,1);
    if n <= 1/18
        ew(s,k,j)=-9;
    elseif n <= 3/18
        ew(s,k,j)=-3;
    else ew(s,k,j)=1;
    end

    epred(s,k+1,j)= 1/(efil(s,k,j)^2+3)+ew(s,k,j);

end

pred(s,k+1)=mean(epred(s,k+1,:));
Cpred(s,k+1)=var(epred(s,k+1,:));

end %iter
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO MIXTURAS LINEALIZADO %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%fil2=filtro mixturas linealizado (FML)
%pred2=predictor mixturas linealizado (PML)
%Cfil2=varianza del FML
%Cpred2=varianza del PML
%F=derivada de la función f en el FML
%H=derivada de la función h en el PML
%K2=ganancia del FML
%DC_fil2=cuadrado de la diferencia entre la señal y el FML

%%%%%% Condiciones iniciales %%%%%%

fil02(s)=0;
Cfil02(s)=P0;
pred2(s,1)=1/(fil02(s)^2+3);
F0(s)=(-2)*fil02(s)/(fil02(s)^2+3)^2;
Cpred2(s,1)=F0(s)*Cfil02(s)*F0(s)'+Q;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    H(s,k)=2*pred2(s,k)+exp(pred2(s,k));
    Cinnov(s,k)=p*H(s,k)*Cpred2(s,k)*H(s,k)'+...
                p*(1-p)*(pred2(s,k)^2+...
                exp(pred2(s,k)))*(pred2(s,k)^2+...
                exp(pred2(s,k)))'+R;
    K2(s,k)=p*Cpred2(s,k)*H(s,k) '*Cinnov(s,k)^(-1);

    fil2(s,k)=pred2(s,k)+K2(s,k)*(y(s,k)- p*(pred2(s,k)^2+...
                exp(pred2(s,k))));
    Cfil2(s,k)=(1-p*K2(s,k)*H(s,k))*Cpred2(s,k);

    F(s,k)=(-2)*fil2(s,k)/(fil2(s,k)^2+3)^2;

    pred2(s,k+1)=1/(fil2(s,k)^2+3);

    Cpred2(s,k+1)=F(s,k)*Cfil2(s,k)*F(s,k)'+Q;

    DC_fil2(s,k)=(x(s,k)-fil2(s,k))^2;

end %iter

end %simul
```

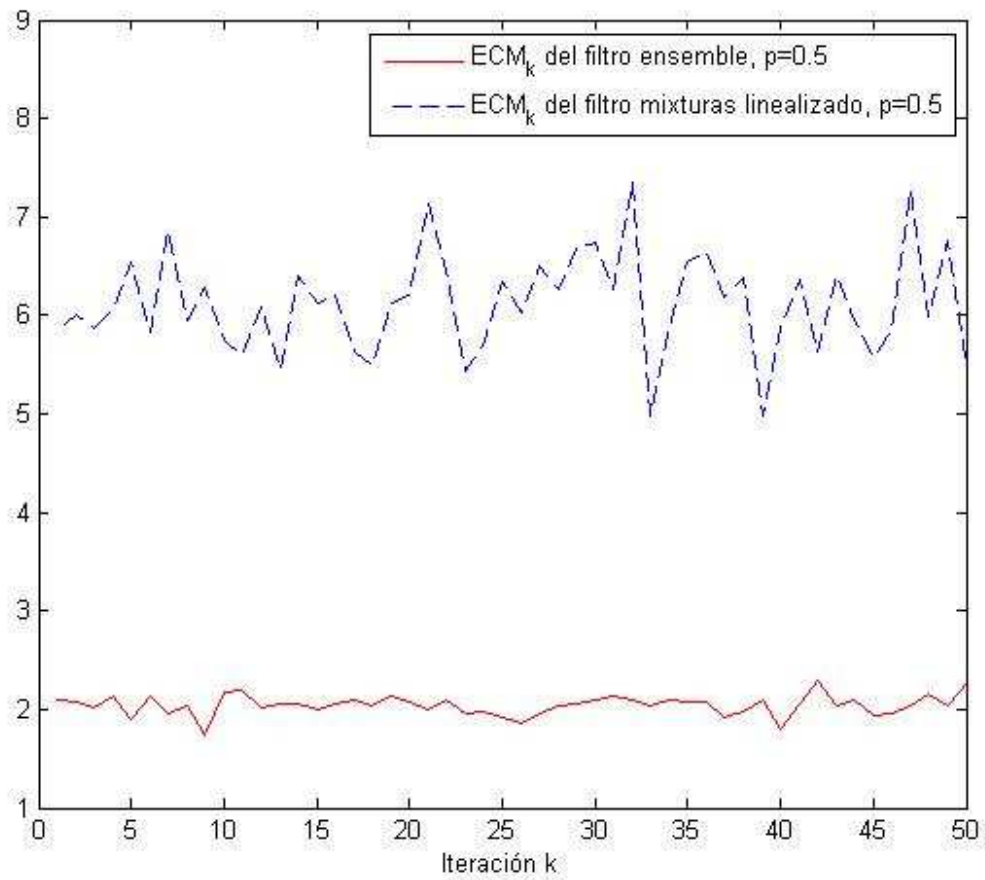
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Errores cuadráticos medios del FEn y FML %%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
for k=1:n_iter  
    SCE_fil(1,k)=DC_fil(1,k);  
    SCE_fil2(1,k)=DC_fil2(1,k);  
  
    for s=2:n_simul  
        SCE_fil(s,k)=SCE_fil(s-1,k)+DC_fil(s,k);  
        SCE_fil2(s,k)=SCE_fil2(s-1,k)+DC_fil2(s,k);  
    end  
  
    ECM_fil(k)=sqrt(SCE_fil(n_simul,k)/n_simul)  
    ECM_fil2(k)=sqrt(SCE_fil2(n_simul,k)/n_simul);  
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Gráfica de los errores cuadráticos medios %%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
for k=1:n_iter  
    time(k)=k;  
    pECM_fil(k)=ECM_fil(k);  
    pECM_fil2(k)=ECM_fil2(k);  
end
```

```
graf(3.3)=plot(time,pECM_fil,'r-',time,pECM_fil2,'b--');  
xlabel('Iteración k')  
legend('ECMk del filtro ensemble, p=0.5',...  
      'ECMk del filtro mixturas linealizado, p=0.5')  
saveas(gcf, 'Fig3_3', 'fig')
```



