



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

Curso Académico:

AGRADECIMIENTOS

A mi familia
por la paciencia

RESUMEN

Uno de los aspectos esenciales en la robótica móvil es la obtención de la localización del robot en el espacio de movimiento puesto que esto es necesario para generar los movimientos deseados del robot. Sin embargo, obtener la localización precisa no es una tarea fácil ni trivial puesto que suelen influir varios factores: ruido o mal precisión en los sensores, deslizamientos de las ruedas, etc. Para evitar estos problemas se puede desarrollar sistemas que mejoran la estimación de las variables de estado de los sistemas mediante una fusión de la mayor cantidad posible de fuentes de información (sensores).

En el presente Trabajo Fin de Grado (TFG) se va a trabajar con el desarrollo de diferentes sistemas de fusión de datos. En concreto se van a desarrollar diversos Filtros de Kalman (KF), que es una de las herramientas más potentes para la resolución de estos problemas. En el TFG se abordará el diseño de KF para dos sistemas reales: un robot móvil y un sistema para la determinación de la posición de segmentos corporales de personas.

RESUM

Un dels aspectes essencials en la robòtica mòbil és l'obtenció de la localització del robot en l'espai de moviment ja que açò és necessari per a generar els moviments desitjats del robot. No obstant això, obtenir la localització precisa no és una tasca fàcil ni trivial ja que solen influir diversos factors: soroll o mal precisió en els sensors, lliscaments de les rodes, etc. Per a evitar estos problemes es pot desenvolupar sistemes que milloren l'estimació de les variables d'estat dels sistemes per mitjà d'una fusió de la major quantitat possible de fonts d'informació (sensors).

En el present Treball Fi de Grau (TFG) se'n va a treballar amb el desenvolupament de diferents sistemes de fusió de dades. En concret es van a desenvolupar diversos Filtres de Kalman (KF), que és una de les eines més potents per a la resolució d'estos problemes. En el TFG s'abordarà el disseny de KF per a dos sistemes reals: un robot mòbil i un sistema per a la determinació de la posició de segments corporals de persones.

Abstract

One of the essential aspects in mobile robotics is to obtain the location of the robot in the movement space, as this is necessary to generate the desired movements of the robot. However, obtaining the exact location is not an easy or trivial task since many factors usually influence: noise or poor precision in the sensors, slides of the wheels, etc. To avoid these problems, systems can be developed that improve the estimation of the state. Variables of the systems by a fusion of the greatest possible number of sources of information (sensors).

In this Work End of Degree (TFG) will work with the development of different systems of data fusion. Specifically, several Kalman filters (KF) will be developed, which is one of the most powerful tools to solve these problems. In the TFG will be addressed the design of KF for two real systems: a mobile robot and a system to determine the position of the body segments of people.

ÍNDICE

DOCUMENTOS CONTENIDOS EN EL TFG:

- Memoria
- Presupuesto
- Anejos

Índice de la memoria

1. Introducción	13
2. Bases teóricas	15
2.1 Introducción y notación	15
2.2. Los observadores de estado.....	17
2.2.1. Observadores de estado en sistemas lineales	17
2.2.2. Observadores de estado en sistemas no lineales	17
2.3. Los filtros de estimación.....	18
2.3.1. Filtros para la estimación de sistemas lineales	18
2.3.2. Filtros para la estimación de sistemas no lineales	22
2.4. Conclusiones.....	24
3. Objetivos	27
4. Metodología	29
4.1. Introducción	29
4.2. Equipamiento empleado.....	29
4.2.1. LEGO Mindstorm NXT.....	29
4.2.2. Dispositivo de tecnología vestible (wearable)	31
4.3. Planteamiento de las problemáticas.....	35
4.3.1. Problemática con las estimaciones simples	35
4.3.2. Problemática con la toma de datos de los sensores del brazalete	37
4.3. Modelización del KF en el LEGO Mindstorm NXT	38
4.3.1. Modelización para la inicialización de P, Q y R	38
4.3.2. Modelización para la mejora de las medidas de los encóders.....	39
4.3.3. Modelización para la estimación de las velocidades lineal y angular	40
4.4. Modelización del KF en el dispositivo de tecnología vestible	41
4.5. Implementación de los códigos de programación	42
4.5.1. Implementación en el LEGO.....	42

