

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS  
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA



***Trabajo Fin de Grado***

**Estimación de parámetros en redes  
inalámbricas de detección distribuida  
(Parameter Estimation in Distributed  
Detection Wireless Networks)**

Para acceder al Título de

***Graduado en  
Ingeniería de Tecnologías de Telecomunicación***

Autor: Blanca Ruiz Cubero

Julio -2024

# GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

## CALIFICACIÓN DEL TRABAJO FIN DE GRADO

**Realizado por:** Blanca Ruiz Cubero

**Director del TFG:** Jesus Perez Arriaga

**Título:** “Estimación de parámetros en redes inalámbricas de detección distribuida”

**Title:** “Parameter Estimation in Distributed Detection Wireless Networks”

**Presentado a examen el día:**

para acceder al Título de

## GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

### Composición del Tribunal:

Presidente (Apellidos, Nombre): Ibañez Diaz, Jesus Maria

Secretario (Apellidos, Nombre): Laso Perez, Alberto

Vocal (Apellidos, Nombre): Irastorza Teja, Jose Angel

Este Tribunal ha resuelto otorgar la calificación de: .....

**Fdo: El Presidente**

**Fdo: El Secretario**

**Fdo: El Vocal**

**Fdo: El Director del TFG**  
(sólo si es distinto del Secretario)

Vº Bº del Subdirector

Trabajo Fin de Grado Nº  
(a asignar por Secretaría)



---

### *Agradecimientos*

---

Quiero expresar mi más sincero agradecimiento a Jesús Pérez Arriaga por su invaluable orientación y apoyo durante la realización de mi Trabajo de Fin de Grado. Su guía experta y su dedicación han sido fundamentales para el éxito de ese proyecto académico.

A lo largo de este proceso, Jesús ha compartido generosamente su conocimiento y experiencia, brindándome la orientación necesaria para navegar por los desafíos del TFG.

También quiero expresar mi agradecimiento a todos los profesores que me han acompañado a lo largo de mi carrera universitaria. Sus conocimientos, enseñanzas y consejos han sido fundamentales para mi formación académica y personal.

Por último, quiero agradecer a mis compañeros de carrera por su apoyo y compañerismo durante estos años.

## Contenido

<b>1. Introducción.....</b>	<b>7</b>
1.1. Abstract .....	7
1.2. Palabras clave .....	7
1.3. Contexto y propósitos.....	8
1.4. Descripción del problema.....	11
1.5. Estructura de la memoria.....	12
<b>2. Fundamentos Teóricos .....</b>	<b>13</b>
2.1. Modelo del Problema.....	13
2.2. Estimación de parámetros: Algoritmo EM .....	14
2.2.1. Paso E .....	15
2.2.2. Paso M .....	15
2.3. Estimador supervisado .....	18
2.4. Detección: Regla de Fusión .....	19
2.5. Extensión: desconocimiento de las probabilidades a priori .....	21
2.5.1. Estimación de parámetros $d$ y $u$ .....	21
2.5.2. Estimador supervisado para $d$ y $u$ .....	22
2.5.3. Detección .....	22
<b>3. Implementación software .....</b>	<b>23</b>
<b>4. Resultados .....</b>	<b>25</b>
4.1. Prestaciones del estimador de $d$ y de la regla de decisión .....	25
4.2. Prestaciones del estimador de $d$ y $u$ , y de la regla de decisión .....	30
4.3. Comparaciones con la regla de voto Mayoritario.....	35
<b>5. Conclusiones .....</b>	<b>38</b>
<b>6. Apéndices.....</b>	<b>39</b>
6.1. Apéndice A: Resolución Paso E para la estimación de $d$ y $u$ .....	39
6.2. Apéndice B: Resolución Paso M para la estimación de $d$ y $u$ .....	40
<b>7. Referencias .....</b>	<b>41</b>

## INDICE DE FIGURAS

Figura 1: sistema DD-WSN .....	11
Figura 2: Funciones matlab .....	23
Figura 3: RMSE en función del número de periodos de senado .....	26
Figura 4: Probabilidad de error en función del número de periodos de sensado .....	26
Figura 5: RMSE en función de la probabilidad a priori.....	27
Figura 6: Probabilidad de error en función de la probabilidad a priori.....	27
Figura 7: RMSE en función del número de sensores .....	28
Figura 8: Probabilidad de error en función del número de sensores .....	28
Figura 9: RMSE en función de la probabilidad de error media de los canales BSC .....	29
Figura 10: Probabilidad de error en función de la probabilidad de error media de los canales BSC .....	29
Figura 11: RMSE en función del número de periodos de sensado .....	31
Figura 12: Probabilidad de error en función del número de periodos de sesnado .....	31
Figura 13: RMSE en función de la probabilidad a priori.....	32
Figura 14: Probabilidad de error en función de la probabilidad a priori.....	32
Figura 15: RMSE en función del número de sensores .....	33
Figura 16: Probabilidad de error en función del número de sensores.....	33
Figura 17: RMSE en función de la probabilidad de error media de los canales BSC .....	34
Figura 18: Probabilidad de error en función de la probabilidad de error media de los canales BSC .....	34
Figura 19: Probabilidad de error en función del número de periodos de sensado .....	35
Figura 20: Probabilidad e error en función de la probabilidad a priori.....	36
Figura 21: Probabilidad de error en función del número de sensores .....	36
Figura 22: Probabilidad de error en función del error medio del canal.....	37

# 1. Introducción

## 1.1. Abstract

This work presents an algorithm to blindly estimate the model parameters of decision fusion systems over wireless sensor networks. In particular, it considers the so-called canonical distributed detection systems, where the sensors report their decisions to the fusion center (FC), through independent binary symmetric channels. Then, the FC makes the final decision by combining the noisy sensor decisions according to a certain fusion rule. We consider fully heterogeneous networks where the sensors can have different probabilities of detection and false-alarm, and the reporting channels can have different crossover probabilities. When the FC knows all these model parameters the optimal fusion rule is the likelihood ratio (LR) test. But the likelihood ratio depends on the model parameters, which may be unknown (all or some of them) in many practical cases, making the LR test inapplicable. In this work, we present an algorithm for the FC to blindly learn the sensor probabilities of detection from the noisy sensor decisions received after a number of sensing periods. The algorithm can also estimate the prior probabilities of the null and alternative hypothesis when they are unknown by the FC. Then, based on the estimates of these model parameters, a channel-aware fusion rule is derived. Simulation results show that, after sufficient sensing periods, the model parameter estimates are accurate enough for the fusion rule to exhibit near-optimal detection performance.

## 1.2. Palabras clave

Redes de sensores inalámbricos, centro de fusión, algoritmo EM, mínima probabilidad de error, fusión de decisiones, test de cociente de verosimilitudes, umbral de decisión, detección distribuida.

### 1.3. Contexto y propósitos

Este trabajo aborda el problema de detección distribuida [1] en redes de sensores inalámbricos (DD-WSN: Distributed detection in Wireless sensor networks). Consiste en un conjunto de sensores distribuidos espacialmente y un centro de fusión (FC). Los sensores informan sus decisiones binarias, sobre la presencia o ausencia de un evento/objeto de interés, al FC a través de canales inalámbricos dedicados. El FC luego fusiona las decisiones ruidosas recibidas de los sensores para tomar la decisión final según una regla de fusión dada. Los sensores no se comunican entre sí y no hay retroalimentación del FC a los sensores. Se asume que la comunicación es solo entre cada sensor y el FC a través del canal de reporte correspondiente. Por lo tanto, la única información en el FC son las decisiones ruidosas de los sensores recibidos por los canales de reporte.

En general, el diseño de sistemas DD-WSN requiere determinar las reglas de decisión local en los sensores y la regla de fusión en el FC [1]. Variaciones de esta formulación incluyen la optimización solo de las reglas de decisión local para una regla de fusión dada [2]–[4], y la optimización de la regla de fusión para reglas de decisión de sensores dadas [5]–[9]. En este trabajo nos centramos en el último caso.

En las WSN homogéneas todos los sensores exhiben un rendimiento idéntico. En otras palabras, operan con las mismas probabilidades de falsa alarma y detección [10]–[12]. Pero en muchos casos, donde los sensores se colocan en diferentes ubicaciones, los sensores tienen diferentes probabilidades de detección, incluso si los sensores son tecnológicamente idénticos. Algunos autores consideran WSN semi-homogéneas donde los sensores tienen diferentes probabilidades de detección pero idénticas probabilidades de falsa alarma [13], [14]. En algunas aplicaciones, los sensores también operan con diferentes probabilidades de falsa alarma [7], [9], [15]. A esas se les llama redes completamente heterogéneas. En este trabajo consideramos un marco general que asume WSN completamente heterogéneas.

La fiabilidad de las decisiones de los sensores observadas por el FC no solo depende del rendimiento de detección de los sensores en sí, sino también de la fiabilidad de sus canales de reporte. Por lo tanto, el diseño de las reglas de fusión también requiere considerar el efecto de los canales de reporte. Esas reglas se llaman reglas de fusión conscientes de los canales [16].

En la literatura de WSN, los dos modelos de canales de reporte más comunes son el NFC (Noise Fading Channel) [7]–[9], [17] y el BSC (binary symmetric channel) [2], [12]–[14], [18]–[20]. En el primer caso, el FC recibe una versión desvanecida y ruidosa de la decisión del sensor. Por lo tanto, la salida de un NFC es un valor aleatorio complejo caracterizado por la respuesta del canal y la varianza del ruido. En el modelo BSC, las decisiones locales se codifican y transmiten al FC. Luego, el FC decodifica la señal recibida, por lo que la salida del BSC es un valor binario ruidoso. En un BSC, la fiabilidad del canal se caracteriza por su probabilidad de error. Esos FC a veces se llaman sistemas de decodificar y luego fusionar [19], [21].

En este trabajo consideramos el caso BSC, en el que los canales de reporte pueden tener diferentes probabilidades de error [12]. Como es común en la literatura técnica, asumimos que el FC conoce las probabilidades de error [13], [18], [22]. Esta es una suposición

razonable en escenarios de baja movilidad donde el FC puede estimar las tasas de error de los BSC basándose en símbolos de entrenamiento transmitidos por los sensores.

El diseño de la regla de fusión es conceptualmente sencillo cuando el FC conoce el rendimiento de los sensores y las probabilidades de error de los BSC. En este caso, la regla de fusión óptima se basa en el test de cociente de verosimilitudes (LRT: likelihood ratio test), que depende de las probabilidades de detección y falsa alarma de los sensores, y de las probabilidades de error de los canales [14]. Para tomar la decisión final, el FC compara el LR con un umbral de decisión, cuyo valor depende del criterio de detección empleado [23], [24]. Si se utiliza el criterio de riesgo mínimo de Bayes (que incluye la mínima probabilidad de error como caso particular), el umbral de decisión también depende de las probabilidades a priori [5]. Alternativamente, si se emplea el criterio de Neyman-Pearson (NP), el umbral de decisión depende de las probabilidades de falsa alarma de los sensores [6]. El hecho de que el test de LR requiera conocimiento de los parámetros del modelo es una gran desventaja porque algunos de ellos, o incluso todos, pueden ser desconocidos por el FC en casos prácticos. En esos casos, el problema de detección se convierte en un problema de prueba de hipótesis compuesta [23].

Se han propuesto dos enfoques para superar esta dificultad: 1) usar reglas de fusión subóptimas independientes de los parámetros del modelo desconocidos, y 2) estimar los parámetros del modelo desconocidos y diseñar una regla de fusión en consecuencia. Ejemplos del primer enfoque son:

- Regla de Sensores Ideales (IS): Aproxima el LR asumiendo sensores ideales (sus probabilidades de detección y falsa alarma igual a uno y cero, respectivamente) [13]. La principal ventaja de este enfoque es que la estadística de prueba resultante solo depende de las probabilidades de error de los canales. Puede aplicarse a WSN heterogéneas y no requiere conocer el rendimiento local de los sensores. La regla IS puede extenderse fácilmente al caso en que el FC también conoce las probabilidades de falsa alarma de los sensores. En este caso, la regla también considera el LR en el caso límite cuando la probabilidad de detección de los sensores es uno, pero usando las probabilidades de falsa alarma conocidas.

- Reglas de conteo: También se llaman reglas de votación [1], [14], [20]. Su principal ventaja es su simplicidad y el hecho de que son reglas completamente ciegas. Esto significa que el FC no necesita conocer ningún parámetro del modelo (fiabilidad de los sensores, tasas de error de los canales, probabilidades a priori, etc.). Esas reglas de fusión solo dependen del hiperparámetro  $C$ , que indica el número mínimo de sensores que apuestan por la hipótesis alternativa para que la regla de fusión decida esa hipótesis. Se han propuesto varios métodos para seleccionar el valor óptimo de  $C$  [1], [11], [20], pero requieren conocer la fiabilidad de las decisiones de los sensores en el FC, es decir, los parámetros del modelo. En cualquier caso, la regla de votación óptima depende en gran medida de los parámetros del modelo. Las llamadas OR, AND y mayoría de votos son casos particulares de las reglas de votación  $C$ , que corresponden a valores específicos de  $C$ . A veces, la mayoría de votos es la regla de votación  $C$  óptima, pero no siempre. Por ejemplo, en la detección cooperativa de espectro se usa comúnmente la regla OR [10].

- Regla de detección localmente óptima (LOD): En [13], los autores derivan el LOD [23] asumiendo que se conocen las probabilidades de falsa alarma de los sensores y las probabilidades de error, pero las probabilidades de detección son desconocidas. La regla

de fusión propuesta es aplicable a redes completamente heterogéneas, pero está restringida al criterio de detección NP.

Ejemplos del segundo enfoque son:

- Regla GLRT ( General likelihood ratio test ) aproximada: En [12], los autores derivan una estadística de prueba GLRT aproximada basada en un estimador ML (máxima verosimilitud) aproximado de la probabilidad de detección de los sensores. Su aplicabilidad está limitada a redes homogéneas (sensores con rendimiento idéntico) y requiere que el FC conozca tanto las probabilidades de error de los BSC como la probabilidad de falsa alarma de los sensores.

- Regla de Wu: También fue propuesta en [12] como una mejora de la regla GLRT. Basándose en la probabilidad de detección estimada utilizada para la GLRT, los autores proponen una prueba diferente (conocida como "regla de Wu" en la literatura técnica [13]) que supera a la prueba GLRT en todos los escenarios considerados. Su aplicación también está limitada a redes homogéneas cuando el FC conoce tanto las probabilidades de error de los BSC como la probabilidad de falsa alarma de los sensores. Es aplicable bajo el criterio de detección NP.