### CAPÍTULO 3: Figura 3.5

```

%Modelo de la Señal:      x(k+1)=1/(x(k)^2+3)+w(k)   donde  x0-->N(0,1)
%                          f(x)=1/(x^2+3)

%Observaciones:         y(k)= gamma(k)*(x(k)^2+exp{x(k)})+v(k)  donde
%                          h(x)=x^2+exp{x}
%                          gamma(k)-->B(p)

%   P[w(k)=1]=15/18      P[w(k)=-3]=2/18      P[w(k)=-9]=1/18
%   P[v(k)=1]=15/18      P[v(k)=-3]=2/18      P[v(k)=-9]=1/18

n_simul=1000; %número de simulaciones
n_iter=50;    %número de iteraciones

for s=1:n_simul

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%% Generación de la señal x(k) %%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    P0=1;          %varianza de la señal inicial
    P3=0;          %momento centrado de tercer orden de la señal inicial
    P4=3*P0^2-P0^2;%momento centrado de cuarto orden de la señal inicial

    x0(s)=P0^(1/2)*randn(1,1);

    Q=19/3;        %varianza del ruido de la ecuación de la señal w(k)
    Q3=-128/3;     %momento centrado de tercer orden del ruido w(k)
    Q4=1123/3-Q^2; %momento centrado de cuarto orden del ruido w(k)

    %% Concidiciones iniciales %%

    a=rand(1,1);
    if a <= 1/18
        w0(s)=-9;
    elseif a <= 3/18
        w0(s)=-3;
    else w0(s)=1;
    end

    x(s,1)=1/(x0(s)^2+3)+w0(s);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

for k=1:n_iter

    b=rand(1,1);
    if b <= 1/18
        w(s,k)=-9;
    elseif b <= 3/18
        w(s,k)=-3;
    else w(s,k)=1;
    end

    x(s,k+1)=1/(x(s,k)^2+3)+w(s,k);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%% Generación de las observaciones y(k) %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

p=0.5; %parámetro de la distribución Bernoulli

R=19/3;          %varianza del ruido de la ecuación observación v(k)
R3=-128/3;      %momento centrado de tercer orden del ruido v(k)
R4=1123/3-R^2; %momento centrado de cuarto orden del ruido v(k)

for k=1:n_iter

    gamma(s,k)=binornd(1,p);
    c=rand(1,1);
    if c <= 1/18
        v(s,k)=-9;
    elseif c <= 3/18
        v(s,k)=-3;
    else v(s,k)=1;
    end

    y(s,k)=gamma(s,k)*(x(s,k)^2+exp(x(s,k)))+v(s,k);

end

```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO ENSEMBLE %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%fil=filtro ensemble (FEn)
%pred=predictor ensemble (PEn)
%Cfil=cuasivarianza del FEn
%Cpred=cuasivarianza del PEn
%K=ganancia del FEn
%DC_fil=cuadrado de la diferencia entre la señal y el FEn

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Generación del ensemble de filtrado y predicción %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %efil=conjunto de filtros
    %epred=conjunto de predictores
    %ew=conjunto de valores simulados a partir de la distribución de w
    %ev=conjunto de valores simulados a partir de la distribución de v
    %egamma=conjunto de valores simulados a partir de la distribución del
    %      ruido multiplicativo de la ecuación de observación
    %obs=conjunto de predictores de las observaciones
    %yest=media de obs
    %Cyy=cuasivarianza de obs
    %Cxy=cuasivarianza del PEn y obs

q=500; %tamaño del conjunto de valores simulados

% Simulación del ensemble de filtrado y predicción inicial %
for j=1:q
    efil0(s,j)=P0^(1/2)*randn(1,1);

    d=rand(1,1);
    if d <= 1/18
        ew0(s,j)=-9;
    elseif d <= 3/18
        ew0(s,j)=-3;
    else ew0(s,j)=1;
    end

    epred(s,1,j)= 1/(efil0(s,j)^2+3)+ew0(s,j);

end

% Cálculo del PEn y su cuasivarianza para k=1 %%

pred(s,1)=mean(epred(s,1,:));
Cpred(s,1)=var(epred(s,1,:));

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%% Cálculo del FEn y su cuasivarianza para k %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    %% Cálculo de los predictores de las observaciones en k %%

    for j=1:q

        egamma(s,k,j)=binornd(1,p);

        e=rand(1,1);
        if e <= 1/18
            ev(s,k,j)=-9;
        elseif e <= 3/18
            ev(s,k,j)=-3;
        else ev(s,k,j)=1;
        end

        obs(s,k,j)=egamma(s,k,j)*(epred(s,k,j)^2+...
            exp(epred(s,k,j)))+ev(s,k,j);

    end

    yest(s,k)=mean(obs(s,k,:));
    Cyy(s,k)=var(obs(s,k,:));

    %% Cálculo de la cuasicovarianza del PEn y obs en k %%

    restapred(s,k,1)=epred(s,k,1)-pred(s,k);
    restayy(s,k,1)=obs(s,k,1)-yest(s,k);
    Cxysum(s,k,1)=restapred(s,k,1)*restayy(s,k,1);

    for j=2:q
        restapred(s,k,j)=epred(s,k,j)-pred(s,k);
        restayy(s,k,j)=obs(s,k,j)-yest(s,k);
        Cxysum(s,k,j)=Cxysum(s,k,j-1)+...
            (restapred(s,k,j)*restayy(s,k,j));
    end

    Cxy(s,k)=Cxysum(s,k,q)/(q-1);

    %% Cálculo del FEn y su cuasivarianza en k %%

    K(s,k)=Cxy(s,k)*Cyy(s,k)^(-1);

```

```

for j=1:q
    efil(s,k,j)=epred(s,k,j)+K(s,k)*(y(s,k)-obs(s,k,j));
end

fil(s,k)=mean(efil(s,k,:));
Cfil(s,k)=var(efil(s,k,:));
DC_fil(s,k)=(x(s,k)-fil(s,k))^2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del PEn y su cuasivarianza para k+1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for j=1:q

    n=rand(1,1);
    if n <= 1/18
        ew(s,k,j)=-9;
    elseif n <= 3/18
        ew(s,k,j)=-3;
    else ew(s,k,j)=1;
    end

    epred(s,k+1,j)= 1/(efil(s,k,j)^2+3)+ew(s,k,j);

end

pred(s,k+1)=mean(epred(s,k+1,:));
Cpred(s,k+1)=var(epred(s,k+1,:));

end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILTRO ENSEMBLE CUADRÁTICO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Fil=filtro ensemble cuadrático (FEnC)
%Pred=predicador ensemble cuadrático (PEnC)
%CFil=cuasivarianza del FEnC
%CPred=cuasivarianza del PEnC
%KK=ganancia del FEnC
%DC_Fil=cuadrado de la diferencia entre la señal y el FEnC

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Generación del ensemble de filtrado y predicción %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%EFil=conjunto de filtros
%EW=conjunto de valores simulados a partir de la distribución del
% ruido de la señal aumentada
%Phi=componente (1,2) de EW
%EV=conjunto de valores simulados a partir de la distribución del
% ruido de la observación aumentada
%Psi=componente (1,2) de EV
%EPred=conjunto de predictores
%Obs=conjunto de predictores de las observaciones
%Yest=media de Obs
%CYC=cuasicovarianza de Obs
%CCY=cuasicovarianza del PEnC y Obs

%% Simulación del ensemble de filtrado y predicción inicial %%

for j=1:q
    Efil0(:,s,j)=[efil0(s,j); efil0(s,j)^2];
    f0(s,j)=1/(efil0(s,j)^2+3);
    Phi0(s,j)=2*f0(s,j)*ew0(s,j)+ew0(s,j)^2;
    EW0(:,s,j)=[ew0(s,j); Phi0(s,j)];
    EPred(:,s,1,j)=[f0(s,j); f0(s,j)^2]+EW0(:,s,j);
end