4.5.1. Implementación del filtro de Kalman en el brazalete	45
4.6. Experimentación con LEGO Mindstorm NXT	46
4.6.1. Estudio de la inicialización de las matrices características del KF.....	46
4.6.2. El filtro de Kalman para mejorar la medida de los encóders	47
4.6.3. Estimación de la velocidad lineal y angular del LEGO Mindstorm NXT.....	47
4.7. Experimentación con el brazalete.....	48
5. Resultados	51
5.1. Resultados con el LEGO Mindstorm NXT	51
5.1.1. Estudio de la inicialización de las matrices características del KF.....	51
5.1.2. El filtro de Kalman para mejorar la medida de los encóders	52
5.1.3. Estimación de la velocidad lineal y angular del LEGO Mindstorm NXT.....	56
5.2. Resultados con el brazalete	58
Referencias.....	67

Índice del presupuesto

Unidades de obra	71
Lista de materiales	71
Amortización	71
Gastos de material	72
Gastos de personal.....	72
Rendimientos y mediciones	72
Cuadro de precios descompuestos	72
Presupuesto de Ejecución Material (PEM).....	73

Índice de los anejos

Anejo I	77
Anejo II	79

MEMORIA

1. Introducción

A lo largo de la historia se han ido desarrollando diversas líneas de investigación para el control de procesos. Dichas líneas de investigación han supuesto el perfeccionamiento de los controladores automáticos, los cuales son cada día más demandados en los diferentes campos de la ingeniería debido al rápido desarrollo tecnológico que está experimentando la sociedad actual.

Para poder controlar cualquier proceso de la naturaleza, se requiere acceder a la mayor cantidad posible de información del entorno (variables a controlar) mediante distintas fuentes de información (sensores). Además, si dicho proceso se debe controlar de manera automática, la complejidad aumenta, entre otros factores, debido a las limitaciones tecnológicas y económicas. Por ejemplo, si se requiere de buena precisión en los sensores, el coste de la instrumentación de control aumenta, o bien, si aparecen ruidos en los sensores, las variables a controlar no se pueden medir con la suficiente precisión.

Todo esto ha conllevado a una búsqueda continua de sistemas que estimaran los valores de las variables a controlar con la menor cantidad posible de sensores. De esta forma, si no se puede acceder a la totalidad de la información de las variables a controlar, se pueden calcular sus valores y trabajar con ellos. En ocasiones, las variables a controlar se pueden estimar mediante aproximaciones simples como es el caso de la velocidad, que es la derivada de la posición respecto del tiempo, pero no siempre se obtienen buenos resultados. Sin embargo, existen unos sistemas conocidos como estimadores de estado que mejoran el comportamiento de las estimaciones simples basándose en teorías probabilísticas y estadísticas. Es por ello que dichos estimadores se pueden emplear en cualquier campo de la ingeniería, pero también en economía, ya que su función es predecir una variable matemática.

La industria 4.0 o el internet de las cosas (IoT) son algunas de las tecnologías más conocidas que están actualmente en vías de desarrollo. Sin embargo, no son las únicas que requieren de controladores automáticos y de estimadores de estado. En el caso de la ingeniería, uno de los campos en el que se requiere de estimadores para el control automático de sus variables, es el campo de la robótica, donde se está investigando con nuevos procesos. Ya no solo se busca medir variables de procesos de control tradicionales, como por ejemplo la posición del motor para el control de la trayectoria de un robot móvil, sino también medir variables de procesos no controlables para darles nuevas aplicaciones, este es el caso del movimiento del brazo de una persona para aplicaciones biotecnológicas.