Contribución: En este trabajo fin de grado, presentamos un método general para que el FC aprenda conjuntamente los parámetros del modelo de los sistemas DD-WSN cuando los canales de reporte se modelan como BSC. El método de aprendizaje es ciego en el sentido de que la única información en el FC son las decisiones binarias ruidosas de los sensores. Los parámetros a aprender son las probabilidades de detección y las probabilidades a priori. Se supone que el FC conoce las probabilidades de falsa alarma y las probabilidades de error de los canales BSC.

## 1.4. Descripción del problema

En la figura 1 se muestra un sistema DD-WSN con  $K$  sensores distribuidos espacialmente. Cada sensor va a detectar la ausencia o presencia de un objeto/evento de interés,  $z \in \{0,1\}$ . Las decisiones de los sensores están sujetos a sus probabilidades de detección y falsa alarma locales,  $P_{d,k}$  y  $P_{f,k}$ , siendo  $k$  el índice de los sensores. Los sensores van a reportar estas decisiones ( $y_k$ ) por los canales BSC al FC. Estos canales BSC tienen una probabilidad de error  $e_k$  conocida.

El FC recibe únicamente las decisiones ruidosas de los sensores a la salida de los canales BSC,  $x_k$ , y con esta información va a estimar los parámetros necesarios para posteriormente derivar una regla de fusión con el criterio de mínima probabilidad de error y tomar la decisión final,  $\hat{z}(\mathbf{x})$ .

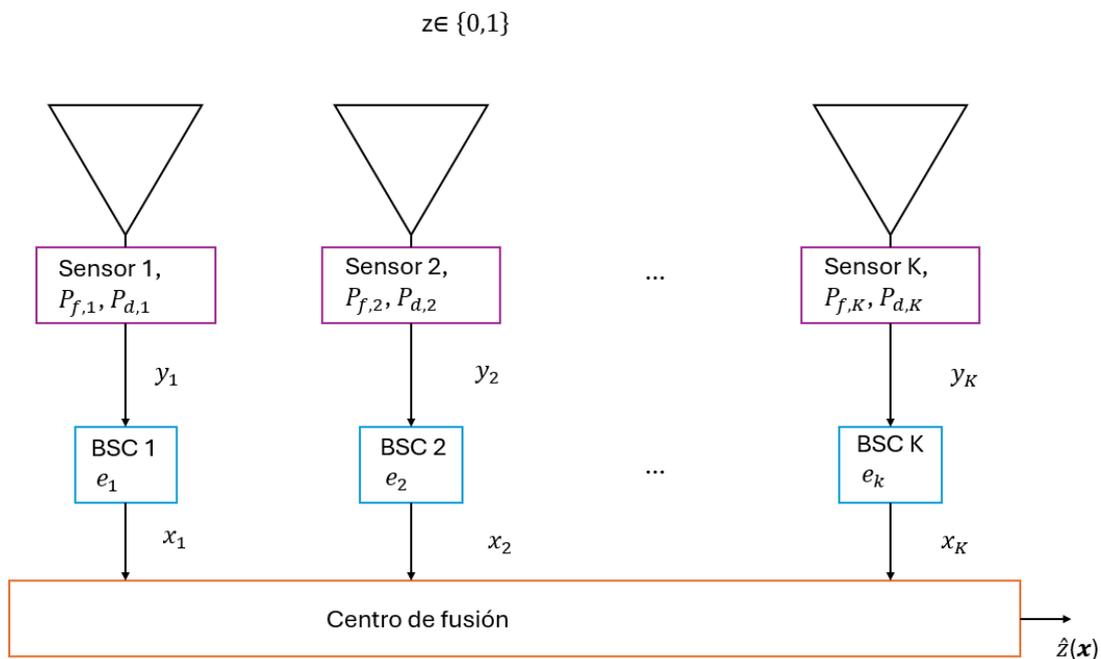


Figura 1: sistema DD-WSN

## 1.5. Estructura de la memoria

### Capítulo 1: Fundamentos Teóricos

En este capítulo del Trabajo de Fin de Grado, se establecen los fundamentos teóricos necesarios para abordar el problema en cuestión. Este capítulo se divide en cuatro subcapítulos:

- Modelo del problema: Se modela matemáticamente el problema en cuestión.
- Estimación de parámetros: Algoritmo EM: Se detalla el problema de estimación de parámetros y se presenta su resolución completa mediante la aplicación del algoritmo EM.
- Detección: Regla de Fusión: Aquí se explica el criterio utilizado para la detección y su aplicación en el contexto del problema en estudio.
- Extensión: desconocimiento de la probabilidad a priori: Se amplía el alcance del problema del TFG para el caso en el que tampoco se conocen la probabilidad a priori y se propone una solución integral.

### Capítulo 2: Implementación Software

Este capítulo ofrece una visión detallada de la implementación software de la solución del problema utilizando la herramienta MATLAB. Se incluye un diagrama que ilustra las funciones utilizadas en el proceso.

### Capítulo 3: Resultados

En este apartado se recopilan todos los resultados obtenidos a partir de las simulaciones realizadas. Se presentan distintas graficas en distintas situaciones para evaluar el funcionamiento del FC y se ofrecen comentarios sobre los mismos.

### Capítulo 4: Conclusiones

Finalmente, se exponen las conclusiones derivadas del trabajo realizado.

## 2. Fundamentos Teóricos

### 2.1. Modelo del Problema

Los sensores del sistema DD-WSN detectan la ausencia o presencia de un evento/objeto de interés,  $z = 0$  o  $z = 1$ . Estas dos hipótesis las denotaremos como  $H_0$  y  $H_1$  respectivamente. Denotaremos con  $y_k \in \{0,1\}$  la decisión del sensor  $k$  sujeta a su probabilidad de detección  $P_{d,k} = P(y_k = 1|H_1)$  y su probabilidad de falsa alarma  $P_{f,k} = P(y_k = 1|H_0)$ . Los sensores informan sus decisiones al FC a través de los canales BSC que están sujetos a una probabilidad de error de bit  $e_k = P(x_k \neq y_k)$ , donde  $x_k$  corresponde a la salida binaria del canal  $k$ . A partir de  $\mathbf{x} = (x_1, x_2, \dots, x_K)$  el FC toma una decisión final de acuerdo con una regla de fusión  $\hat{z}(\mathbf{x}) \in \{0,1\}$ . Asumimos que los sensores no se comunican entre ellos y que no hay feedback del FC a los sensores.

Las probabilidades de falsa alarma y detección en el FC (incluyendo el efecto de los BSC) vienen dadas por [14], [20]:

$$f_k = P(x_k = 1|H_0) = e_k + P_{f,k}(1 - 2e_k) \quad (1)$$

$$d_k = P(x_k = 1|H_1) = e_k + P_{d,k}(1 - 2e_k) \quad (2)$$

Asumimos que las probabilidades de error de los canales BSC toman valores en el intervalo  $0 < e_k < 0.5$ . También asumimos que  $P_{f,k} \leq P_{d,k}$ , por lo tanto las probabilidades están acotadas de la siguiente manera:

$$0 < e_k \leq f_k \leq d_k \leq 1 - e_k < 1 \quad (3)$$

Asumimos que el FC conoce las probabilidades de error de los canales  $\mathbf{e} = (e_1, e_2, \dots, e_K)$  y las probabilidades a priori  $P(H_0) = 1 - u$ ,  $P(H_1) = u$ . También conoce las probabilidades de falsa alarma de los sensores  $\mathbf{f} = (f_1, f_2, \dots, f_K)$  pero no las de detección  $\mathbf{d} = (d_1, d_2, \dots, d_K)$ . Estas suposiciones ocurren en muchos casos prácticos [13], [12], [28].

La salida del canal BSC  $k$ ,  $x_k$ , condicionada a  $H_0$  y  $H_1$  responde a distribuciones de Bernuilli con parámetros  $f_k$  y  $d_k$  respectivamente. Suponiendo que las decisiones de los sensores ( $\mathbf{x}_k$ ) son condicionalmente independientes, las funciones de probabilidad de masa (pmf) condicionadas vienen dadas por:

$$p(\mathbf{x}|H_0) = \prod_{k=1}^K f_k^{x_k} (1 - f_k)^{1-x_k} \quad (4)$$

$$p(\mathbf{x}|H_1, \mathbf{d}) = \prod_{k=1}^K d_k^{x_k} (1 - d_k)^{1-x_k} \quad (5)$$

La distribución marginal de  $\mathbf{x}$  es la mezcla de las pmf condicionales con coeficientes dados por las probabilidades a priori:

$$p(\mathbf{x}|\mathbf{d}) = (1 - u) p(\mathbf{x}|H_0) + u p(\mathbf{x}|H_1, \mathbf{d}) \quad (6)$$

La pmf conjunta de  $\mathbf{x}$  y  $z$  viene dada por:

$$p(\mathbf{x}, z|\mathbf{d}) = [(1 - u)p(\mathbf{x}|H_0)]^{1-z} [u p(\mathbf{x}|H_1, \mathbf{d})]^z \quad (7)$$

## 2.2. Estimación de parámetros: Algoritmo EM

El algoritmo EM es un enfoque aplicable para el cálculo iterativo de estimaciones de ML que se utiliza en problemas de datos incompletos [26], [27]. El objetivo de la estimación de ML es estimar parámetros que maximicen la probabilidad de haber recibido ciertas mediciones de un vector aleatorio distribuido con una función de densidad de probabilidad (p.d.f) que depende de los parámetros a estimar.

En nuestro caso tenemos un vector  $\mathbf{x}_n = (x_{1,n}, x_{2,n}, \dots, x_{K,n})$  donde  $n = 1, 2, \dots, N$  representa el índice a los periodos de sensado y  $N$  es el número total de periodos de sensado. Toda esta información se puede organizar en una matriz  $\mathbf{X}$  de tamaño  $K \times N$ , donde  $\mathbf{x}_n$  es la  $n$ -ésima columna de  $\mathbf{X}$ . La probabilidad de haber recibido toda la serie de mediciones independientes, suponiendo que los vectores  $\mathbf{x}_n$  son independientes es:  $p(\mathbf{X}|\mathbf{d}) = \prod_{n=1}^N p(\mathbf{x}_n|\mathbf{d})$ .

Para una matriz  $\mathbf{X}$ , dada la función de verosimilitudes es  $L(\mathbf{d}) = p(\mathbf{X}|\mathbf{d})$ . La estima de máxima verosimilitud de  $\mathbf{d}$  es  $\hat{\mathbf{d}} = \operatorname{argmax}_{\mathbf{d} \in S_d} \log L(\mathbf{d})$ , donde  $S_d = \{\mathbf{d} | f_k \leq d_k \leq 1 - e_k, \forall k\}$ . Este problema de optimización no tiene una solución cerrada, por lo tanto vamos a emplear el algoritmo EM.

Para emplear el algoritmo EM consideramos las variables ocultas o variables latentes binarias  $\mathbf{z} = (z_1, z_2, \dots, z_N)$  que indican la hipótesis  $H_1$  o  $H_0$  real en cada periodo de sensado. Estas variables no son conocidas. Si las conociésemos el problema de estimación ML sería simple y con solución cerrada. El algoritmo EM es un método iterativo para calcular la estimación de ML cuando no se conocen las variables latentes. Entonces, en lugar de examinar el logaritmo de la función de verosimilitud para los datos observados, consideramos el logaritmo de la función de verosimilitud de los datos completos, estos son las variables observadas  $\mathbf{x}_n$  y las variables latentes  $\mathbf{z}_n$ . La función de verosimilitud de los datos completos es:

$$L_c(\mathbf{d}) = \prod_{n=1}^N p(\mathbf{x}_n, z_n|\mathbf{d}) = \prod_{n=1}^N [(1 - u)p(\mathbf{x}_n|H_0)]^{1-z_n} [u p(\mathbf{x}_n|H_1, \mathbf{d})]^{z_n}.$$

Por lo tanto si  $z_n = 0$ , nos quedaríamos con el primer factor que corresponde a la hipótesis  $H_0$ , y si  $z_n = 1$  nos quedaríamos con el segundo factor que corresponde a la hipótesis  $H_1$ .

El algoritmo EM consta de dos pasos que se repiten iterativamente hasta que converge [25]. Estos pasos son el paso E y el paso M.

El paso E (Expectación), consiste en calcular la esperanza de la función de verosimilitud de los datos completos condicionada a los datos observados y estima de los parámetros obtenidos en la iteración anterior.

$$Q^{(i)}(\mathbf{d}) = \mathbb{E}[\log L_c(\mathbf{d})|\mathbf{X}, \hat{\mathbf{d}}^{(i-1)}] \quad (8)$$

donde  $i$  es el índice de iteraciones,  $\hat{\mathbf{d}}^{(i)}$  es la estima de  $\mathbf{d}$  después de la iteración  $i$  y  $\log L_c(\mathbf{d})$  es el logaritmo de la función de verosimilitud de los datos completos.