%% Cálculo del predictor para k=1 %%

pred1(s,1)=mean(EPred(1,s,1,:));
pred2(s,1)=mean(EPred(2,s,1,:));
Pred(:,s,1)=[pred1(s,1);pred2(s,1)];

%% Cálculo de la matriz de cuasicovarianzas del predictor para k=1 %%

restaPred(:,s,1,1)=EPred(:,s,1,1)-Pred(:,s,1);
CPredsum(:,s,1,1)=restaPred(:,s,1,1)*restaPred(:,s,1,1)';

```

```

for j=2:q
    restaPred(:,s,1,j)=EPred(:,s,1,j)-Pred(:,s,1);
    CPredsum(:,s,1,j)=CPredsum(:,s,1,j-1)+...
        (restaPred(:,s,1,j)*restaPred(:,s,1,j)');
end

CPred(:,s,1)=CPredsum(:,s,1,q)/(q-1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del FEnC y su cuasicovarianza para k %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter

    % Cálculo de las observaciones estimadas en k %

    for j=1:q
        h(s,k,j)=EPred(1,s,k,j)^2+exp(EPred(1,s,k,j));
        Psi(s,k,j)=2*egamma(s,k,j)*h(s,k,j)*ev(s,k,j)+ev(s,k,j)^2;
        EV(:,s,k,j)=[ev(s,k,j); Psi(s,k,j)];

        Obs(:,s,k,j)= egamma(s,k,j)*...
            [h(s,k,j); h(s,k,j)^2]+EV(:,s,k,j);
    end

    yest1(s,k)=mean(Obs(1,s,k,:));
    yest2(s,k)=mean(Obs(2,s,k,:));
    Yest(:,s,k)=[yest1(s,k); yest2(s,k)];

    % Cálculo de la matriz de cuasicovarianzas del predictor de la %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% observación en k %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    restaYY(:,s,k,1)=Obs(:,s,k,1)-Yest(:,s,k);
    CYYsum(:,s,k,1)=restaYY(:,s,k,1)*restaYY(:,s,k,1)';

    for j=2:q
        restaYY(:,s,k,j)=Obs(:,s,k,j)-Yest(:,s,k);
        CYYsum(:,s,k,j)=CYYsum(:,s,k,j-1)+...
            (restaYY(:,s,k,j)*restaYY(:,s,k,j)');
    end
    CYY(:,s,k)=CYYsum(:,s,k,q)/(q-1);

    % Cálculo de la cuasicovarianza del PEnC y Obs en k %

    CXYsum(:,s,k,1)=restaPred(:,s,k,1)*restaYY(:,s,k,1)';

```

```

for j=2:q
    CXYsum(:, :, s, k, j)=CXYsum(:, :, s, k, j-1)+...
        (restaPred(:, s, k, j)*restaYY(:, s, k, j)');
end

CXY(:, :, s, k)=CXYsum(:, :, s, k, q)/(q-1);

%% Cálculo del FEnC y su cuasicovarianza en el instante k %%

Y(:, s, k)=[y(s, k); y(s, k)^2];

KK(:, :, s, k)=CXY(:, :, s, k)*CYY(:, :, s, k)^(-1);

for j=1:q
    EFil(:, s, k, j)=EPred(:, s, k, j)+KK(:, :, s, k)*...
        (Y(:, s, k)-Obs(:, s, k, j));
end

fill(s, k)=mean(EFil(1, s, k, :));
fil2(s, k)=mean(EFil(2, s, k, :));
Fil(:, s, k)=[fill(s, k); fil2(s, k)];

DC_Fil(s, k)=(x(s, k)-fill(s, k))^2;

restaFil(:, s, k, 1)=EFil(:, s, k, 1)-Fil(:, s, k);
CFilsum(:, :, s, k, 1)=restaFil(:, s, k, 1)*restaFil(:, s, k, 1)';

for j=2:q
    restaFil(:, s, k, j)=EFil(:, s, k, j)-Fil(:, s, k);
    CFilsum(:, :, s, k, j)=CFilsum(:, :, s, k, j-1)+...
        (restaFil(:, s, k, j)*restaFil(:, s, k, j)');
end

CFil(:, :, s, k)=CFilsum(:, :, s, k, q)/(q-1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del PEnC y su cuasicovarianza para k+1 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for j=1:q
    f(s, k, j)=1/(EFil(1, s, k, j)^2+3);
    Phi(s, k, j)=2*f(s, k, j)*ev(s, k, j)+ev(s, k, j)^2;
    EW(:, s, k, j)=[ev(s, k, j); Phi(s, k, j)];
    EPred(:, s, k+1, j)= [f(s, k, j); f(s, k, j)^2]+EW(:, s, k, j);
end

```

```

    pred1(s,k+1)=mean(EPred(1,s,k+1,:));
    pred2(s,k+1)=mean(EPred(2,s,k+1,:));
    Pred(:,s,k+1)=[pred1(s,k+1);pred2(s,k+1)];

    %%Cálculo de la matriz de covarianzas del predictor en k+1%%

    restaPred(:,s,k+1,1)=EPred(:,s,k+1,1)-Pred(:,s,k+1);
    CPredsum(:,s,k+1,1)=restaPred(:,s,k+1,1)*restaPred(:,s,k+1,1)';

    for j=2:q
        restaPred(:,s,k+1,j)=EPred(:,s,k+1,j)-Pred(:,s,k+1);
        CPredsum(:,s,k+1,j)=CPredsum(:,s,k+1,j-1)+...
            (restaPred(:,s,k+1,j)*restaPred(:,s,k+1,j)');
    end

    CPred(:,s,k+1)=CPredsum(:,s,k+1,q)/(q-1);

end %iter

end %simul

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Errores cuadráticos medios del FEn y FEnC %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