Así pues, en el presente Trabajo de Final de Grado (TFG) se afrontan dos problemáticas cuyas soluciones requieren de la aplicación de estimadores de estado. La primera de ellas surge cuando una de las ruedas de un robot o vehículo móvil derrapa. En este caso, el controlador del sistema estima la posición del robot asumiendo que la velocidad del mismo ha variado, pero esto, en realidad, no ha sucedido. En cambio, mediante la estimación de la velocidad del robot a partir de la información procedente de varios sensores, este problema se puede solventar. La segunda de las problemáticas surge en la investigación con nuevas aplicaciones biotecnológicas. En este caso, se requiere fusionar los datos procedentes de una cámara de

visión artificial y de un giróscopo para determinar la posición de un segmento corporal en todo momento. Además, es necesario dotar de mayor importancia a uno u otro sensor dependiendo del contexto, es decir, hay ocasiones en las que la cámara no detecta la posición del segmento y hay que llevar a cabo la estimación de la posición únicamente con los datos del giróscopo, en cambio, en otras situaciones, los dos sensores trabajarán adecuadamente y habrá que fusionar los dos datos. Todo esto requiere la búsqueda de un estimador de estado adecuado.

2. Bases teóricas

En el presente capítulo se detalla toda la teoría de control de procesos y estimadores de estado que es requerida para encontrar un estimador de estado que permita el desarrollo del trabajo experimental llevado a cabo en este TFG. Principalmente se describen: conocimientos básicos acerca de los diagramas de control de procesos además de una pequeña notación que será común en todos los capítulos y se introducen los estimadores de estado más importantes tanto para sistemas lineales como para sistemas no lineales.

2.1 Introducción y notación

Una primera aproximación a los estimadores de estado se puede hacer mediante su lazo o bucle de control. A continuación, se analiza un lazo de control cerrado simple para después incorporarle los estimadores de estado y así poder explicar los distintos sistemas donde se aplican dichos estimadores.

Para el control de un proceso de la naturaleza que sea dinámico y observable, cuyo estado x_k sea perturbado por el ruido w_k , se debe observar su salida y_k mediante un sensor cuya medida z_k estará afectada por un ruido n_k . Para completar el lazo de control (figura 1) se debe incorporar al mismo un controlador que a partir del error e_k , calculado como la diferencia entre la referencia y la medida del sensor ($e_k=r_k-z_k$), le proporcione una entrada u_k al proceso que permita dirigir el comportamiento del mismo.

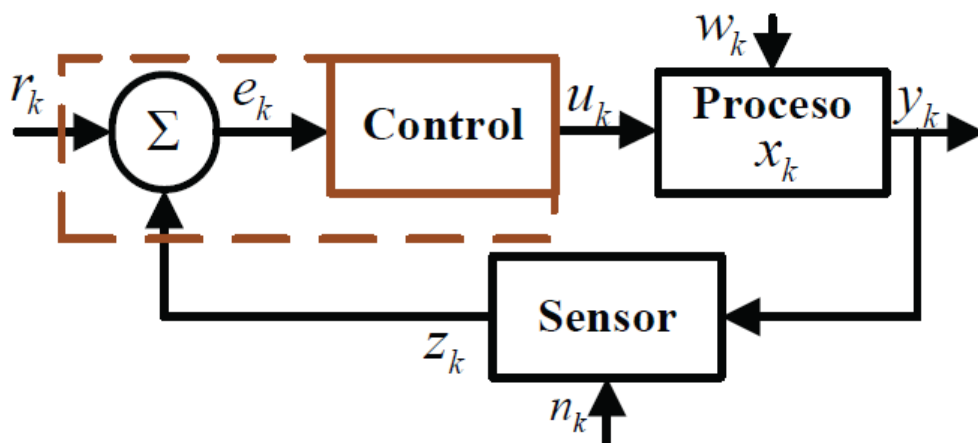


Figura 1. Lazo de control cerrado de un proceso [1]