El paso M (Maximización), se maximiza la esperanza calculada en el paso anterior con respecto a los parámetros del modelo.

$$\hat{\mathbf{d}}^{(i)} = \operatorname{argmax}_{\mathbf{d} \in S_d} Q^{(i)}(\mathbf{d}) \quad (9)$$

### 2.2.1. Paso E

En el paso E, siguiendo las ecuaciones de nuestro problema tendríamos:

$$Q^{(i)}(\mathbf{d}) = \mathbb{E}\left[\sum_{n=1}^N \log [(1-u)p(\mathbf{x}_n|H_0)]^{1-z_n} [u p(\mathbf{x}_n|H_1, \mathbf{d})]^{z_n} |X, \hat{\mathbf{d}}^{(i-1)}]\right]$$

$$= \mathbb{E}\left[\sum_{n=1}^N (1-z_n) \log [(1-u)p(\mathbf{x}_n|H_0)] + z_n \log [u p(\mathbf{x}_n|H_1, \mathbf{d})] |X, \hat{\mathbf{d}}^{(i-1)}\right] \quad (10)$$

Y debido a la linealidad del operador  $\mathbb{E}$ :

$$Q^{(i)}(\mathbf{d}) = \sum_{n=1}^N \mathbb{E}[(1-z_n)|X, \hat{\mathbf{d}}^{(i-1)}] \log [(1-u)p(\mathbf{x}_n|H_0)]$$

$$+ \mathbb{E}[z_n|X, \hat{\mathbf{d}}^{(i-1)}] \log [u p(\mathbf{x}_n|H_1, \mathbf{d})] =$$

$$\sum_{n=1}^N (1 - \mathbb{E}[z_n]|X, \hat{\mathbf{d}}^{(i-1)}) \log [(1-u)p(\mathbf{x}_n|H_0)] + \mathbb{E}[z_n|X, \hat{\mathbf{d}}^{(i-1)}] \log [u p(\mathbf{x}_n|H_1, \mathbf{d})] \quad (11)$$

donde

$$\mathbb{E}[z_n|X, \hat{\mathbf{d}}^{(i-1)}] = 0 * P(z_n = 0|x_n, \hat{\mathbf{d}}^{(i-1)}) + 1 * P(z_n = 1|x_n, \hat{\mathbf{d}}^{(i-1)}) =$$

$$P(z_n = 1|x_n, \hat{\mathbf{d}}^{(i-1)}) \quad (12)$$

Aplicando el Teorema de Bayes

$$P(z_n = 1|x_n, \hat{\mathbf{d}}^{(i-1)}) = \frac{u p(\mathbf{x}_n|H_1, \hat{\mathbf{d}}^{(i-1)})}{(1-u)p(\mathbf{x}_n|H_0) + u p(\mathbf{x}_n|H_1, \hat{\mathbf{d}}^{(i-1)})} \quad (13)$$

Para simplificar la notación, vamos a denominar este cociente como  $t_n^{(i)}$ , que se puede interpretar como una estima de  $z_n$  después de observar  $\mathbf{x}_n$  cuando las probabilidades de detección son  $\hat{\mathbf{d}}^{(i-1)}$ .

Por lo tanto tendríamos:

$$Q^{(i)}(\mathbf{d}) = \sum_{n=1}^N (1 - t_n^{(i)}) \log [(1-u)p(\mathbf{x}_n|H_0)] + t_n^{(i)} \log [u p(\mathbf{x}_n|H_1, \mathbf{d})] \quad (14)$$

Una vez hemos llegado a esta ecuación, procedemos a realizar el paso M.

### 2.2.2. Paso M

Para ello sustituimos las pmf (4) y (5) en la ecuación (14)

$$Q^{(i)}(\mathbf{d}) = \sum_{n=1}^N (1 - t_n^{(i)}) \log (1-u) + (1 - t_n^{(i)}) \log \prod_{k=1}^K f_k^{x_{k,n}} (1 - f_k)^{1-x_{k,n}}$$

$$+ t_n^{(i)} \log u + t_n^{(i)} \log \prod_{k=1}^K d_k^{x_{k,n}} (1 - d_k)^{1-x_{k,n}}$$

$$= \sum_{n=1}^N (1 - t_n^{(i)}) \log (1-u) + (1 - t_n^{(i)}) \sum_{k=1}^K (x_{k,n} \log f_k + (1 - x_{k,n}) \log (1 - f_k)) +$$

$$t_n^{(i)} \log u + t_n^{(i)} \sum_{k=1}^K (x_{k,n} \log d_k + (1 - x_{k,n}) \log (1 - d_k)) \quad (15)$$

Despreciando los términos de  $Q^{(i)}(\mathbf{d})$  que no contengan  $\mathbf{d}$ .

$$\hat{\mathbf{d}}^{(i)} = \underset{\mathbf{d} \in S_d}{\operatorname{argmax}} Q^{(i)}(\mathbf{d}) = \underset{\mathbf{d} \in S_d}{\operatorname{argmax}} \sum_{n=1}^N t_n^{(i)} \sum_{k=1}^K (x_{k,n} \log d_k + (1 - x_{k,n}) \log(1 - d_k)) \quad (16)$$

Como los términos que dependen de  $d_k$  están desacoplados del resto, se puede fragmentar (16) en  $K$  problemas independientes.

$$\hat{d}_k^{(i)} = \underset{f_k \leq d_k \leq 1 - e_k}{\operatorname{argmax}} N_{k,1,1}^{(i)} \log d_k + N_{k,0,1}^{(i)} \log(1 - d_k) \quad (17)$$

donde  $N_{k,1,1}^{(i)} = \sum_{n=1}^N x_{k,n} t_n^{(i)}$ ,  $N_{k,0,1}^{(i)} = \sum_{n=1}^N (1 - x_{k,n}) t_n^{(i)}$

Este problema de optimización, es un problema de variable única, no lineal con restricciones. Estas restricciones son  $f_k - d_k \leq 0$ , y  $d_k - (1 - e_k) \leq 0$ .

En (17) la función objetivo la denotaremos como  $M(d_k) = N_{k,1,1}^{(i)} \log d_k + N_{k,0,1}^{(i)} \log(1 - d_k)$ , y las restricciones como  $g_1(d_k) = f_k - d_k$ ,  $g_2(d_k) = d_k - (1 - e_k)$ .

Las derivadas de estas funciones son:

$$\begin{aligned} \frac{\partial M(d_k)}{\partial d_k} &= \frac{N_{k,1,1}^{(i)}}{d_k} - \frac{N_{k,0,1}^{(i)}}{1 - d_k} \\ \frac{\partial g_1(d_k)}{\partial d_k} &= -1 \\ \frac{\partial g_2(d_k)}{\partial d_k} &= 1 \end{aligned}$$

El conjunto factible,  $[f_k, 1 - e_k]$ , es compacto y convexo, y la función objetivo es diferenciable, acotada y estrictamente cóncava en el conjunto factible, por lo tanto habrá solo un máximo. Como no hay puntos irregulares asociadas a las restricciones, la solución será el único valor de  $d_k$  que cumpla las condiciones KKT (Karush-Kuhn-Tucker):

$$\begin{aligned} -\frac{N_{k,1,1}^{(i)}}{d_k} + \frac{N_{k,0,1}^{(i)}}{1 - d_k} - \lambda_1 + \lambda_2 &= 0 \\ \lambda_1(f_k - d_k) &= 0 \\ \lambda_2(d_k - (1 - e_k)) &= 0 \\ f_k \leq d_k, d_k \leq 1 - e_k, \lambda_1 \geq 0, \lambda_2 \geq 0 \end{aligned}$$

Donde  $\lambda_1, \lambda_2$ , son los multiplicadores de KKT.

Particularizamos para los posibles valores de  $d_k$  en el conjunto factible.

- Si  $d_k$  fuese un punto interno, es decir  $f_k < d_k < 1 - e_k$ , entonces se tiene que cumplir que  $\lambda_1 = 0$ ,  $\lambda_2 = 0$ .

Por lo tanto las condiciones KKT se reducen a

$$\begin{aligned} -\frac{N_{k,1,1}^{(i)}}{d_k} + \frac{N_{k,0,1}^{(i)}}{1 - d_k} &= 0 \\ f_k \leq d_k, d_k \leq 1 - e_k \end{aligned}$$

Por lo tanto obtenemos

$$\hat{d}_k^{(i)} = \frac{N_{k,1,1}^{(i)}}{N_1^{(i)}}$$

donde  $N_1^{(i)} = N_{k,1,1}^{(i)} + N_{k,0,1}^{(i)} = \sum_n t_n^{(i)}$

Y las desigualdades requieren

$$f_k < \frac{N_{k,1,1}^{(i)}}{N_1^{(i)}} < 1 - e_k$$

- Si  $d_k$  estuviese en el punto  $d_k = f_k$ , entonces se tiene que cumplir que  $\lambda_2 = 0$ .

Por lo tanto las condiciones KKT se reducen a

$$\begin{aligned} -\frac{N_{k,1,1}^{(i)}}{d_k} + \frac{N_{k,0,1}^{(i)}}{1 - d_k} - \lambda_1 &= 0 \\ f_k &= d_k \\ \lambda_1 &\geq 0 \end{aligned}$$

Entonces  $\hat{d}_k^{(i)} = f_k$ , y la desigualdad requiere  $\frac{N_{k,1,1}^{(i)}}{N_1^{(i)}} \leq f_k$

- Si  $d_k$  estuviese en el punto  $d_k = (1 - e_k)$ , entonces se tiene que cumplir que  $\lambda_1 = 0$ .

Por lo tanto las condiciones KKT se reducen a

$$\begin{aligned} -\frac{N_{k,1,1}^{(i)}}{d_k} + \frac{N_{k,0,1}^{(i)}}{1 - d_k} + \lambda_2 &= 0 \\ d_k &= (1 - e_k) \\ \lambda_2 &\geq 0 \end{aligned}$$

Entonces  $\hat{d}_k^{(i)} = 1 - e_k$ , y la desigualdad requiere  $\frac{N_{k,1,1}^{(i)}}{N_1^{(i)}} \leq 1 - e_k$

Por lo tanto, la solución única es

$$\hat{d}_k^{(i)} = \begin{cases} \frac{N_{k,1,1}^{(i)}}{N_1^{(i)}} & \text{si } f_k < \frac{N_{k,1,1}^{(i)}}{N_1^{(i)}} < 1 - e_k \\ f_k & \text{si } \frac{N_{k,1,1}^{(i)}}{N_1^{(i)}} \leq f_k \\ 1 - e_k & \text{si } \frac{N_{k,1,1}^{(i)}}{N_1^{(i)}} \leq 1 - e_k \end{cases} \quad (18)$$

El algoritmo EM requiere una estima inicial de  $\mathbf{d}$ , que denominaremos  $\hat{\mathbf{d}}^{(0)}$ . Para ello escogemos unos valores iniciales para  $t_n^{(0)}$  y aplicamos el paso M para obtener  $\hat{\mathbf{d}}^{(0)}$ . Eligiendo  $t_n^{(i)} \in (0,1)$ , entonces  $N_1^{(0)} \in (0, N)$ , por lo tanto  $0 < \frac{N_{k,1,1}^{(0)}}{N_1^{(0)}} < 1$  y  $\hat{\mathbf{d}}^{(0)} \in S_d$ . En las simulaciones del capítulo 4, se han utilizado los siguientes valores iniciales.

$$t_n^{(0)} = \begin{cases} \epsilon_t & \text{si } \mathbf{x}_n^T \mathbf{1}_K < K/2 \\ 0.5 & \text{si } \mathbf{x}_n^T \mathbf{1}_K = K/2 \\ 1 - \epsilon_t & \text{si } \mathbf{x}_n^T \mathbf{1}_K > K/2 \end{cases}$$

Con  $\epsilon_t = 10^{-3}$

### 2.3. Estimador supervisado

Como se mencionó previamente, hemos realizado la estima de  $\mathbf{d}$  sin tener conocimiento de  $\mathbf{z}$ , es decir, sin saber qué hipótesis real se está produciendo en cada periodo de sensado. También es interesante explorar cómo sería el estimador ML en el caso de tener conocimiento de la hipótesis, es decir, si supiéramos  $z_n$ . Esto se denomina estimador supervisado, donde las observaciones serían  $\mathbf{X}$  y  $\mathbf{z}$ .

A partir de (7) la log-verosimilitud sería  $\log p(\mathbf{X}, \mathbf{z} | \mathbf{d}) = \sum_{n=1}^N (1 - z_n) \log[(1 - u)p(\mathbf{x}_n | H_0)] + z_n \log[u p(\mathbf{x}_n | H_1, \mathbf{d})]$  que coincide con (14), cambiando  $t_n^{(i)}$  por  $z_n$ . Por eso la solución del Paso M tiene la misma forma del caso supervisado. De manera que en la ecuación (18)  $N_{k,1,1}^{(i)}$  y  $N_1^{(i)}$  se obtienen a partir de los  $z_n$  en vez de los  $t_n^{(i)}$ :

$$N_{k,1,1}^{(i)} = \sum_{n=1}^N x_{k,n} z_n, N_{k,0,1}^{(i)} = \sum_{n=1}^N (1 - x_{k,n}) z_n, N_1^{(i)} = N_{k,1,1}^{(i)} + N_{k,0,1}^{(i)} = \sum_n z_n$$

Finalmente, el estimado supervisado sería:

$$\hat{\mathbf{d}} = \operatorname{argmax}_{\mathbf{d} \in S_d} \log L_c(\mathbf{d}) = \hat{\mathbf{d}}_{ML}$$

## 2.4. Detección: Regla de Fusión

En los sistemas de fusión de decisiones, el FC toma la decisión final a partir de las decisiones reportadas por los sensores individuales. Por lo tanto, el objetivo final es diseñar una regla de fusión precisa, que generalmente dependerá de los parámetros del modelo y la precisión de las estimaciones de  $\hat{\mathbf{d}}$ .

$\mathbf{x}$  es un vector aleatorio cuyo espacio muestral es el conjunto de K-tuplas binarias  $\Omega_x$  con cardinalidad  $|\Omega_x| = 2^K$ . El FC toma la decisión final a partir de  $\mathbf{x}$  según una regla  $\hat{z}(\mathbf{x}) \in \{0,1\}$ , que será una función lógica con  $K$  entradas binarias y una salida binaria.

Las probabilidades de falsa alarma y detección de la regla de fusión  $\hat{z}(\mathbf{x})$  son:

$$P_f = P(\hat{z}(\mathbf{x}) = 1|H_0) = \sum_{\mathbf{x} \in \Omega_x} \hat{z}(\mathbf{x})p(\mathbf{x}|H_0)$$

$$P_d = P(\hat{z}(\mathbf{x}) = 1|H_1) = \sum_{\mathbf{x} \in \Omega_x} \hat{z}(\mathbf{x})p(\mathbf{x}|H_1, \mathbf{d})$$

Y la probabilidad de error es

$$P_e = P(\hat{z}(\mathbf{x}) \neq z) = P_f(1 - u) + (1 - P_d)u = \sum_{\mathbf{x} \in \Omega_x} \hat{z}(\mathbf{x})p(\mathbf{x}|\mathbf{d})$$

Vemos que el rendimiento de la regla de fusión depende del parámetro a estimar  $\mathbf{d}$ .