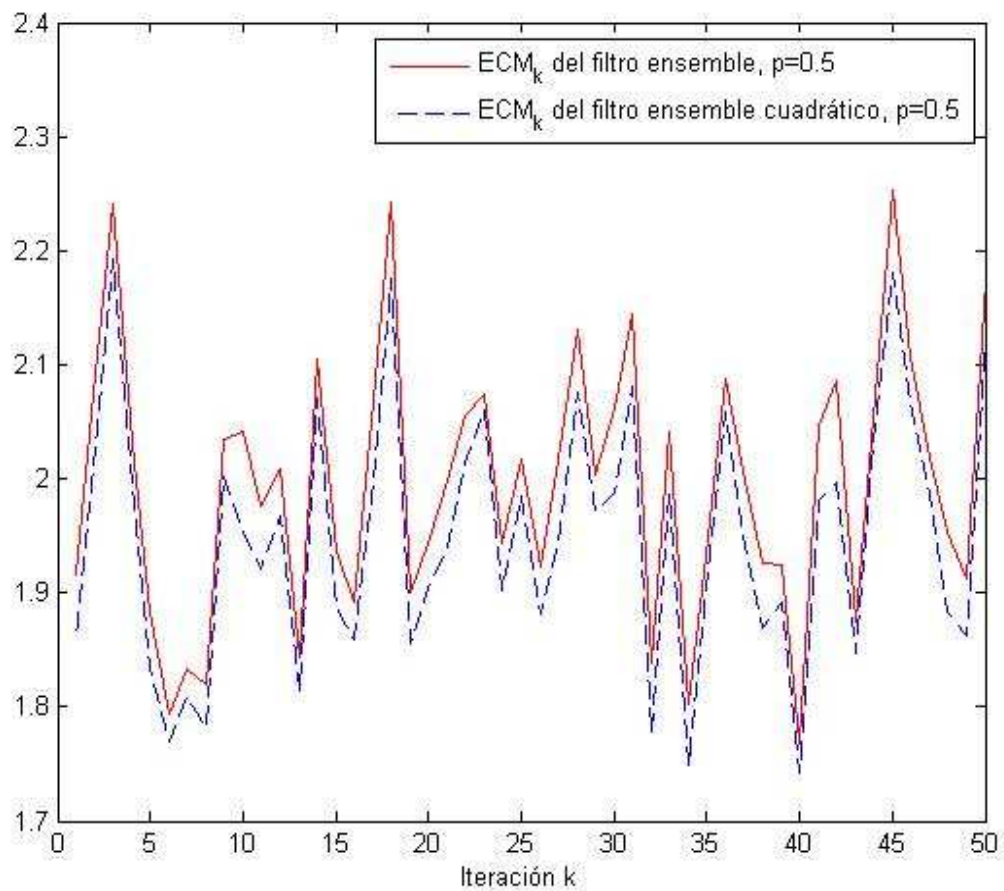
for k=1:n_iter
    SCE_fil(1,k)=DC_fil(1,k);
    SCE_Fil(1,k)=DC_Fil(1,k);
    for s=2:n_simul
        SCE_fil(s,k)=SCE_fil(s-1,k)+DC_fil(s,k);
        SCE_Fil(s,k)=SCE_Fil(s-1,k)+DC_Fil(s,k);
    end
    ECM_fil(k)=sqrt(SCE_fil(n_simul,k)/n_simul);
    ECM_Fil(k)=sqrt(SCE_Fil(n_simul,k)/n_simul)
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Gráfica de los errores cuadráticos medios %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:n_iter
    time(k)=k;
    pECM_fil(k)=ECM_fil(k);
    pECM_Fil(k)=ECM_Fil(k);
end

```

```
graf(3.5)=plot(time,pECM_fil,'r-',time,pECM_Fil,'b--');  
xlabel('Iteración k')  
legend('ECM_k del filtro ensemble, p=0.5',...  
      'ECM_k del filtro ensemble cuadrático, p=0.5')  
saveas(gcf, 'Fig3_5', 'fig')
```





### CAPÍTULO 3: Figura 3.7

```

%Modelo de la Señal:      x(k+1)=1/(x(k)^2+3)+w(k)   donde  x0-->N(0,1)
%                          f(x)=1/(x^2+3)

%Observaciones:         y(k)= gamma(k)*(x(k)^2+exp{x(k)})+v(k)  donde
%                          h(x)=x^2+exp{x}
%                          gamma(k)-->B(p)

%   P[w(k)=1]=15/18      P[w(k)=-3]=2/18      P[w(k)=-9]=1/18
%   P[v(k)=1]=15/18      P[v(k)=-3]=2/18      P[v(k)=-9]=1/18

n_simul=1000; %número de simulaciones
n_iter=50;    %número de iteraciones

for s=1:n_simul

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%% Generación de la señal x(k) %%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    P0=1;          %varianza de la señal inicial
    P3=0;          %momento centrado de tercer orden de la señal inicial
    P4=3*P0^2-P0^2; %momento centrado de cuarto orden de la señal inicial

    x0(s)=P0^(1/2)*randn(1,1);

    Q=19/3;        %varianza del ruido de la ecuación de la señal w(k)
    Q3=-128/3;     %momento centrado de tercer orden del ruido w(k)
    Q4=1123/3-Q^2; %momento centrado de cuarto orden del ruido w(k)

    a=rand(1,1);
    if a <= 1/18
        w0(s)=-9;
    elseif a <= 3/18
        w0(s)=-3;
    else w0(s)=1;
    end

    x(s,1)=1/(x0(s)^2+3)+w0(s);

```

```

for k=1:n_iter

    b=rand(1,1);
    if b <= 1/18
        w(s,k)=-9;
    elseif b <= 3/18
        w(s,k)=-3;
    else w(s,k)=1;
    end
    x(s,k+1)=1/(x(s,k)^2+3)+w(s,k);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Generación de las observaciones y(k) %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

p=0.5; %parámetro de la distribución Bernoulli

R=19/3;          %varianza del ruido de la ecuación observación v(k)
R3=-128/3;      %momento centrado de tercer orden del ruido v(k)
R4=1123/3-R^2;  %momento centrado de cuarto orden del ruido v(k)

for k=1:n_iter

    gamma(s,k)=binornd(1,p);
    c=rand(1,1);
    if c <= 1/18
        v(s,k)=-9;
    elseif c <= 3/18
        v(s,k)=-3;
    else v(s,k)=1;
    end
    y(s,k)=gamma(s,k)*(x(s,k)^2+exp(x(s,k)))+v(s,k);

end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO EXTENDIDO CUADRÁTICO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Fil=filtro del sistema aumentado
%Pred=predicador del sistema aumentado
%CFil=matriz de covarianzas de Fil
%CPred=matriz de covarianzas de Pred
%filtro=filtro extendido cuadrático (FEC)
%predicador=predicador extendido cuadrático (PEC)
%FQ=derivada de la función f en el FEC
%FF= matriz de la ecuación de la señal aumentada
%HQ=derivada de la función h en el PEC
%HH=matriz de la ecuación observación aumentada
%d=esperanza de la señal linealizada
%D=varianza de la señal linealizada
%vv=ruido del sistema linealizado
%r=varianza de vv
%r3=momento centrado de tercer orden de vv
%r4=momento centrado de cuarto orden de vv
%QQ=matriz de covarianzas del ruido de la ecuación de la señal
%  aumentada
%Q_ij=componente ij de la matriz QQ
%VV=ruido del sistema aumentado
%V=predicador de VV
%RR=matriz de covarianzas de VV
%R_ij=componente ij de la matriz RR
%innov=innovación
%Pi=matriz de covarianzas de la innovación
%KK=ganancia del sistema aumentado
%G=matriz de ganancia del predictor de VV
%Px=matriz de covarianzas de la señal aumentada
%Pv=matriz de covarianzas de V
%Pvx=matriz de covarianzas de los errores de predicción de
%  la señal y el ruido
%DC_filtro=cuadrado de la diferencia entre la señal y el FEC

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%% Condiciones iniciales %%%%%%

%%% k=0

%filtro

Fil0(:,s)=[0;0];
CFil0(:,s)=[P0 P30;P30' P40];

```

```

d0(s)=0;%esperanza de x_0
filtro0(s)=Fil0(1,s)+d0(s);