Los estimadores de estado se añaden al lazo de control (figura 2) en aquellas situaciones donde no se puede acceder a la medición de todos los estados que influyen sobre un proceso. Así pues, la función del observador es estimar el valor de todos los estados del proceso a partir, únicamente, de las medidas accesibles (z_k) mediante los sensores de los que se disponga y de las variables de entrada al sistema u_k . Además, se considera que la estimación es lo suficientemente buena cuando el error entre el valor real del estado x_k y el valor estimado del mismo \hat{x}_k tiende a cero, es decir, $x_k - \hat{x}_k \cong 0$. Así pues, el estado estimado se introduce en el controlador para el cálculo del error entre la referencia y el estado estimado y, a su vez, este

error se emplea en las ecuaciones de control pertinentes para dirigir el comportamiento del sistema.

Los estimadores de estado pueden ser aplicados a dos tipos de sistemas: los sistemas lineales y los sistemas no lineales. Para simplificar la notación de los modelos matemáticos de los procesos a implementar, se definen como sistemas lineales aquellos cuyo algoritmo es de la forma:

$$x_k = A_{k-1} * x_{k-1} + B_{k-1} * u_{k-1} + w_{k-1} \quad (2.1)$$

$$z_k = H_k * x_k + n_k$$

Donde $u_k \in \mathbb{R}^u$ es el vector de entrada, $z_k \in \mathbb{R}^m$ es el vector de medición y $x_k \in \mathbb{R}^n$ es el vector de estado. En cuanto a las matrices A_k , B_k y H_k , que son las matrices de estados, de entradas y de medición respectivamente, en el caso de los sistemas lineales, sus entradas serán constantes. Además, estas matrices se diseñan para que el valor estimado converja al valor real del estado con la mayor celeridad y precisión posible.

Por otro lado, se considera un sistema como no lineal si su algoritmo es de la forma:

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \quad (2.2)$$

$$z_k = h(x_k, n_k)$$

Donde f y h son funciones no lineales, es decir, las matrices A_k , B_k y H_k están formadas por valores dependientes de las variables de estado del sistema y ya no son constantes como en el caso de los sistemas lineales.

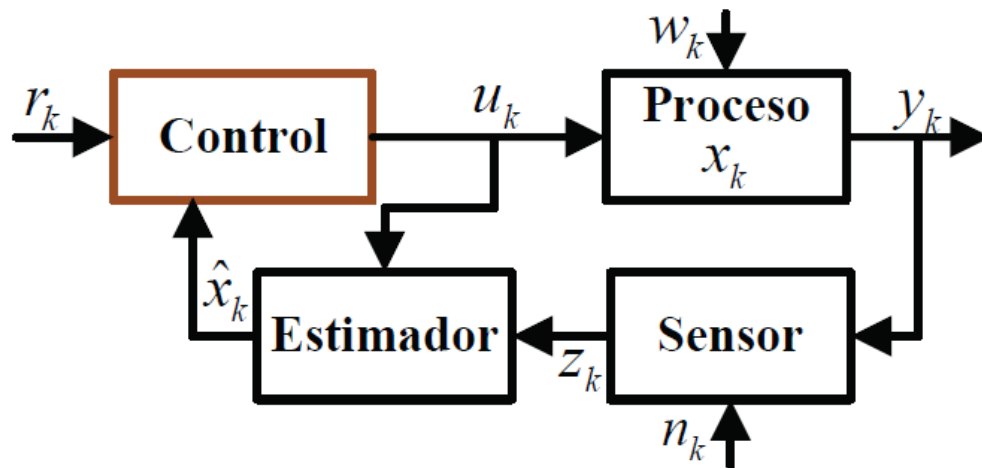


Figura 2. Lazo de control cerrado de un proceso con estimador [1]

Así pues, un estimador de estado es una técnica que se puede aplicar tanto en sistemas lineales como no lineales para estimar el estado de un proceso a partir de las medidas de los sensores disponibles y de las variables de entrada al proceso.