Consideramos el enfoque bayesiano para la regla de fusión cuyo objetivo es minimizar el riesgo Bayesiano [23], [24]:

$$B = \sum_{i=0}^1 \sum_{j=0}^1 P(\hat{z}(\mathbf{x}) = i|z = j)P(z = j)C_{i,j}$$

Donde  $C_{i,j}$  es el coste de la decisión global  $\hat{z}(\mathbf{x}) = i$  cuando la verdad es  $z = j$  y las probabilidades a priori son  $P(z = 1) = u$  y  $P(z = 0) = 1 - u$ , la expresión anterior supone que  $C_{0,0} < C_{1,0}$  y  $C_{1,1} < C_{0,1}$ .

el detector que minimiza el riesgo Bayesiano es [23]

$$\frac{P(H_1|\mathbf{x}, \mathbf{d})}{P(H_0|\mathbf{x})} \underset{H_0}{\underset{H_1}{\gtrless}} \frac{C_{1,0} - C_{0,0}}{C_{0,1} - C_{1,1}} \rightarrow \frac{p(\mathbf{x}|H_1, \mathbf{d})}{p(\mathbf{x}|H_0)} \underset{H_0}{\underset{H_1}{\gtrless}} \frac{1-u}{u} \frac{C_{1,0} - C_{0,0}}{C_{0,1} - C_{1,1}}$$

En este problema consideramos el detector de mínima probabilidad de error que es un caso particular donde  $C_{1,0} = C_{0,1} = 1$ ,  $C_{0,0} = C_{1,1} = 0$ . En este caso el coste Bayesiano coincide con la probabilidad de error.

Por lo tanto el detector será

$$\frac{p(\mathbf{x}|H_1, \mathbf{d})}{p(\mathbf{x}|H_0)} \underset{H_0}{\underset{H_1}{\gtrless}} \frac{1-u}{u}$$

Aplicando el logaritmo a ambos lados de la ecuación se tiene

$$\hat{z}(\mathbf{x}) = \begin{cases} 1 & \text{si } \Delta(\mathbf{x}|\mathbf{d}) > \gamma \\ 0 & \text{si } \Delta(\mathbf{x}|\mathbf{d}) \leq \gamma \end{cases}$$

donde

$$\Delta(\mathbf{x}|\mathbf{d}) = \log \left( \frac{p(\mathbf{x}|H_1, \mathbf{d})}{p(\mathbf{x}|H_0)} \right)$$

$$\gamma = \log \left( \frac{1-u}{u} \right)$$

Como vemos, para aplicar la regla de decisión, el FC tiene que saber el parámetro  $\mathbf{d}$ , y en nuestro problema este parámetro es desconocido. En este trabajo se propone usar la regla de decisión con la estima de  $\mathbf{d}$  dada por el algoritmo EM. En el capítulo 2.2 se explica detalladamente como hemos estimado  $\mathbf{d}$  con el algoritmo EM.

$$\hat{z}(\mathbf{x}) = \begin{cases} 1 & \text{si } \Delta(\mathbf{x}|\hat{\mathbf{d}}) > \gamma \\ 0 & \text{si } \Delta(\mathbf{x}|\hat{\mathbf{d}}) \leq \gamma \end{cases} \quad (19)$$

Alternativamente, las reglas de fusión ciegas no necesitan saber los parámetros del modelo. La más popular es la regla del voto mayoritario, donde la decisión final se toma mediante un proceso de votación, donde cada sensor tiene un voto igual y la decisión final se determina por mayoría de votos.

$$\hat{z}(\mathbf{x}) = \begin{cases} 1 & \text{si } \mathbf{x}^T \mathbf{1}_K > C \\ 0 & \text{si } \mathbf{x}^T \mathbf{1}_K \leq C \end{cases}, \quad C = (K + 1)/2$$

## 2.5. Extensión: desconocimiento de las probabilidades a priori

Una vez que hemos abordado la resolución de este problema de detección cuando la probabilidad a priori es conocida, resulta interesante analizar el caso en el que es desconocida. Como hemos observado, el umbral de la regla de decisión depende de  $u$ . Se propone sustituir  $u$  por su estima  $\hat{u}$ .

### 2.5.1. Estimación de parámetros $\mathbf{d}$ y $u$

#### 2.5.1.1. Paso E

En este caso tendríamos

$$Q^{(i)}(\mathbf{d}, u) = \mathbb{E}[\log L_c(\mathbf{d}, u) | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}]$$

Siguiendo el procedimiento del paso E (Mirar 6.1 Apéndice A: Resolución paso E para la extensión)

$$Q^{(i)}(\mathbf{d}, u) = \sum_{n=1}^N (1 - t_n^{(i)}) \log[(1 - u)p(\mathbf{x}_n | H_0)] + t_n^{(i)} \log[u p(\mathbf{x}_n | H_1, \mathbf{d})]$$

$$\text{donde } t_n^{(i)} = \frac{\hat{u}^{(i-1)} p(\mathbf{x}_n | H_1, \hat{\mathbf{d}}^{(i-1)})}{(1 - \hat{u}^{(i-1)}) p(\mathbf{x}_n | H_0) + \hat{u}^{(i-1)} p(\mathbf{x}_n | H_1, \hat{\mathbf{d}}^{(i-1)})}$$

nótese que tiene la misma forma que (13) cambiando  $u$  por  $\hat{u}^{(i-1)}$ .

#### 2.5.1.2. Paso M

$$\{\hat{\mathbf{d}}^{(i)}, \hat{u}^{(i)}\} = \operatorname{argmax}_{\mathbf{d} \in \mathcal{S}_d, 0 < u < 1} Q^{(i)}(\mathbf{d}, u) =$$

$$\operatorname{argmax}_{\mathbf{d} \in \mathcal{S}_d, 0 < u < 1} \sum_{n=1}^N (1 - t_n^{(i)}) \log(1 - u) + t_n^{(i)} \log u + t_n^{(i)} \sum_{k=1}^K (x_{k,n} \log d_k + (1 - x_{k,n}) \log(1 - d_k))$$

Los términos que dependen de  $u$  y  $\mathbf{d}$  están desacoplados, por lo tanto, este problema de optimización se puede separar en dos problemas independientes.

$$\hat{u}^{(i)} = \operatorname{argmax}_{0 < u < 1} \sum_{n=1}^N (1 - t_n^{(i)}) \log(1 - u) + t_n^{(i)} \log u$$

$$\hat{\mathbf{d}}^{(i)} = \operatorname{argmax}_{\mathbf{d} \in \mathcal{S}_d} \sum_{n=1}^N t_n^{(i)} \sum_{k=1}^K (x_{k,n} \log d_k + (1 - x_{k,n}) \log(1 - d_k))$$

El problema de optimización para  $\mathbf{d}$  es el mismo que el del capítulo 2.2, su solución viene dada por (18).

El problema de optimización para  $u$  se puede escribir así:

$$\hat{u}^{(i)} = \operatorname{argmax}_{0 < u < 1} N_0^{(i)} \log(1 - u) + N_1^{(i)} \log u$$

donde

$$N_0^{(i)} = \sum_{n=1}^N (1 - t_n^{(i)}), N_1^{(i)} = \sum_{n=1}^N t_n^{(i)}$$

Si hacemos la derivada de la función a maximizar y la igualamos a cero, el resultado es (mirar 6.2 Apéndice B: Resolución Paso M para la Extensión):

$$\hat{u}^{(i)} = \frac{N_1^{(i)}}{N}$$

### 2.5.2. Estimador supervisado para $\mathbf{d}$ y $u$

En el estimador supervisado tenemos conocimiento de las  $\mathbf{z}$ , por lo tanto las observaciones serían  $\mathbf{X}$  y  $\mathbf{z}$ . Repetimos el mismo proceso que en el capítulo 2.3 pero teniendo en cuenta que  $u$  también es desconocida. Finalmente obtenemos:

$$\hat{\mathbf{d}} = \operatorname{argmax}_{\mathbf{d} \in S_d} \log L_c(\mathbf{d}) = \hat{\mathbf{d}}_{ML}$$

$$\hat{u} = \operatorname{argmax}_{0 < u < 1} \log L_c(u) = \hat{u}_{ML}$$

### 2.5.3. Detección

Seguiríamos la teoría y procedimientos del capítulo 2.4 y a partir de (19) sustituyendo  $u$  por  $\hat{u}$ , se obtiene:

$$\Delta(\mathbf{x}|\hat{\mathbf{d}}) = \log \left( \frac{p(\mathbf{x}|H_1, \hat{\mathbf{d}})}{p(\mathbf{x}|H_0)} \right)$$

$$\hat{\gamma} = \log \left( \frac{1 - \hat{u}}{\hat{u}} \right)$$

$$\hat{z}(\mathbf{x}) = \begin{cases} 1 & \text{si } \Delta(\mathbf{x}|\hat{\mathbf{d}}) > \hat{\gamma} \\ 0 & \text{si } \Delta(\mathbf{x}|\hat{\mathbf{d}}) \leq \hat{\gamma} \end{cases}$$

### 3. Implementación software

A continuación se presenta un desglose de todas las funciones implementadas en lenguaje MATLAB para obtener los resultados del capítulo 4.

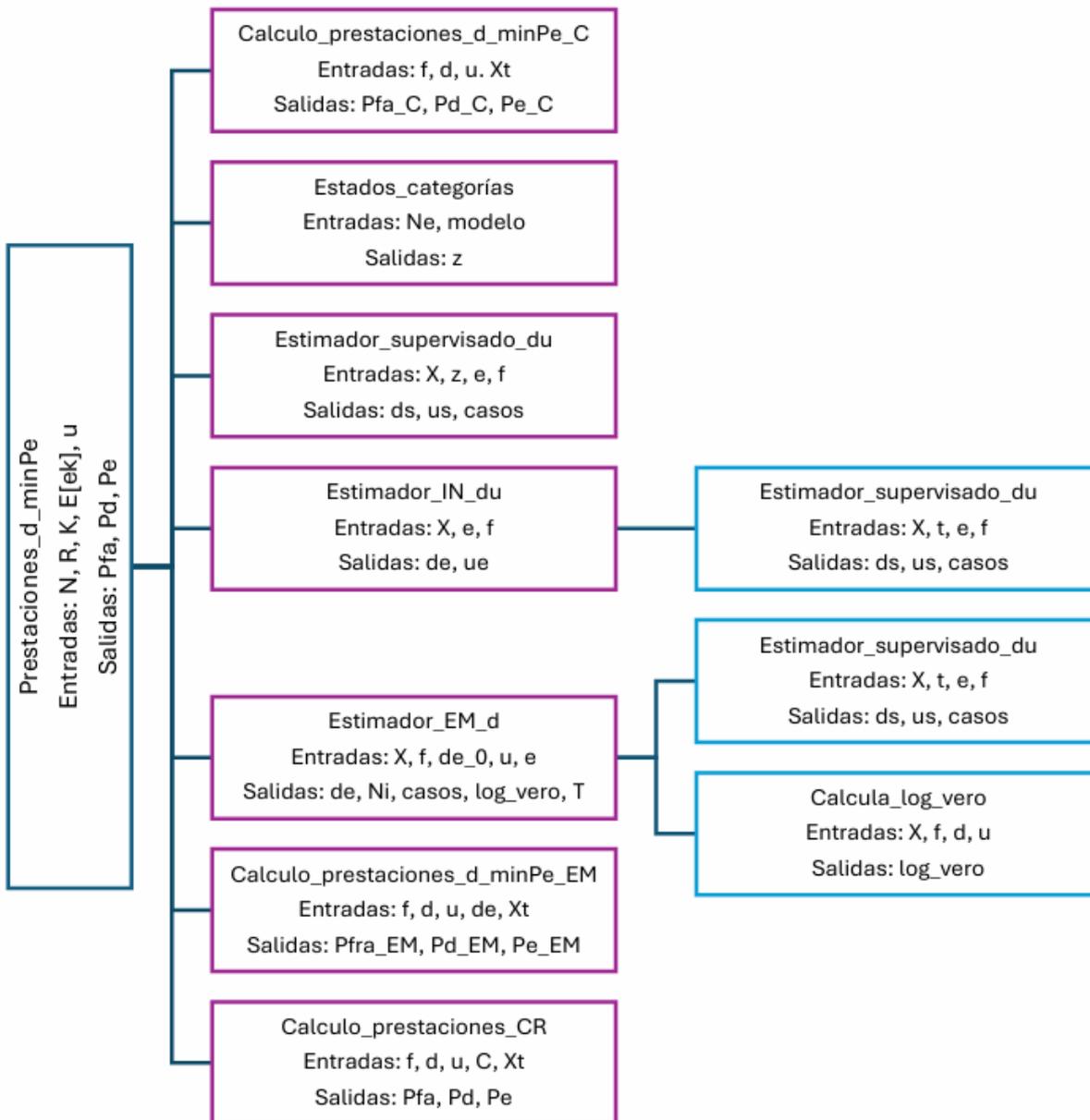


Figura 2: Funciones matlab

Partimos de un programa principal *prestaciones\_d\_minPe* que tiene como datos de entrada el número de periodos de sensado, el número de simulaciones de MonteCarlo, el número de sensores, la probabilidad de error medio de los canales BSC y la probabilidad a priori,  $N$ ,  $R$ ,  $K$ ,  $E[ek]$  y  $u$  respectivamente y como salida las prestaciones del FC. En este programa se realizan  $R$  simulaciones de Montecarlo donde se dan valores aleatorios a las probabilidades de falsa alarma y de detección de los sensores y se llamarán a las respectivas funciones detalladas a continuación.

La primera función llamada es *calculo\_prestaciones\_d\_minPe\_C*, encargada de calcular las prestaciones del detector Clairvoyant, el cual utiliza todos los datos de entrada sin realizar estimaciones, representando así el caso óptimo al que nos queremos aproximar. Dentro de esta función se calcula un estadístico que es el cociente de las verosimilitudes con el valor real de  $\mathbf{d}$ , y se calculan las prestaciones con el umbral del criterio de mínima probabilidad de error.

La siguiente función implementada es *estados\_categorias*, la cual calcula los valores reales de  $z$  para utilizarlos en el estimador supervisado. A continuación se utiliza la función *estimador\_supervisado\_du* para obtener la estima del supervisado.

Posteriormente, la función *estimador\_IN\_du* obtiene una estima inicial de  $\mathbf{d}$ . Para ello, como se explica en el capítulo de *Estimación de parámetros: Algoritmo EM* de este trabajo, define una casuística inicial para  $t^{(0)}$  y a partir de ahí aplica el paso M, llamando a la función *estimador\_supervisado\_du*. Una vez obtenida una estima inicial de  $\mathbf{d}$ , se pasa a la función *estimador\_EM\_d*, que tras realizar  $N_i$  iteraciones, devuelve la estima final de  $\mathbf{d}$ .