%predictor de la señal

FQ0(s)=(-2)*filtro0(s)/(filtro0(s)^2+3)^2;
FF0(:,s)=[FQ0(s) 0;0 FQ0(s)^2];
Pred(:,s,1)=FF0(:,s)*Fil0(:,s);

D0(s)=P0; %E[(x0-E[x0]) (x0-E[x0])^T]
Q0_22=4*FQ0(s)^2*D0(s)*Q+Q4;
QQ0(:,s)=[Q Q3;Q3' Q0_22];
CPred(:,s,1)=FF0(:,s)*CFil0(:,s)*FF0(:,s)'+QQ0(:,s);

u0(s)=1/(filtro0(s)^2+3)-FQ0(s)*filtro0(s);
d(s,1)=FQ0(s)*d0(s)+u0(s); %E[x(1)]
predictor(s,1)= Pred(1,s,1)+d(s,1);

%%% k=1

%observación aumentadas

D(s,1)=FQ0(s)^2*D0(s)+Q; %E[(x(1)-E[x(1)])(x(1)-E[x(1)])']
X(:,s,1)=[x(s,1); x(s,1)^2];
EX(:,s,1)=[d(s,1);D(s,1)];
XX(:,s,1)=X(:,s,1)-EX(:,s,1);

HQ(s,1)=2*predictor(s,1)+exp(predictor(s,1));
HH(:,s,1)=[HQ(s,1) 0;0 HQ(s,1)^2];

z(s,1)=predictor(s,1)^2+exp(predictor(s,1))-HQ(s,1)*predictor(s,1);
c(s,1)=HQ(s,1)*d(s,1)+z(s,1);
vv(s,1)=(gamma(s,1)-p)*c(s,1)+v(s,1);
psi(s,1)=2*HQ(s,1)*XX(1,s,1)*gamma(s,1)*vv(s,1)+vv(s,1)^2;

r(s,1)=p*(1-p)*c(s,1)^2+R;
r3(s,1)=p*(1-p)*(1-2*p)*c(s,1)^3+R3;
r4(s,1)=(p*(1-p)*(3*p^2-3*p+1)-p^2*(1-p)^2)*c(s,1)^4+...
        R4+4*p*(1-p)*c(s,1)^2*R;

VV(:,s,1)=[vv(s,1); psi(s,1)-r(s,1)]+...
        (gamma(s,1)-p)*HH(:,s,1)*EX(:,s,1);

Y(:,s,1)=gamma(s,1)*HH(:,s,1)*XX(:,s,1)+VV(:,s,1);

```

```

%innovación, filtro

Innov(:,s,1)=Y(:,s,1)-p*HH(:,s,1)*Pred(:,s,1);

Px(:,s,1)=FF0(:,s,1)*CFil0(:,s,1)*FF0(:,s,1)'+QQ0(:,s,1);

R_11(s,1,1)=r(s,1);
R_12(s,1,1)=r3(s,1)+p*(1-p)*c(s,1)*D(s,1)*HQ(s,1)^2;
R_22(s,1,1)=4*HQ(s,1)^2*D(s,1)*(p*(1-p)^2*c(s,1)^2+p*R)+...
            r4(s,1)+p*(1-p)*(1-2*p)*(c(s,1)^2*D(s,1)*HQ(s,1)^2+...
            HQ(s,1)^2*D(s,1)*c(s,1)^2)+p*(1-p)*HQ(s,1)^4*D(s,1)^2;
RR(:,s,1,1)=[R_11(s,1,1) R_12(s,1,1);R_12(s,1,1)' R_22(s,1,1)];

Pv0(:,s,1,1)=RR(:,s,1,1);

Pi(:,s,1)=p*(1-p)*HH(:,s,1)*Px(:,s,1)*HH(:,s,1)'+...
          p^2*HH(:,s,1)*CPred(:,s,1)*HH(:,s,1)'+...
          Pv0(:,s,1,1);

KK(:,s,1)=p*CPred(:,s,1)*HH(:,s,1)'^*Pi(:,s,1)^(-1);

Fil(:,s,1)=Pred(:,s,1)+KK(:,s,1)*Innov(:,s,1);
CFil(:,s,1)=CPred(:,s,1)-KK(:,s,1)*Pi(:,s,1)*KK(:,s,1)';
filtro(s,1)=Fil(1,s,1)+d(s,1);
DC_filtro(s,1)=(x(s,1)-filtro(s,1))^2;

%predicador de la señal

FQ(s,1)=(-2)*filtro(s,1)/(filtro(s,1)^2+3)^2;
FF(:,s,1)=[FQ(s,1) 0;0 FQ(s,1)^2];
Pred(:,s,2)=FF(:,s,1)*Fil(:,s,1);

Q_22(s,1)=4*FQ(s,1)^2*D(s,1)*Q+Q4;
QQ(:,s,1)=[Q Q3;Q3' Q_22(s,1)];
CPred(:,s,2)=FF(:,s,1)*CFil(:,s,1)*FF(:,s,1)'+QQ(:,s,1);

u(s,1)=1/(filtro(s,1)^2+3)-FQ(s,1)*filtro(s,1);
d(s,2)=FQ(s,1)*d(s,1)+u(s,1);
predicador(s,2)= Pred(1,s,2)+d(s,2);

```

```

%%% k=2

%observación aumentadas

D(s,2)=FQ(s,1)^2*D(s,1)+Q; %E[(x(2)-E[x(2)])(x(2)-E[x(2)])']
X(:,s,2)=[x(s,2); x(s,2)^2];
EX(:,s,2)=[d(s,2);D(s,2)];

XX(:,s,2)=X(:,s,2)-EX(:,s,2);

HQ(s,2)=2*predicador(s,2)+exp(predicador(s,2));
HH(:,s,2)=[HQ(s,2) 0;0 HQ(s,2)^2];

z(s,2)=predicador(s,2)^2+exp(predicador(s,2))-HQ(s,2)*predicador(s,2);
c(s,2)=HQ(s,2)*d(s,2)+z(s,2);
vv(s,2)=(gamma(s,2)-p)*c(s,2)+v(s,2);
psi(s,2)=2*HQ(s,2)*XX(1,s,2)*gamma(s,2)*vv(s,2)+vv(s,2)^2;

r(s,2)=p*(1-p)*c(s,2)^2+R;
r3(s,2)=p*(1-p)*(1-2*p)*c(s,2)^3+R3;
r4(s,2)=(p*(1-p)*(3*p^2-3*p+1)-p^2*(1-p)^2)*c(s,2)^4+...
        R4+4*p*(1-p)*c(s,2)^2*R;

VV(:,s,2)=[vv(s,2); psi(s,2)-r(s,2)]+...
        (gamma(s,2)-p)*HH(:,s,2)*EX(:,s,2);

Y(:,s,2)=gamma(s,2)*HH(:,s,2)*XX(:,s,2)+VV(:,s,2);

%predicador del ruido (algoritmo en j)