Existen diferentes métodos para la estimación del estado de un proceso. Dos de ellas son los observadores de estado y los filtros de estimación. A continuación, se desarrollan estos dos tipos de estimadores y se escoge aquella solución que, debido a sus características, mejor se adapte a las problemáticas planteadas para este TFG.

También cabe notar que en el anejo I se incluye un glosario con la notación más importante del TFG.

2.2. Los observadores de estado

Los observadores de estado son estimadores de estado que pueden ser aplicados tanto a sistemas lineales como a sistemas no lineales. Sin embargo, los observadores de estado empleados en la estimación de estados de sistemas lineales no se pueden utilizar para la estimación de estados de sistemas no lineales ya que no están diseñados para ello. Así pues, se pasan a caracterizar las diferencias que existen entre los observadores de estado para sistemas lineales y los observadores de estado para sistemas no lineales.

2.2.1. Observadores de estado en sistemas lineales

Los observadores de estado empleados en sistemas lineales se caracterizan porque requieren solamente de información del sistema que están observando para hallar la estimación de las variables de estado. Asimismo, estos modelos son no estocásticos, es decir, consideran que no existen ruidos ni en el proceso ni en la medición.

Por lo tanto, la ecuación (2.1) debe ser modificada y w_k debe ser redefinido como la diferencia entre el valor de la salida en el instante anterior, y_{k-1} , y el valor estimado de dicha salida en el instante actual, $\hat{y}_k = C\hat{x}_k$, multiplicado por una matriz de ganancia K_0 que hace que la dinámica del sistema sea más lenta que la dinámica del error de estimación mediante la ubicación adecuada de los polos del sistema.

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_{k-1} + K_0(y_{k-1} - C\hat{x}_k) \quad (2.3)$$

A la ecuación que redefine el ruido del proceso se le conoce como observador de Luenberger y se calcula según la ecuación (2.3) donde C es la matriz de salida y sus entradas son constantes, al igual que las de A y B . Por otra parte, el valor de K_0 es constante en muchos diseños, lo que supone que no se puede ajustar la dinámica en tiempo real. Sin embargo, debido a esto, el tiempo de computación disminuye y esto conlleva a una reducción de la memoria necesaria.

El diseño de estos estimadores no tiene en cuenta la precisión de los sensores ni para su diseño ni para el cálculo de la ganancia K_0 , lo cual es un inconveniente cuando se utilizan para la fusión de datos ya que no se les puede dar mayor importancia a aquellos sensores que tengan mejores características físicas.

2.2.2. Observadores de estado en sistemas no lineales

En este tipo de sistemas se emplean observadores estables, es decir, aquellos cuyo error de estimación tiende asintóticamente a cero o que hayan sido probados previamente en el modelo y se conozca que estabilizan a dicho modelo. Además, utilizan ganancias no lineales para la redefinición del ruido del proceso por lo que se recurre a las técnicas de estabilización de Lyapunov para poder obtenerlas [1]. Esto supone mayor tiempo para el diseño del

estimador. La ecuación que sintetiza estos observadores se conoce como observador no lineal de estado completo y viene dada por la ecuación (2.4):

$$\hat{x}_k = f(\hat{x}_{k-1}, u_{k-1}) + K_0(k, y_{k-1} - h(\hat{x}_{k-1})) \quad (2.4)$$

El diseño de dichos observadores se basa en que el modelo del sistema tiene que ser muy preciso, lo que hace que sea difícilmente parametrizable. Además, al producirse cualquier modificación en el proceso, como la conexión o desconexión de un sensor, deben ser rediseñados y esto los hace todavía más complejos.