Esta función para estimar  $\mathbf{d}$  se divide en dos bucles: el primero corresponde únicamente a la primera iteración, ya que parte del valor inicial de  $\mathbf{d}$  de la función anterior, y el siguiente bucle corresponde a las demás iteraciones, utilizando los datos de la iteración anterior y teniendo un criterio de parada de convergencia. Cuando esta función ha terminado, proporciona la estima de  $\mathbf{d}$  aplicando el algoritmo EM.

A partir de aquí, se calculan las prestaciones del detector que estima  $\mathbf{d}$  con el algoritmo EM con la función *calculo\_prestaciones\_d\_minPe\_EM*, aplicando el umbral de detección del criterio de mínima probabilidad de error.

Finalmente, la última función calcula las prestaciones utilizando reglas de fusión ciegas.

## 4. Resultados

Los resultados que se presentan se han obtenido realizando  $R=5 \times 10^4$  simulaciones independientes de MonteCarlo. Las probabilidades de detección y falsa alarma de los sensores se eligen de acuerdo a las siguientes distribuciones uniformes:

$$P_{f,k} \sim U(0,0.1), P_{d,k} - P_{f,k} \sim U(0,0.9) \quad (20)$$

Asumimos que los sensores emplean señalización On-Off para transmitir decisiones al FC [12], [13], [28]. La señal recibida en el FC desde el sensor  $k$  se modela como  $r_{k,n} = h_k y_{k,n} + w_{k,n}$ , donde  $h_k$  representa la respuesta del canal y  $w_{k,n}$  es un ruido blanco gaussiano con distribución:  $w_{k,n} \sim \mathcal{N}_{\mathbb{C}}(0,1)$ . Se supone que el FC conoce  $h_k$  y utiliza detección coherente con umbral  $h_k/2$  para obtener  $x_{k,n}$  de  $r_{k,n}$ , resultando en una tasa de error de bit  $e_k = Q(\frac{|h_k|}{2})$ . Además, asumimos que los canales experimentan desvanecimientos Rayleigh independientes, con  $h_k \sim \mathcal{N}_{\mathbb{C}}(0, \sigma_h^2)$ .

### 4.1. Prestaciones del estimador de $\mathbf{d}$ y de la regla de decisión

Se compararán las prestaciones de la regla de decisión del FC con las del detector Clairvoyant, que opera con conocimiento completo de las probabilidades de detección  $\mathbf{d}$ , aplicando el test de cociente de verosimilitudes (LRT). La métrica de evaluación del detector será la probabilidad de error. La leyenda de la curva de probabilidad de error del detector Clairvoyant se identificará como LRT, mientras que la curva de probabilidad de error del detector que emplea el algoritmo EM para estimar  $\mathbf{d}$  será denotada como EM-d, y la del supervisado como S-d. Por otro lado, las prestaciones del estimador se evaluarán utilizando el RMSE con la siguiente formula:

$$RMSE = \sqrt{\frac{1}{RK} \sum_{r=1}^R \|\hat{\mathbf{d}}^{(r)} - \mathbf{d}^{(r)}\|^2} \quad (21)$$

Donde  $\hat{\mathbf{d}}^{(r)}$  es el valor de la estima de  $\mathbf{d}$  en la simulación  $r$ , y  $\mathbf{d}^{(r)}$  es el valor real de  $\mathbf{d}$  en la simulación  $r$ , por lo tanto cuanto menor sea el valor de RMSE mejor será la estima.

Se presentan las curvas de RMSE y Probabilidad de error ( $P_e$ ) en función del número de periodos de sensado  $N$ . Estos resultados se observan con  $K=6$  sensores, una probabilidad de error medio de los canales BSC de  $E[ek]=0.05$  y probabilidad a priori  $u=0.5$ .

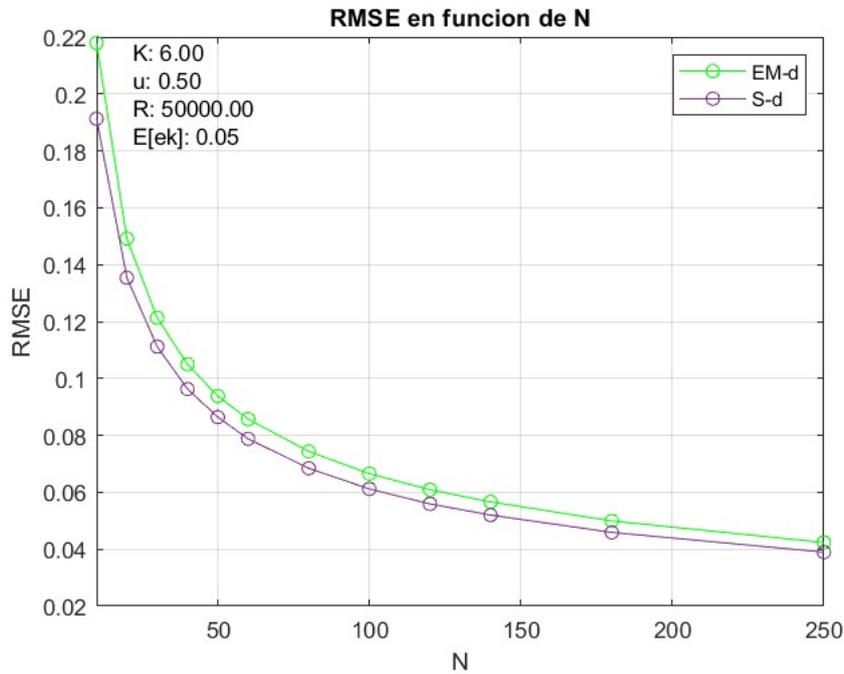


Figura 3: RMSE en función del número de periodos de sensado

Dado que  $N$  solo afecta a la estimación de  $\mathbf{d}$ , se observa que a medida que  $N$  aumenta, las prestaciones del algoritmo EM y supervisado mejoran, mientras que el detector Clairvoyant no depende de  $N$ .

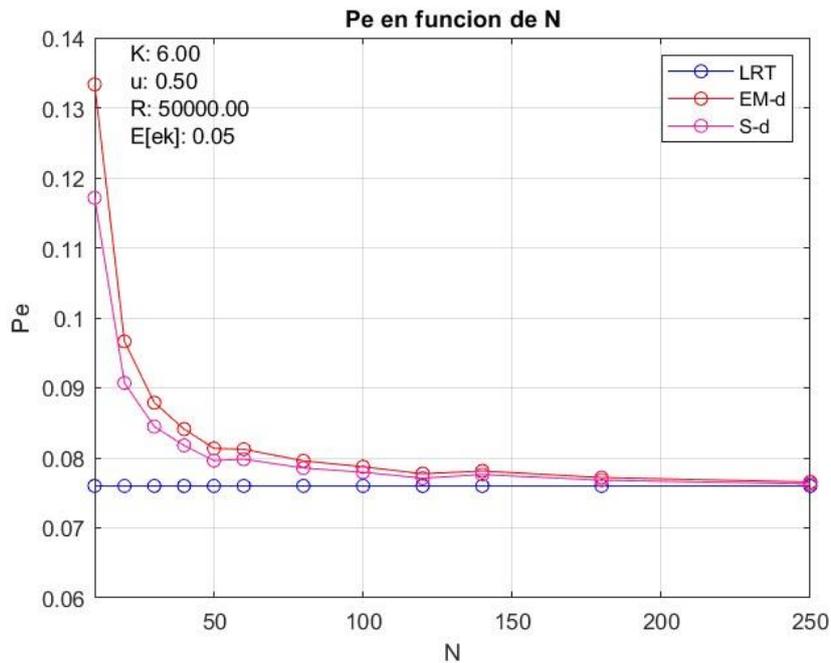


Figura 4: Probabilidad de error en función del número de periodos de sensado

A continuación, se presentan las curvas de RMSE y Probabilidad de Error en función de la probabilidad a priori  $u$ . Estos resultados se observan tras  $N=40$  periodos de sensado.

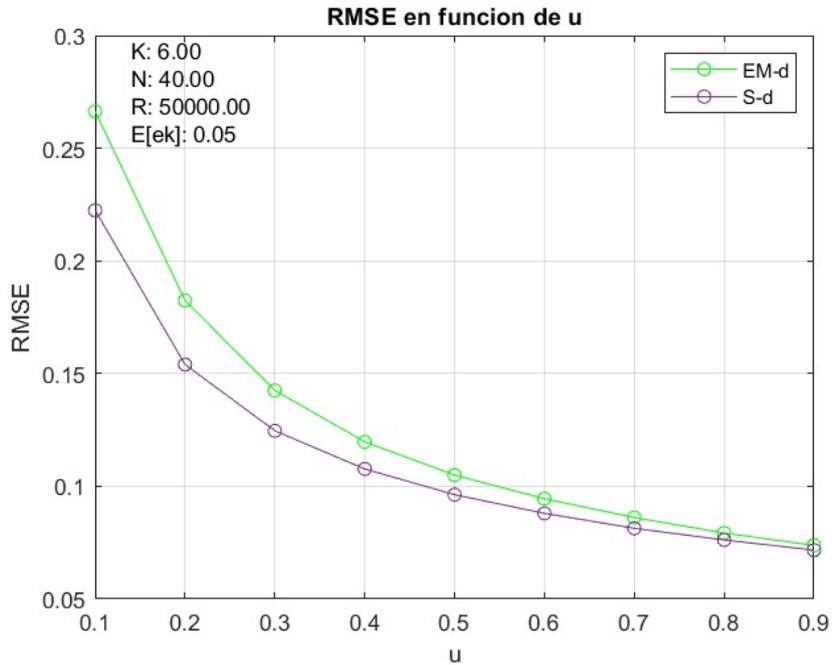


Figura 5: RMSE en función de la probabilidad a priori

Los estimadores mejoran a medida que  $u$  aumenta, esto se debe a que, cuanto mayor es  $u$ , más observaciones se encuentran bajo la hipótesis  $H_1$ , por lo tanto, se estima mejor  $d$ .

La probabilidad de error alcanza sus valores más desfavorables en alrededor de  $u = 0.5$ , esto se debe a que cuando  $u = 0.5$  la probabilidad a priori de las hipótesis  $H_1$  y  $H_0$  es idéntica, lo que resulta en una distribución equitativa entre ambas hipótesis.

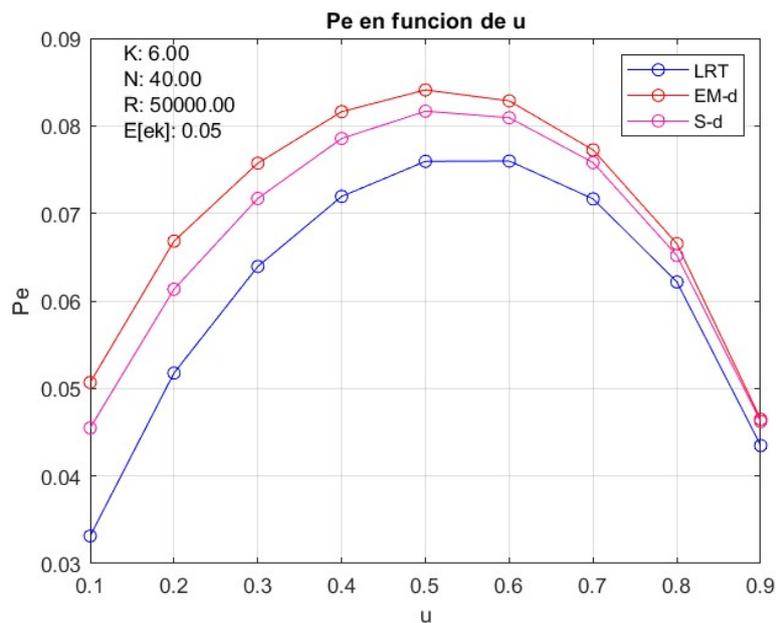


Figura 6: Probabilidad de error en función de la probabilidad a priori

En las siguientes graficas se presentan las curvas de RMSE y Pe en función del número de sensores. El estimador EM mejora a medida que aumenta el número de sensores, mientras que el supervisado se mantiene constante.

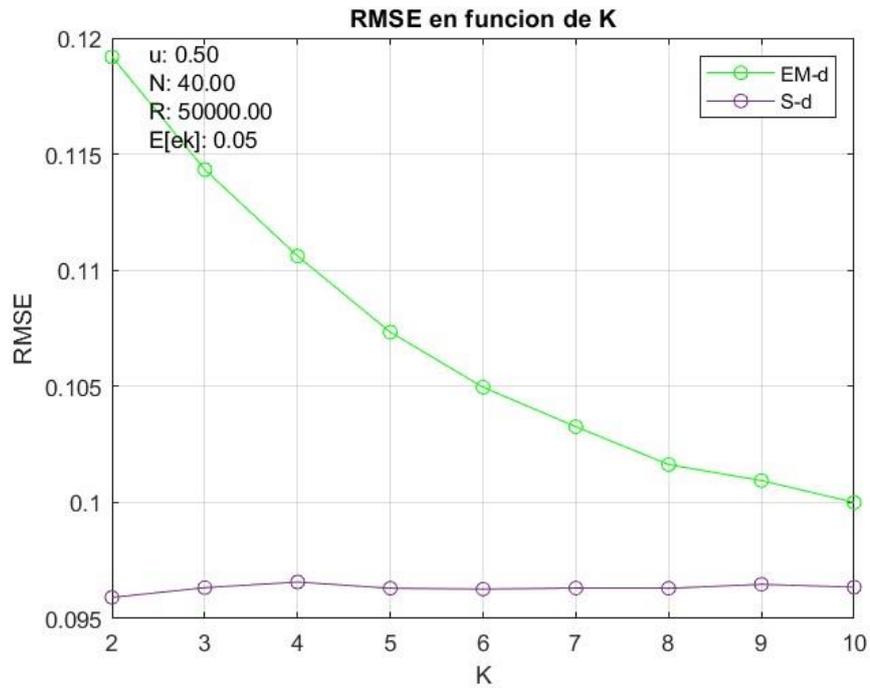


Figura 7: RMSE en función del número de sensores

Vemos que a medida que aumenta el número de sensores, las prestaciones de las reglas de fusión mejoran. La ganancia obtenida al aumentar el número de sensores se hace menor a medida que el número de sensores crece.