%condiciones iniciales j=0

V0(:,s,2)=[0;0];
Pvx0(:,s,2,1)=[0 0;0 0];

for i=1:2
    if i==2
        R_11(s,2,i)=p*(1-p)*c(s,2)^2+R;
        R_12(s,2,i)=r3(s,2)+p*(1-p)*c(s,2)*D(s,2)*HQ(s,2)^2;
        R_22(s,2,i)=4*HQ(s,2)^2*D(s,2)*(p*(1-p)^2*c(s,2)^2+p*R)+...
            r4(s,2)+p*(1-p)*(1-2*p)*(c(s,2)^2*D(s,2)*HQ(s,2)^2+...
            HQ(s,2)^2*D(s,2)*c(s,2)^2)+p*(1-p)*HQ(s,2)^4*D(s,2)^2;
    else
        R_11(s,2,i)=0;
        R_12(s,2,i)=0;
        R_22(s,2,i)=4*HQ(s,2)^2*D(s,2)*p^2*(1-p)^2*c(s,2)^2;
    end
end

```

```

        RR(:,:,s,2,i)=[R_11(s,2,i) R_12(s,2,i);R_12(s,2,i)' R_22(s,2,i)];
        Pv0(:,:,s,2,i)=RR(:,:,s,2,i);
    end %i

```

---

```

G(:,:,s,2,1)=(p*Pvx0(:,:,s,2,1)*HH(:,:,s,1)'+...
        Pv0(:,:,s,2,1))*Pi(:,:,s,1)^(-1);

```

---

```

V(:,s,2,1)=V0(:,s,2)+G(:,:,s,2,1)*Innov(:,s,1);

```

---

**%innovación, filtro**

```

Innov(:,s,2)=Y(:,s,2)-p*HH(:,:,s,2)*Pred(:,s,2)-V(:,s,2,1);

Px(:,:,s,2)=FF(:,:,s,1)*Px(:,:,s,1)*FF(:,:,s,1)'+QQ(:,:,s,1);

Pv(:,:,s,2,2,1)=Pv0(:,:,s,2,2,1)-G(:,:,s,2,1)*Pi(:,:,s,1)*...
        G(:,:,s,2,1)';

Pvx(:,:,s,2,2,1)=(Pvx0(:,:,s,2,1)-...
        G(:,:,s,2,1)*Pi(:,:,s,1)*KK(:,:,s,1)')*FF(:,:,s,1)';

Pi(:,:,s,2)=p*(1-p)*HH(:,:,s,2)*Px(:,:,s,2)*HH(:,:,s,2)'+...
        p^2*HH(:,:,s,2)*CPred(:,:,s,2)*HH(:,:,s,2)'+...
        Pv(:,:,s,2,2,1)+p*HH(:,:,s,2)*Pvx(:,:,s,2,2,1)'+...
        p*Pvx(:,:,s,2,2,1)*HH(:,:,s,2)';

KK(:,:,s,2)=(p*CPred(:,:,s,2)*HH(:,:,s,2)'+...
        Pvx(:,:,s,2,2,1)')*Pi(:,:,s,2)^(-1);

Fil(:,s,2)=Pred(:,s,2)+KK(:,:,s,2)*Innov(:,s,2);
CFil(:,:,s,2)=CPred(:,:,s,2)-KK(:,:,s,2)*Pi(:,:,s,2)*KK(:,:,s,2)';
filtro(s,2)=Fil(1,s,2)+d(s,2);
DC_filtro(s,2)=(x(s,2)-filtro(s,2))^2;

```

**%predicador de la señal**

```

FQ(s,2)=(-2)*filtro(s,2)/(filtro(s,2)^2+3)^2;
FF(:,:,s,2)=[FQ(s,2) 0;0 FQ(s,2)^2];
Pred(:,s,3)=FF(:,:,s,2)*Fil(:,s,2);

```

```

Q_22(s,2)=4*FQ(s,2)^2*D(s,2)*Q+Q4;
QQ(:,:,s,2)=[Q Q3;Q3' Q_22(s,2)];
CPred(:,:,s,3)=FF(:,:,s,2)*CFil(:,:,s,2)*FF(:,:,s,2)'+QQ(:,:,s,2);

u(s,2)=1/(filtro(s,2)^2+3)-FQ(s,2)*filtro(s,2);
d(s,3)=FQ(s,2)*d(s,2)+u(s,2);
predictor(s,3)= Pred(1,s,3)+d(s,3);

for k=3:n_iter

%observación aumentadas

D(s,k)=FQ(s,k-1)^2*D(s,k-1)+Q; %E[(x(k)-E[x(k)])(x(k)-E[x(k)])']
X(:,s,k)=[x(s,k); x(s,k)^2];
EX(:,s,k)=[d(s,k);D(s,k)];
XX(:,s,k)=X(:,s,k)-EX(:,s,k);

HQ(s,k)=2*predictor(s,k)+exp(predictor(s,k));
HH(:,:,s,k)=[HQ(s,k) 0;0 HQ(s,k)^2];

z(s,k)=predictor(s,k)^2+exp(predictor(s,k))-HQ(s,k)*predictor(s,k);
c(s,k)=HQ(s,k)*d(s,k)+z(s,k);
vv(s,k)=(gamma(s,k)-p)*c(s,k)+v(s,k);
psi(s,k)=2*HQ(s,k)*XX(1,s,k)*gamma(s,k)*vv(s,k)+vv(s,k)^2;

r(s,k)=p*(1-p)*c(s,k)^2+R;
r3(s,k)=p*(1-p)*(1-2*p)*c(s,k)^3+R3;
r4(s,k)=(p*(1-p)*(3*p^2-3*p+1)-p^2*(1-p)^2)*c(s,k)^4+...
R4+4*p*(1-p)*c(s,k)^2*R;

VV(:,s,k)=[vv(s,k); psi(s,k)-r(s,k)]+...
(gamma(s,k)-p)*HH(:,:,s,k)*EX(:,s,k);

Y(:,s,k)=gamma(s,k)*HH(:,:,s,k)*XX(:,s,k)+VV(:,s,k);

%predictor del ruido (algoritmo en j)

%condiciones iniciales j=0

V0(:,s,k)=[0;0];
Pvx0(:,:,s,k,1)=[0 0;0 0];

```



```

for i=1:k
    if i==k
        R_11(s,k,i)=p*(1-p)*c(s,k)^2+R;
        R_12(s,k,i)=r3(s,k)+p*(1-p)*c(s,k)*D(s,k)*HQ(s,k)^2;
        R_22(s,k,i)=4*HQ(s,k)^2*D(s,k)*(p*(1-p)^2*c(s,k)^2+p*R)+...
            r4(s,k)+p*(1-p)*(1-2*p)*(c(s,k)^2*D(s,k)*HQ(s,k)^2+...
            HQ(s,k)^2*D(s,k)*c(s,k)^2)+p*(1-p)*HQ(s,k)^4*D(s,k)^2;
    else
        R_11(s,k,i)=0;
        R_12(s,k,i)=0;
        R_22(s,k,i)=4*HQ(s,k)^2*D(s,k)*p^2*(1-p)^2*c(s,k)^2;
    end