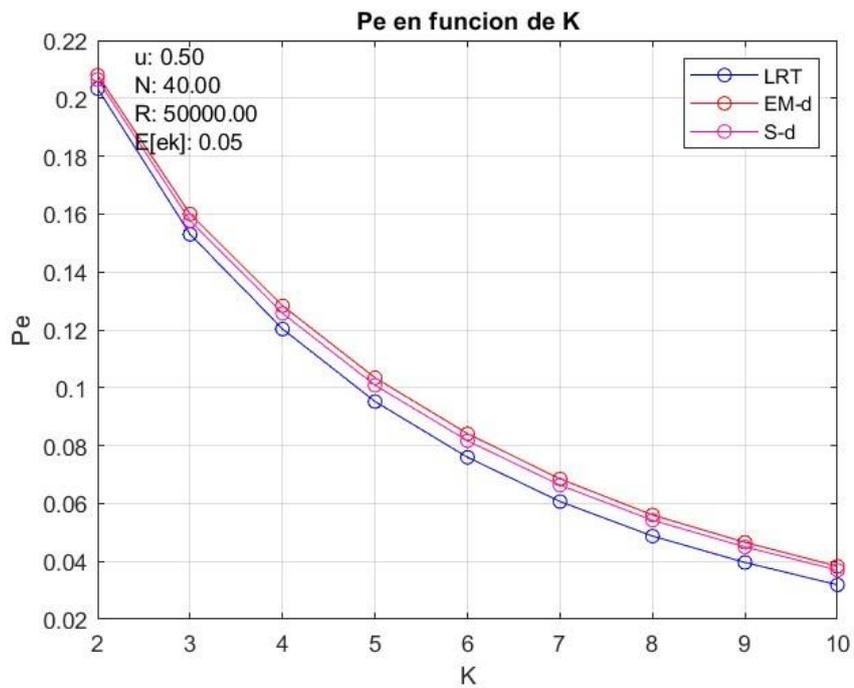


Figura 8: Probabilidad de error en función del número de sensores

A continuación, se presentan las curvas de RMSE y  $P_e$  en función la probabilidad de error medio de los canal BSC. Como es de esperar a medida que el error medio aumenta las prestaciones de las reglas de fusión empeoran y el estimador EM también se verá perjudicado por este error.

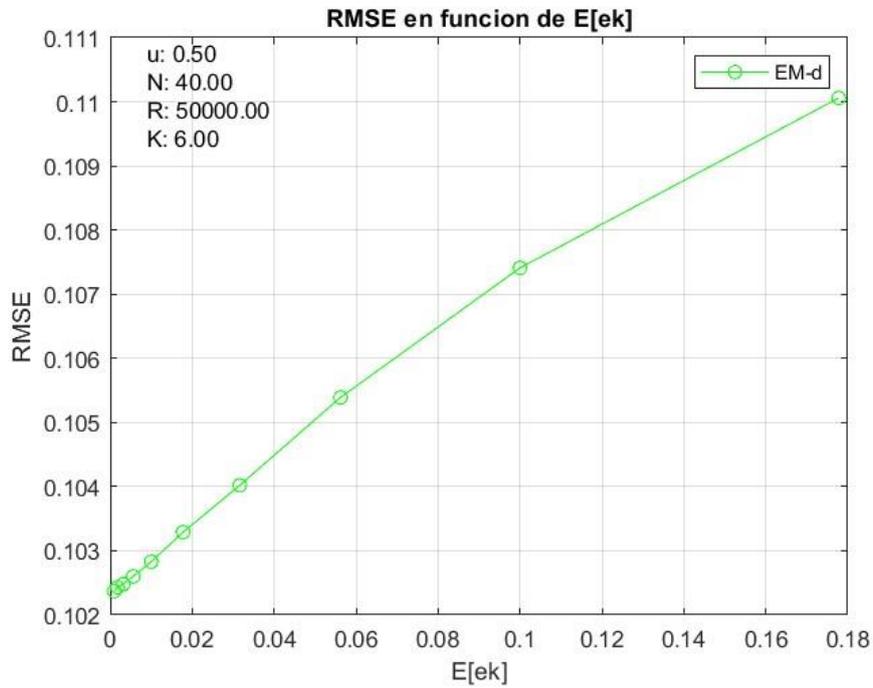


Figura 9: RMSE en función de la probabilidad de error media de los canales BSC

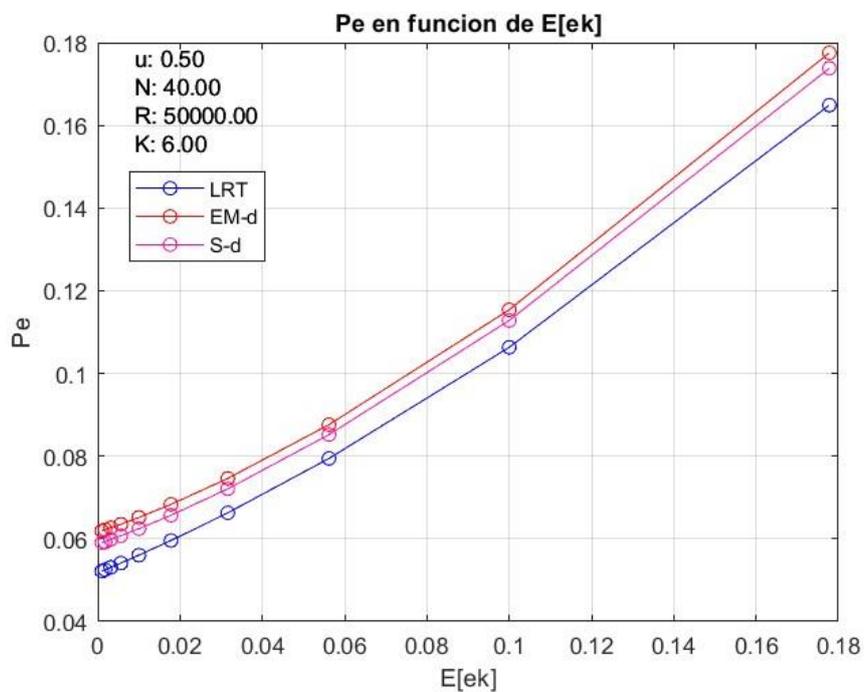


Figura 10: Probabilidad de error en función de la probabilidad de error media de los canales BSC

## 4.2. Prestaciones del estimador de $\mathbf{d}$ y $u$ , y de la regla de decisión

Se compararán las prestaciones de la regla de fusión en el caso en el que se estima  $\mathbf{d}$ , y en el caso en el que se estiman  $\mathbf{d}$  y  $u$ , con las del detector Clairvoyant y los supervisados. La métrica de evaluación del detector será la probabilidad de error y las prestaciones del estimador se evaluarán utilizando el RMSE con la formula (21) para  $\mathbf{d}$  y para la  $u$ :

$$RMSE = \sqrt{\frac{1}{RK} \sum_{r=1}^R |\hat{u}^{(r)} - u^{(r)}|^2}$$

La curva de RMSE del estimador de  $\mathbf{d}$  cuando solo se estima este parámetro se denota como EM-d, mientras que la curva de RMSE del estimador de  $\mathbf{d}$  cuando se estiman  $\mathbf{d}$  y  $u$  será denotada como EM-d,u [d], la curva de RMSE del estimador de  $u$  cuando se estiman  $\mathbf{d}$  y  $u$  será denotada como EM-d,u [u]. Para el estimador supervisado se denotaran como S-d y S-u.

A continuación, se presentan las curvas de RMSE y  $P_e$  en función del número de periodos de sensado  $N$ . Dado que  $N$  solo afecta las estimaciones, observaremos que a medida que  $N$  aumenta, las prestaciones de los detectores que estiman mejoran, mientras que el detector Clairvoyant es independiente de  $N$ .

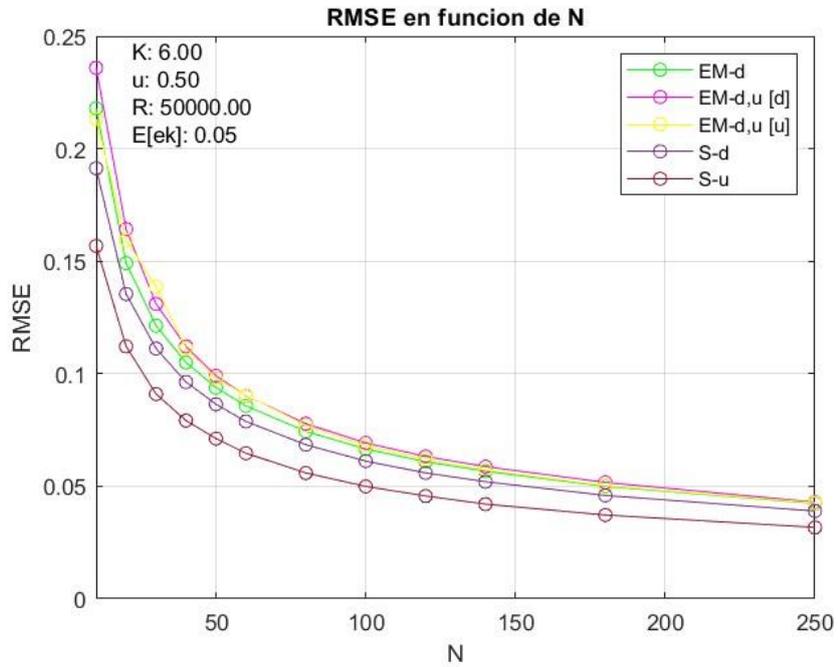


Figura 11: RMSE en función del número de periodos de sensado

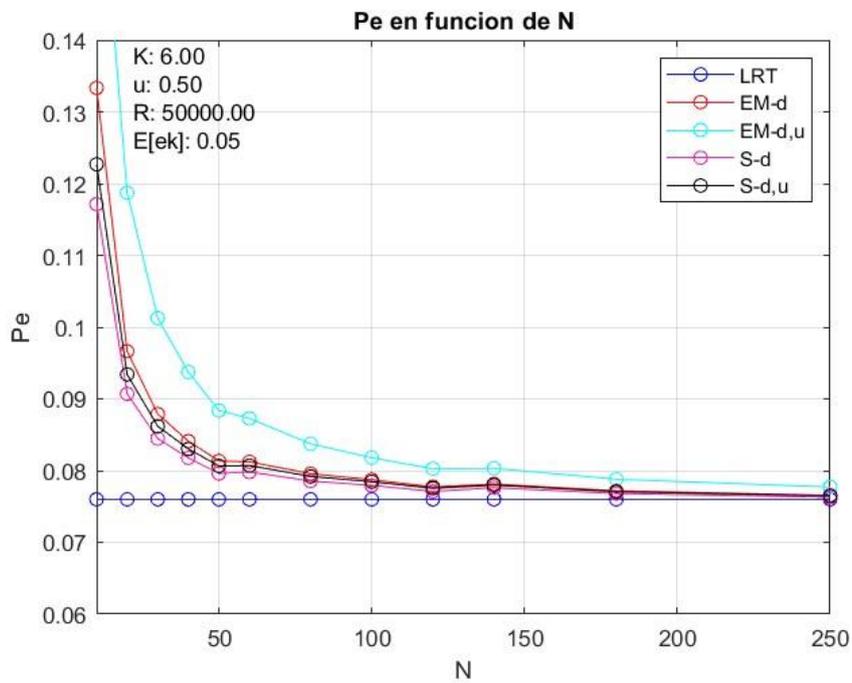


Figura 12: Probabilidad de error en función del número de periodos de sensado

Se presentan las curvas de RMSE y  $P_e$  en función de la probabilidad a priori  $u$ .

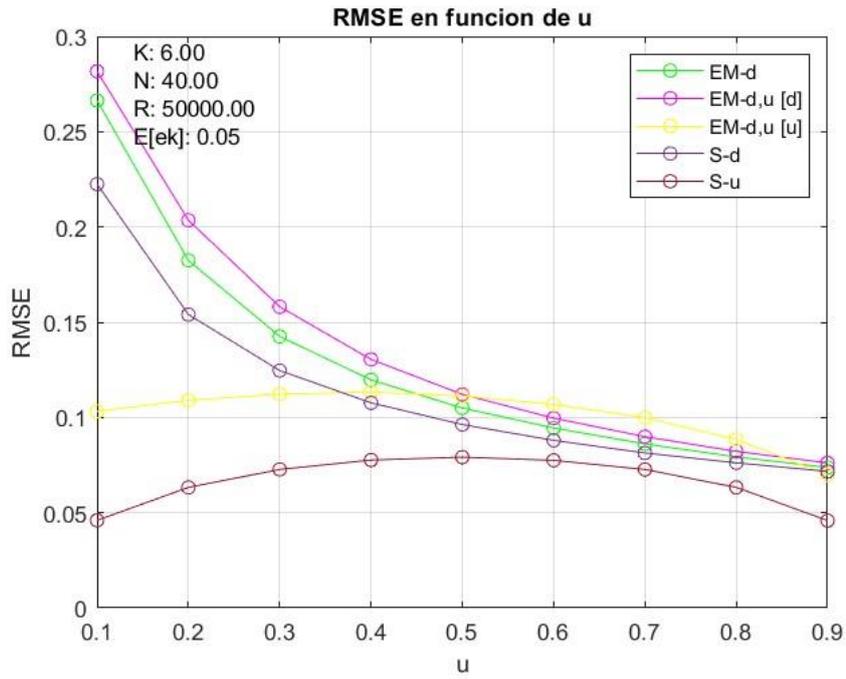


Figura 13: RMSE en función de la probabilidad a priori

Todos los detectores tienen sus valores más desfavorables en torno a  $u = 0.5$ .

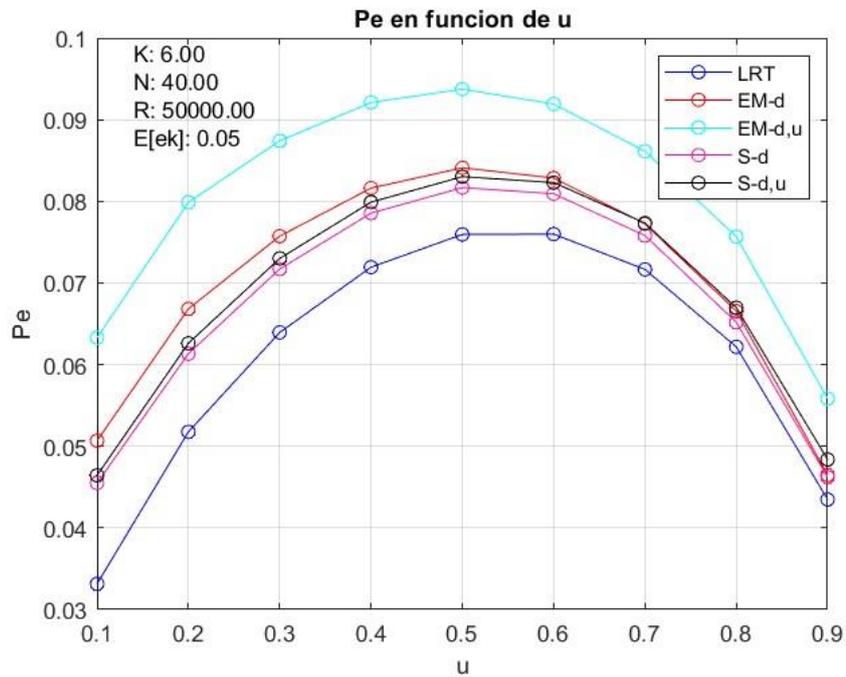


Figura 14: Probabilidad de error en función de la probabilidad a priori

Se presentan las curvas de RMSE y  $P_e$  en función del número de sensores. Vemos que a medida que aumenta el número de sensores, el FC tiene más información por lo tanto las prestaciones de las reglas de fusión mejoran.

Las RMSEs muestran como mejoran las prestaciones de los estimadores según  $K$  aumente. En los casos del estimador supervisado las RMSEs se mantienen constantes, ya que el aumento al número de sensores es proporcional al aumento de variables que hay que estimar.

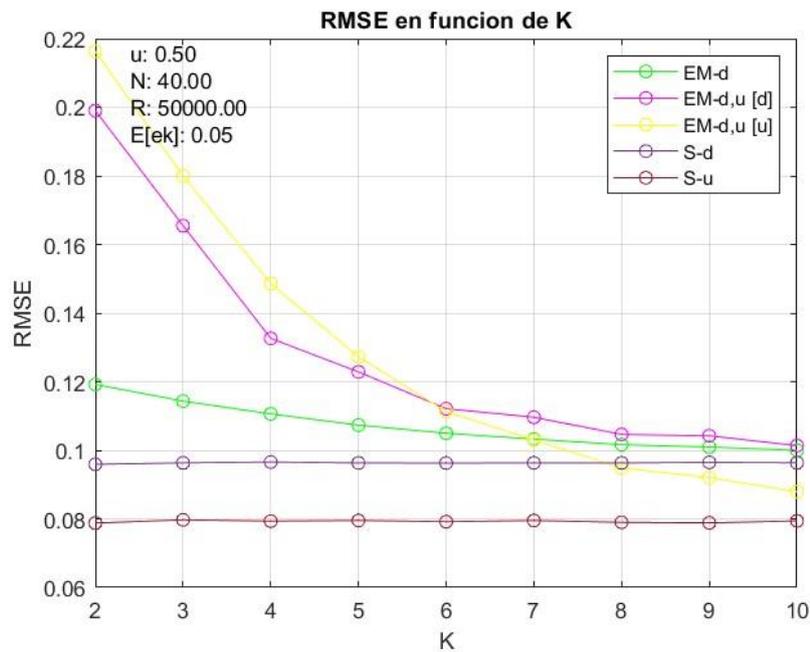


Figura 15: RMSE en función del número de sensores

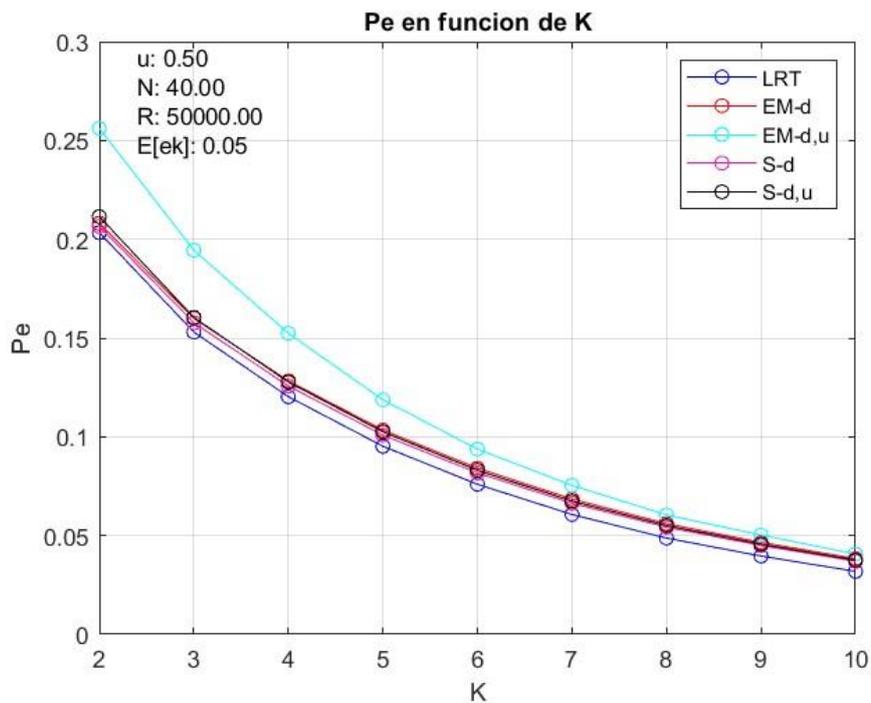


Figura 16: Probabilidad de error en función del número de sensores

Se presentan las curvas de RMSE y  $P_e$  en función de la probabilidad de error media de los canales BSC. Tanto la estimación como la detección empeoran.

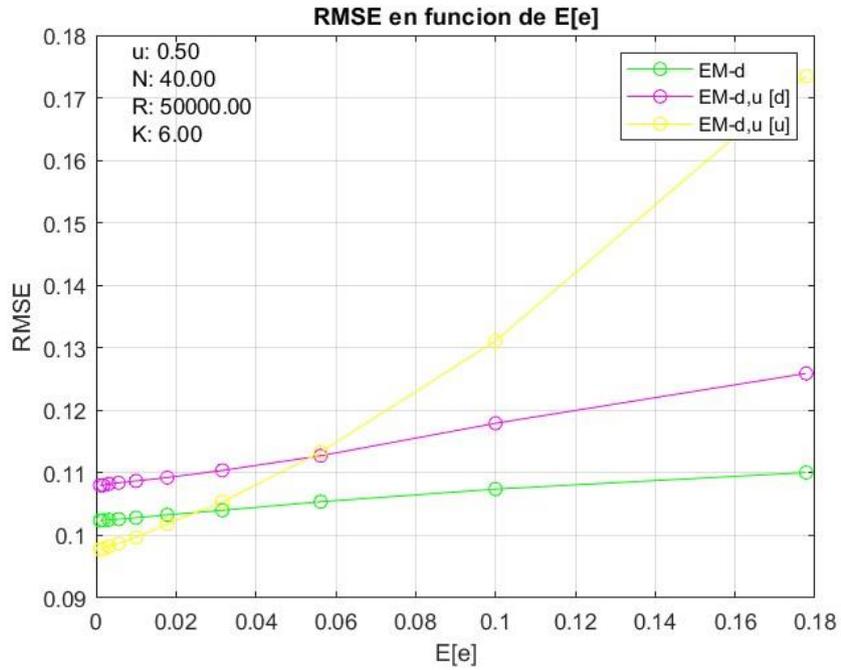


Figura 17: RMSE en función de la probabilidad de error media de los canales BSC

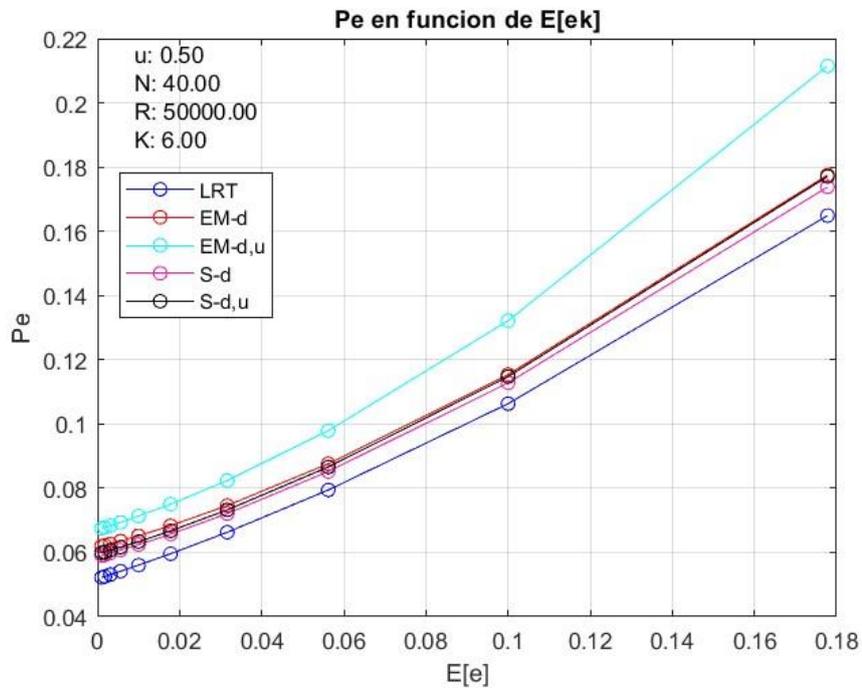


Figura 18: Probabilidad de error en función de la probabilidad de error media de los canales BSC

### 4.3. Comparaciones con la regla de voto Mayoritario

A continuación, se comparan las prestaciones de cuatro detectores: el Clairvoyant (LRT), el Supervisado (S-d), el detector estima con el algoritmo EM (EM-d) y el detector de la Regla de Voto Mayoritario (MVR). Todos ellos para el caso en el que  $d$  es desconocida, menos el Clairvoyant en el que  $d$  se supone conocida.

En la primera gráfica, se muestra las Probabilidades de Error en función del número de periodos de sensado. El detector que usa la regla de voto mayoritario tiene prestaciones mucho peores a los demás detectores, y se mantiene constante ya que como no estima nada es independiente al número de periodos de sensado. Su valor de probabilidad de error es 0.35. Como este valor es alto comparado con el resto de detectores, probamos con todos los valores de C posibles para obtener el valor más bajo, que se obtiene con C=2.

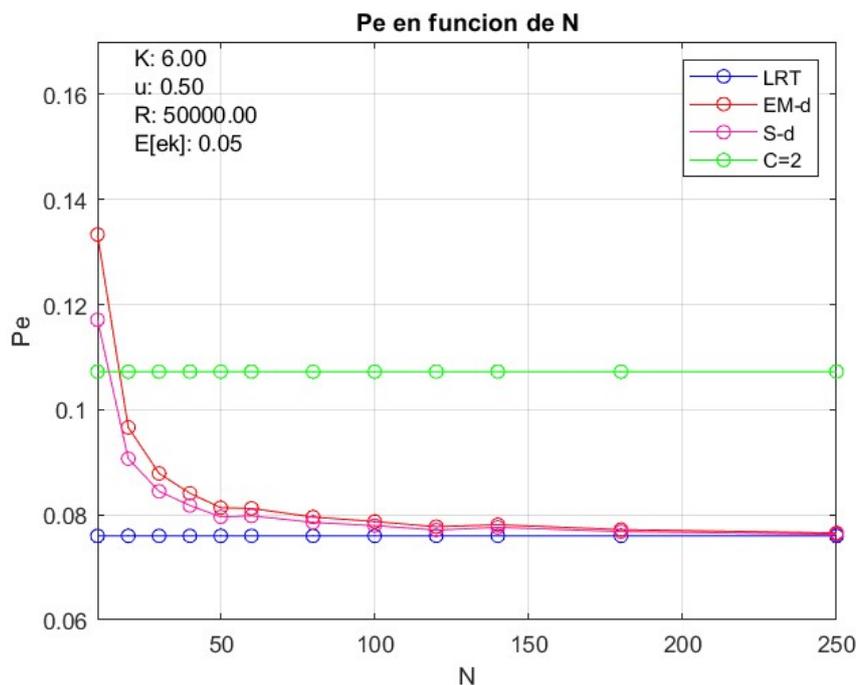


Figura 19: Probabilidad de error en función del número de periodos de sensado

A continuación se muestra la probabilidad de error para los 4 detectores en función de la probabilidad a priori. El detector MVR muestra que empeora de manera lineal ascendente. En cualquier caso, sus prestaciones son las peores de todos los detectores.

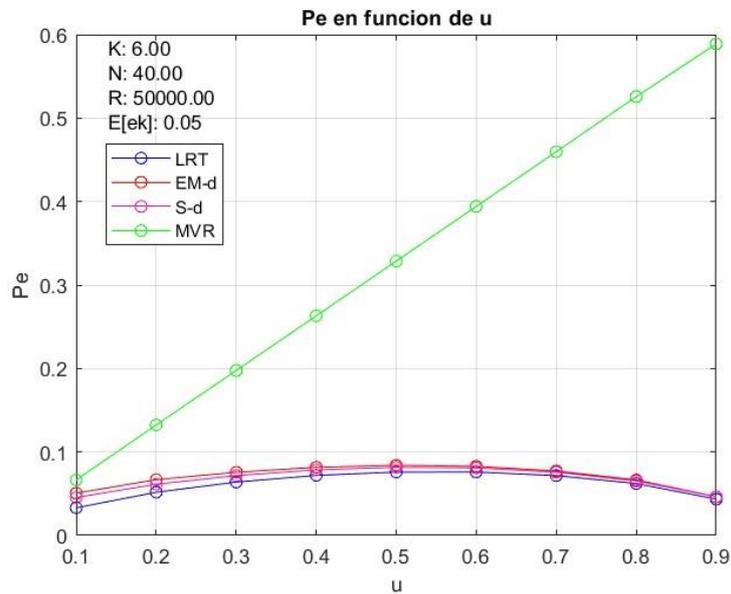


Figura 20: Probabilidad e error en función de la probabilidad a priori

Se muestran las prestaciones de los cuatro detectores en relación al número de sensores. La curva que representa la probabilidad de error del detector de regla de voto mayoritario exhibe un patrón de zigzag, con las probabilidades de error menor cuando el número de sensores es impar. Esto se debe a que, la regla es asimétrica, ya que  $C=(K+1)/2$  por lo tanto con un número de sensores impar la regla funciona mejor. Además, se observa que a medida que aumenta el número de sensores, las prestaciones de este detector tienden a mejorar ligeramente.