    RR(:,:,s,k,i)=[R_11(s,k,i) R_12(s,k,i);R_12(s,k,i)' R_22(s,k,i)];
    Pv0(:,:,s,k,i)=RR(:,:,s,k,i);
end %i

%
G(:,:,s,k,1)=(p*Pvx0(:,:,s,k,1)*HH(:,:,s,1)'+...
    Pv0(:,:,s,k,1))*Pi(:,:,s,1)^(-1);

%
Pv(:,:,s,k,2,1)=Pv0(:,:,s,k,2)-...
    G(:,:,s,k,1)*Pi(:,:,s,1)*G(:,:,s,2,1)';
Pv(:,:,s,k,k,1)=Pv0(:,:,s,k,k)-...
    G(:,:,s,k,1)*Pi(:,:,s,1)*G(:,:,s,k,1)';
Pvx(:,:,s,k,2,1)=(Pvx0(:,:,s,k,1)-...
    G(:,:,s,k,1)*Pi(:,:,s,1)*KK(:,:,s,1)')*FF(:,:,s,1)';

for j=3:k-1
    Pv(:,:,s,k,j,1)=Pv0(:,:,s,k,j)-...
        G(:,:,s,k,1)*Pi(:,:,s,1)*G(:,:,s,j,1)';
    for l=2:j-1
        Pv(:,:,s,k,j,l)=Pv(:,:,s,k,j,l-1)-...
            G(:,:,s,k,1)*Pi(:,:,s,1)*G(:,:,s,j,l)';
    end %l
    Pvx(:,:,s,k,j,j-1)=(Pvx(:,:,s,k,j-1,j-2)-...
        G(:,:,s,k,j-1)*Pi(:,:,s,j-1)*KK(:,:,s,j-1)')*FF(:,:,s,j-1)';
end %j

%
for j=2:k-1
    G(:,:,s,k,j)=(p*Pvx(:,:,s,k,j,j-1)*HH(:,:,s,j)'+...
        Pv(:,:,s,k,j,j-1))*Pi(:,:,s,j)^(-1);
end %j

%

```

```

V(:,s,k,1)=V0(:,s,k)+G(:, :,s,k,1)*Innov(:,s,1);

for j=2:k-1
    V(:,s,k,j)=V(:,s,k,j-1)+G(:, :,s,k,j)*Innov(:,s,j);
end
%
%innovación, filtro

Innov(:,s,k)=Y(:,s,k)-p*HH(:, :,s,k)*Pred(:,s,k)-V(:,s,k,k-1);

Px(:, :,s,k)=FF(:, :,s,k-1)*Px(:, :,s,k-1)*FF(:, :,s,k-1)'+QQ(:, :,s,k-1);

for j=2:k-1
    Pv(:, :,s,k,k,j)=Pv(:, :,s,k,k,j-1)-...
        G(:, :,s,k,j)*Pi(:, :,s,j)*G(:, :,s,k,j)';
end

Pvx(:, :,s,k,k,k-1)=(Pvx(:, :,s,k,k-1,k-2)-...
    G(:, :,s,k,k-1)*Pi(:, :,s,k-1)*KK(:, :,s,k-1)')*FF(:, :,s,k-1)';

Pi(:, :,s,k)=p*(1-p)*HH(:, :,s,k)*Px(:, :,s,k)*HH(:, :,s,k)'+...
    p^2*HH(:, :,s,k)*CPred(:, :,s,k)*HH(:, :,s,k)'+...
    Pv(:, :,s,k,k,k-1)+p*HH(:, :,s,k)*Pvx(:, :,s,k,k,k-1)'+...
    p*Pvx(:, :,s,k,k,k-1)*HH(:, :,s,k)';

KK(:, :,s,k)=(p*CPred(:, :,s,k)*HH(:, :,s,k)'+...
    Pvx(:, :,s,k,k,k-1)')*Pi(:, :,s,k)^(-1);
Fil(:,s,k)=Pred(:,s,k)+KK(:, :,s,k)*Innov(:,s,k);
CFil(:, :,s,k)=CPred(:, :,s,k)-KK(:, :,s,k)*Pi(:, :,s,k)*KK(:, :,s,k)';
filtro(s,k)=Fil(1,s,k)+d(s,k);
DC_filtro(s,k)=(x(s,k)-filtro(s,k))^2;

%predicador de la señal

FQ(s,k)=(-2)*filtro(s,k)/(filtro(s,k)^2+3)^2;
FF(:, :,s,k)=[FQ(s,k) 0;0 FQ(s,k)^2];
Pred(:,s,k+1)=FF(:, :,s,k)*Fil(:,s,k);

Q_22(s,k)=4*FQ(s,k)^2*D(s,k)*Q+Q4;
QQ(:, :,s,k)=[Q Q3;Q3' Q_22(s,k)];
CPred(:, :,s,k+1)=FF(:, :,s,k)*CFil(:, :,s,k)*FF(:, :,s,k)'+QQ(:, :,s,k);

u(s,k)=1/(filtro(s,k)^2+3)-FQ(s,k)*filtro(s,k);
d(s,k+1)=FQ(s,k)*d(s,k)+u(s,k);
predicador(s,k+1)= Pred(1,s,k+1)+d(s,k+1);

end %iter

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FILTRO ENSEMBLE CUADRÁTICO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %fil=filtro ensemble cuadrático (FEnC)
    %pred=predictor ensemble cuadrático (PEnc)
    %Cfil=cuasivarianza del FEnC
    %Cpred=cuasivarianza del PEnc
    %KK=ganancia del FEnC
    %DC_fil=cuadrado de la diferencia entre la señal y el FEnC

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Generación del ensemble de filtrado y predicción %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Efil=conjunto de filtros
    %EW=conjunto de valores simulados a partir de la distribución del
    % ruido aditivo de la ecuación de la señal aumentada
    %Phi=componente (1,2) de EW
    %EV=conjunto de valores simulados a partir de la distribución del
    % ruido aditivo de la ecuación de observación aumentada
    %Psi=componente (1,2) de EV
    %Epred=conjunto de predictores
    %Obs=conjunto de predictores de las observaciones
    %Yest=media de Obs
    %CYY=cuasicovarianza de Obs
    %CXY=cuasicovarianza del PEnc y Obs

q=500; %tamaño del conjunto de valores simulados

%% Simulación del ensemble de filtrado y predicción inicial %%

for j=1:q
    efil0(s,j)=P0^(1/2)*randn(1,1);

    d=rand(1,1);
    if d <= 1/18
        ew0(s,j)=-9;
    elseif d <= 3/18
        ew0(s,j)=-3;
    else ew0(s,j)=1;
    end

    Efil0(:,s,j)=[efil0(s,j); efil0(s,j)^2];
    f0(s,j)=1/(efil0(s,j)^2+3);
    Phi0(s,j)=2*f0(s,j)*ew0(s,j)+ew0(s,j)^2;
    EW0(:,s,j)=[ew0(s,j); Phi0(s,j)];