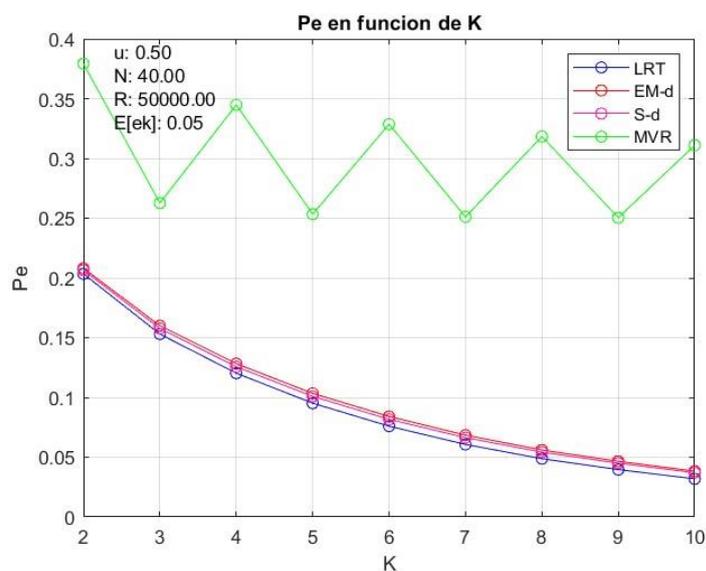


Figura 21: Probabilidad de error en función del número de sensores

A continuación se presentan las curvas de probabilidad de error en función del error medio del canal, lo que resulta en un deterioro de las prestaciones para los cuatro detectores.

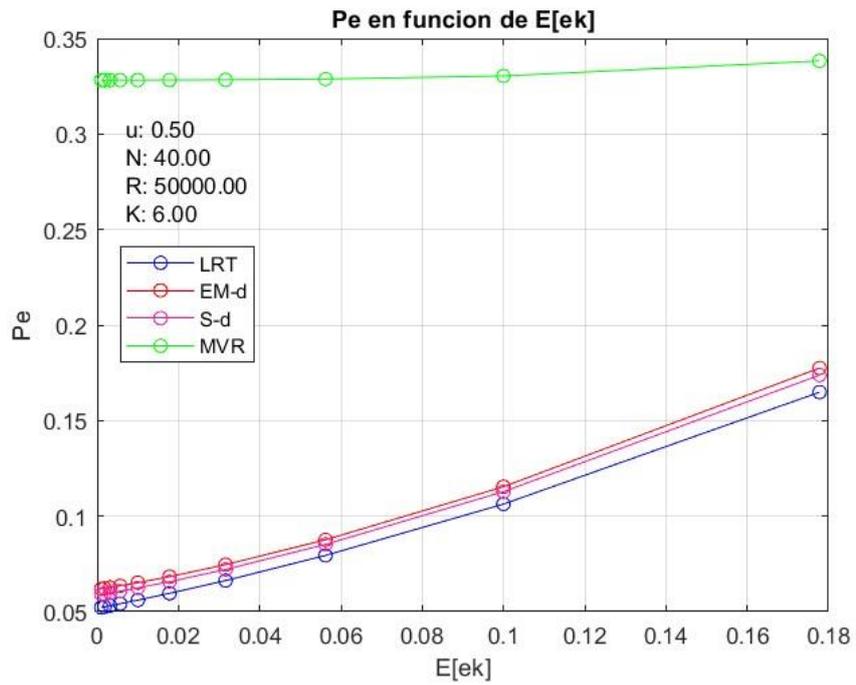


Figura 22: Probabilidad de error en función del error medio del canal

## 5. Conclusiones

En este trabajo, se ha presentado un algoritmo innovador diseñado para la estimación ciega de las probabilidades de detección local en sistemas DD-WSN. Basándose en la premisa de que estas probabilidades permanecen constantes durante un número de periodos de sensado, el algoritmo permite al FC estimar estas probabilidades a partir de señales binarias ruidosas obtenidas de canales BSC. El algoritmo propuesto para esta tarea es el algoritmo EM, cuyos resultados han sido utilizados para derivar la regla de decisión del FC, fundamentada en el criterio de mínima probabilidad de error.

Además, se ha ampliado el alcance de este trabajo al abordar la situación en la que la probabilidad a priori también es desconocida, presentando una solución eficaz utilizando el mismo algoritmo. Se han generado múltiples gráficas que muestran los resultados de la probabilidad de error de diversos detectores en distintos escenarios, permitiendo así una comparación exhaustiva de sus desempeños. Asimismo, se han incluido gráficas que muestran el RMSE para evaluar y comparar el rendimiento de los estimadores propuestos.

Este estudio resalta la utilidad y la eficacia del algoritmo propuesto, así como su capacidad para abordar problemas complejos en el contexto de sistemas de fusión de decisiones sensoriales.

## 6. Apéndices

### 6.1. Apéndice A: Resolución Paso E para la estimación de $\mathbf{d}$ y $u$

$$\begin{aligned}
 Q^{(i)}(\mathbf{d}, u) &= \mathbb{E}[\log L_c(\mathbf{d}, u) | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}] \\
 Q^{(i)}(\mathbf{d}, u) &= \mathbb{E}\left[\sum_{n=1}^N \log [(1-u)p(\mathbf{x}_n|H_0)]^{1-z_n} [u p(\mathbf{x}_n|H_1, \mathbf{d})]^{z_n} | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}]\right] \\
 &= \mathbb{E}\left[\sum_{n=1}^N (1-z_n) \log[(1-u)p(\mathbf{x}_n|H_0)] + z_n \log[u p(\mathbf{x}_n|H_1, \mathbf{d})] | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}]\right]
 \end{aligned}$$

Y debido a la linealidad del operador  $\mathbb{E}$ :

$$\begin{aligned}
 Q^{(i)}(\mathbf{d}, u) &= \sum_{n=1}^N \mathbb{E}[(1-z_n) | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}] \log[(1-u)p(\mathbf{x}_n|H_0)] \\
 &\quad + \mathbb{E}[z_n | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}] \log[u p(\mathbf{x}_n|H_1, \mathbf{d})] = \\
 &\quad \sum_{n=1}^N (1 - \mathbb{E}[z_n | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}]) \log[(1-u)p(\mathbf{x}_n|H_0)] \\
 &\quad + \mathbb{E}[z_n | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}] \log[u p(\mathbf{x}_n|H_1, \mathbf{d})]
 \end{aligned}$$

donde

$$\begin{aligned}
 \mathbb{E}[z_n | \mathbf{X}, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}] &= 0 * P(z_n = 0 | \mathbf{x}_n, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}) + 1 * P(z_n = 1 | \mathbf{x}_n, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}) \\
 &= P(z_n = 1 | \mathbf{x}_n, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)})
 \end{aligned}$$

Aplicando el Teorema de Bayes;

$$P(z_n = 1 | \mathbf{x}_n, \hat{\mathbf{d}}^{(i-1)}, \hat{u}^{(i-1)}) = \frac{\hat{u}^{(i-1)} p(\mathbf{x}_n | H_1, \hat{\mathbf{d}}^{(i-1)})}{(1 - \hat{u}^{(i-1)}) p(\mathbf{x}_n | H_0) + \hat{u}^{(i-1)} p(\mathbf{x}_n | H_1, \hat{\mathbf{d}}^{(i-1)})}$$

Para simplificar la notación, vamos a denominar este cociente como  $t_n^{(i)}$ , que se puede interpretar como una estima de  $z_n$  después de observar  $\mathbf{x}_n$  cuando las probabilidades de detección son  $\hat{\mathbf{d}}^{(i-1)}$  y la probabilidad a priori es  $\hat{u}^{(i-1)}$ .

Por lo tanto tendríamos:

$$Q^{(i)}(\mathbf{d}, u) = \sum_{n=1}^N (1 - t_n^{(i)}) \log[(1-u)p(\mathbf{x}_n|H_0)] + t_n^{(i)} \log[u p(\mathbf{x}_n|H_1, \mathbf{d})]$$

## 6.2 Apéndice B: Resolución Paso M para la estimación de $\mathbf{d}$ y $u$

$$\hat{u}^{(i)} = \operatorname{argmax}_{0 < u < 1} N_0^{(i)} \log(1 - u) + N_1^{(i)} \log u$$

Derivamos la función a optimizar y la igualamos a cero

$$N_0^{(i)}(-1) \frac{1}{1 - u} + N_1^{(i)} \frac{1}{u} = 0$$

$$\frac{-N_0^{(i)} u + N_1^{(i)}(1 - u)}{u(1 - u)} = 0$$

$$-N_0^{(i)} u + N_1^{(i)}(1 - u) = 0$$

$$-u(N_0^{(i)} + N_1^{(i)}) = -N_1^{(i)}$$

$$\hat{u}^{(i)} = \frac{N_1^{(i)}}{N}$$

## 7. Referencias

- [1] P. K. Varshney, *Distributed Detection and Data Fusion*, Springer New York, 1997.
- [2] B. Chen and P.K. Willett, “*On the optimality of the likelihood-ratio test for local sensor decision rules in the presence of nonideal channels*,” *IEEE Transactions on Information Theory*, vol. 51, no. 2, pp. 693–699, 2005.
- [3] A. Kashyap, “*Comments on ”on the optimality of the likelihood-ratio test for local sensor decision rules in the presence of nonideal channels”*,” *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1274–1275, 2006.
- [4] H. Chen, B. Chen, and P. K. Varshney, “*Further results on the optimality of the likelihood-ratio test for local sensor decision rules in the presence of nonideal channels*,” *IEEE Transactions on Information Theory*, vol. 55, no. 2, pp. 828–832, 2009.
- [5] Z. Chair and P.K. Varshney, “*Optimal data fusion in multiple sensor detection systems*,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-22, no. 1, pp. 98–101, 1986.
- [6] S. C. A. Thomopoulos, R. Viswanathan, and D. C. Bougoulas, “*Optimal decision fusion in multiple sensor systems*,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-23, no. 5, pp. 644–653, 1987.
- [7] B. Chen, R. Jiang, T. Kasetkasem, and P.K. Varshney, “*Channel aware decision fusion in wireless sensor networks*,” *IEEE Transactions on Signal Processing*, vol. 52, no. 12, pp. 3454–3458, 2004.
- [8] R. Jiang and B. Chen, “*Fusion of censored decisions in wireless sensor networks*,” *IEEE Transactions on Wireless Communications*, vol. 4, no. 6, pp. 2668–2673, 2005.
- [9] R. Niu, B. Chen, and P.K. Varshney, “*Fusion of decisions transmitted over rayleigh fading channels in wireless sensor networks*,” *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 1018–1027, 2006.
- [10] K. B. Letaief and W. Zhang, “*Cooperative communications for cognitive radio networks*,” *Proceedings of the IEEE*, vol. 97, no. 5, pp. 878–893, 2009.
- [11] W. Zhang, R. K. Mallik, and K. B. Letaief, “*Optimization of cooperative spectrum sensing with energy detection in cognitive radio networks*,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 12, pp. 5761–5766, 2009.
- [12] J. Y. Wu, C. W. Wu, T. Y. Wang, and T. S. Lee, “*Channel-aware decision fusion with unknown local sensor detection probability*,” *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1457–1463, 2010.
- [13] D. Ciuonzo and P. Salvo Rossi, “*Decision fusion with unknown sensor detection probability*,” *IEEE Signal Processing Letters*, vol. 21, no. 2, pp. 208–212, 2014.
- [14] S. Chaudhari, J. Lunden, V. Koivunen, and H. V. Poor, “*Cooperative sensing with imperfect reporting channels: Hard decisions or soft decisions?*,” *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 18–28, 2012.

- [15] A. P. Dawid and A. M. Skene, “*Maximum likelihood estimation of observed error-rates using the em algorithm*,” *Applied Statistics*, vol. 28, no. 1, pp. 20–28, 1979.
- [16] B. Chen, L. Tong, and P. K. Varshney, “*Channel-aware distributed detection in wireless sensor networks*,” *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 16–26, 2006.
- [17] A. Lei and R. Schober, “*Coherent max-log decision fusion in wireless sensor networks*,” *IEEE Transactions on Communications*, vol. 58, no. 5, pp. 1327–1332, 2010.
- [18] S.C.A. Thomopoulos and L. Zhang, “*Distributed decision fusion in the presence of networking delays and channel errors*,” *Information Sciences*, vol. 66, pp. 91–118, 1992.
- [19] Domenico Ciuonzo, Gianmarco Romano, and Pierluigi Salvo Rossi, “*Channel-aware decision fusion in distributed mimo wireless sensor networks: Decode-and-fuse vs. decode-then-fuse*,” *IEEE Transactions on Wireless Communications*, vol. 11, no. 8, pp. 2976–2985, 2012.
- [20] N. R. Banavathu and M. Z. A. Ali, “*Optimization of n-out-of-k rule for heterogeneous cognitive radio networks*,” *IEEE Signal Processing Letters*, vol. 26, no. 3, pp. 445–449, 2019.
- [21] Yongsheng Yan, Haiyan Wang, Xiaohong Shen, and Xionghu Zhong, “*Decision fusion with channel errors in distributed decode-then-fuse sensor networks*,” *Sensors*, vol. 15, no. 8, pp. 19157–19180, 2015.
- [22] Q. Cheng, B. Chen, and P. K. Varshney, “*Detection performance limits for distributed sensor networks in the presence of nonideal channels*,” *IEEE Transactions on Wireless Communications*, vol. 5, no. 11, pp. 3034–3038, 2006.
- [23] S. M. Kay, *Fundamentals of Statistical Signal Processing. Detection Theory*, Prentice Hall. Signal Processing Series, 1998.
- [24] H. V. Poor, *An introduction to signal detection and estimation*, Springer-Verlag, 1994.
- [25] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
- [26] G. J. McLachlan and T. Krishnan, *The EM algorithm and extensions*, John Wiley and Sons, 2008.
- [27] A. P. Dempster, N. M. Laird, and D. B. Rubin, “*Maximum likelihood for incomplete data via the em algorithm*,” *Journal of the Royal Statistics Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.
- [28] F. R. Mohammad, D. Ciuonzo, and Z. A. K. Mohammed, “*Mean-based blind hard decision fusion rules*,” *IEEE Signal Processing Letters*, vol. 25, no. 5, pp. 630–634, 2018.