    Epred(:,s,1,j)=[f0(s,j); f0(s,j)^2]+EW0(:,s,j);
end

```

```

%% Cálculo del predictor para k=1 %%

pred1(s,1)=mean(Epred(1,s,1,:));
pred2(s,1)=mean(Epred(2,s,1,:));
pred(:,s,1)=[pred1(s,1);pred2(s,1)];

%% Cálculo de la matriz de cuasicovarianzas del predictor para k=1 %%

restapred(:,s,1,1)=Epred(:,s,1,1)-pred(:,s,1);

Cpredsum(:, :, s,1,1)=restapred(:,s,1,1)*restapred(:,s,1,1)';

for j=2:q
    restapred(:,s,1,j)=Epred(:,s,1,j)-pred(:,s,1);
    Cpredsum(:, :, s,1,j)=Cpredsum(:, :, s,1,j-1)+...
        (restapred(:,s,1,j)*restapred(:,s,1,j)');
end

Cpred(:, :, s,1)=Cpredsum(:, :, s,1,q)/(q-1);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del FEnC y su cuasicovarianza para k %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

for k=1:n_iter

    %% Cálculo de las observaciones estimadas en k %%

    for j=1:q
        h(s,k,j)=Epred(1,s,k,j)^2+exp(Epred(1,s,k,j));
        Psi(s,k,j)=2*egamma(s,k,j)*h(s,k,j)*ev(s,k,j)+ev(s,k,j)^2;
        EV(:,s,k,j)=[ev(s,k,j); Psi(s,k,j)];
        Obs(:,s,k,j)= egamma(s,k,j)*[h(s,k,j); h(s,k,j)^2]+...
            EV(:,s,k,j);
    end

    yest1(s,k)=mean(Obs(1,s,k,:));
    yest2(s,k)=mean(Obs(2,s,k,:));
    Yest(:,s,k)=[yest1(s,k); yest2(s,k)];

```

```

%% Cálculo de la matriz de cuasicovarianzas del predictor de %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% la observación en k %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

restaYY(:,s,k,1)=Obs(:,s,k,1)-Yest(:,s,k);
CYYsum(:, :, s,k,1)=restaYY(:,s,k,1)*restaYY(:,s,k,1)';

for j=2:q
    restaYY(:,s,k,j)=Obs(:,s,k,j)-Yest(:,s,k);
    CYYsum(:, :, s,k,j)=CYYsum(:, :, s,k,j-1)+...
        (restaYY(:,s,k,j)*restaYY(:,s,k,j)');
end

CYY(:, :, s,k)=CYYsum(:, :, s,k,q)/(q-1);

%% Cálculo de la matriz de cuasicovarianzas cruzada del PEnC %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% y Obs en k %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

CXYSum(:, :, s,k,1)=restapred(:,s,k,1)*restaYY(:,s,k,1)';

for j=2:q
    CXYSum(:, :, s,k,j)=CXYSum(:, :, s,k,j-1)+...
        (restapred(:,s,k,j)*restaYY(:,s,k,j)');
end

CXY(:, :, s,k)=CXYSum(:, :, s,k,q)/(q-1);

Y(:,s,k)=[y(s,k);y(s,k)^2];

%% Cálculo del FEnC y su matriz de cuasicovarianzas del error %%

KK(:, :, s,k)=CXY(:, :, s,k)*CYY(:, :, s,k)^(-1);

for j=1:q
    Efil(:,s,k,j)=Epred(:,s,k,j)+KK(:, :, s,k)*...
        (Y(:,s,k)-Obs(:,s,k,j));
end

fill(s,k)=mean(Efil(1,s,k,:));
fil2(s,k)=mean(Efil(2,s,k,:));
fil(:,s,k)=[fill(s,k);fil2(s,k)];
DC_fil(s,k)=(x(s,k)-fill(s,k))^2;

restafil(:,s,k,1)=Efil(:,s,k,1)-fil(:,s,k);
Cfilsum(:, :, s,k,1)=restafil(:,s,k,1)*restafil(:,s,k,1)';

```

```

for j=2:q
    restafil(:,s,k,j)=Efil(:,s,k,j)-fil(:,s,k);
    Cfilsum(:,:,s,k,j)=Cfilsum(:,:,s,k,j-1)+...
        (restafil(:,s,k,j)*restafil(:,s,k,j)');
end

Cfil(:,:,s,k)=Cfilsum(:,:,s,k,q)/(q-1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cálculo del PEnC y su cuasivarianza para k+1 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for j=1:q
    f(s,k,j)=1/(Efil(1,s,k,j)^2+3);
    Phi(s,k,j)=2*f(s,k,j)*ev(s,k,j)+ev(s,k,j)^2;
    EW(:,s,k,j)=[ev(s,k,j); Phi(s,k,j)];
    Epred(:,s,k+1,j)= [f(s,k,j);f(s,k,j)^2]+EW(:,s,k,j);
end

pred1(s,k+1)=mean(Epred(1,s,k+1,:));
pred2(s,k+1)=mean(Epred(2,s,k+1,:));
pred(:,s,k+1)=[pred1(s,k+1);pred2(s,k+1)];

%%Cálculo de la matriz de cuasivarianzas del predictor en k+1%%

restapred(:,s,k+1,1)=Epred(:,s,k+1,1)-pred(:,s,k+1);
Cpredsum(:,:,s,k+1,1)=restapred(:,s,k+1,1)*restapred(:,s,k+1,1)';

for j=2:q
    restapred(:,s,k+1,j)=Epred(:,s,k+1,j)-pred(:,s,k+1);
    Cpredsum(:,:,s,k+1,j)=Cpredsum(:,:,s,k+1,j-1)+...
        (restapred(:,s,k+1,j)*restapred(:,s,k+1,j)');
end

Cpred(:,:,s,k+1)=Cpredsum(:,:,s,k+1,q)/(q-1);

end %iter

end %simul

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Errores cuadráticos medios del FEC y FEnC %%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
for k=1:n_iter  
    SCE_filtro(1,k)=DC_filtro(1,k);  
    SCE_fil(1,k)=DC_fil(1,k);  
    for s=2:n_simul  
        SCE_filtro(s,k)=SCE_fil(s-1,k)+DC_filtro(s,k);  
        SCE_fil(s,k)=SCE_fil(s-1,k)+DC_fil(s,k);  
    end  
    ECM_filtro(k)=sqrt(SCE_filtro(n_simul,k)/n_simul);  
    ECM_fil(k)=sqrt(SCE_fil(n_simul,k)/n_simul);  
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Gráfica de los errores cuadráticos medios %%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
for k=1:n_iter  
    time(k)=k;  
    pECM_filtro(k)=ECM_filtro(k);  
    pECM_fil(k)=ECM_fil(k);  
end
```

```
graf(3.7)=plot(time,pECM_filtro,'b--',time,pECM_fil,'r-');  
xlabel('Iteración k')  
legend('ECMk del filtro extendido cuadrático, p=0.5',...  
      'ECMk del filtro ensemble Cuadrático, p=0.5')  
saveas(gcf, 'Fig3_7', 'fig')
```