Finalmente, si se tiene en cuenta que requieren sistemas de cálculo muy potentes para computar las ganancias, se puede deducir que son poco aplicables a sistemas que requieren fusión sensorial. Aun así, si se consigue modelizar correctamente el proceso y calcular las ganancias del observador, presentan buenos resultados además de que emplean poco tiempo de computación en aquellos casos donde K_0 es constante o no requiere de cálculos complejos.

2.3. Los filtros de estimación.

Los filtros son un tipo de estimadores de estado frecuentemente empleados en diversas aplicaciones de ingeniería. Los filtros consideran la existencia de ruidos en el proceso, w_k , y en la medición, n_k , ya que tienen en cuenta que el modelo del proceso es estocástico. Un ejemplo de esto son las ecuaciones (2.1) y (2.2). Por tanto, para la implementación de sus algoritmos, se tiene que diseñar la ganancia en el instante k (K_k) y mediante esta añadir el error de estimación. Además, necesitan un método que vaya incorporando y estimando los efectos de los ruidos n_k y w_k . Todos estos diseños diferirán dependiendo del tipo de filtro empleado pero generalmente se basan en teorías probabilísticas y estadísticas para describir el comportamiento de los ruidos y se recurrirá a técnicas de optimización para la obtención de la ganancia K_k con la finalidad de reducir los tiempos de computación [1].

Estos filtros se basan en un proceso matemático iterativo que utiliza un conjunto de ecuaciones y datos calculados en instantes anteriores para obtener rápidamente el valor real de la variable de estado actual asumiendo que los valores medidos contienen una incertidumbre aleatoria o impredecible. Para ello, dividen su metodología en dos etapas: predicción o *a priori* etapa en la cual se actualiza el valor estimado de la variable de estado y corrección o *a posteriori* donde se incorpora la información de los sensores. Todo esto se lleva a cabo para obtener algoritmos que sean computacionalmente eficientes.

A continuación, se introducen las características de los filtros para los dos tipos de sistemas, lineales y no lineales, y se detallan un par de ellos para ambos casos.

2.3.1. Filtros para la estimación de sistemas lineales

Este tipo de filtros recurren a propiedades estadísticas como son la media, la covarianza o la función de densidad de probabilidad *pdf* entre otras de una variable aleatoria para su desarrollo. Las propiedades más importantes y que se deben destacar son las evoluciones en el tiempo de la media (\bar{x}_k) y de la covarianza del estado (P_k) [1].

La media del modelo de un sistema se obtiene al calcular el valor esperado del modelo (2.1). En la ecuación (2.5) se muestra la media del modelo asumiendo que w_k tiene media cero.

$$E[x_k] = \bar{x}_k = A_{k-1}\bar{x}_{k-1} + B_{k-1}u_{k-1} \quad (2.5)$$

Por otro lado, la covarianza entre dos variables aleatorias x-y que son n-dimensionales, es decir, $x=(x_1, \dots, x_n)^t$ y $y=(y_1, \dots, y_n)^t$, se define como:

$$\sigma = E[(x - E[x])(y - E[y])^t] \quad (2.6)$$

Así pues, partiendo de las ecuaciones (2.5) y (2.6), en (2.7) se presenta el valor de la covarianza del modelo de un sistema teniendo en cuenta que el ruido del proceso w_k es gaussiano y blanco con covarianza Q_k .

$$P_k = E[(x_k - \bar{x}_k)(x_k - \bar{x}_k)^T] = A_{k-1}P_{k-1}A_{k-1}^T + Q_{k-1} \quad (2.7)$$

2.3.1.1. Filtro de Kalman (KF)

El filtro de Kalman (KF) proporciona una herramienta computacional eficiente para estimar el estado de un proceso mediante la minimización de la matriz de covarianza del error de estimación P_k . Hay diversas variaciones del filtro, pero este trabajo se centra en la aplicación de una de ellas: el KF en su formulación estándar para un sistema lineal.

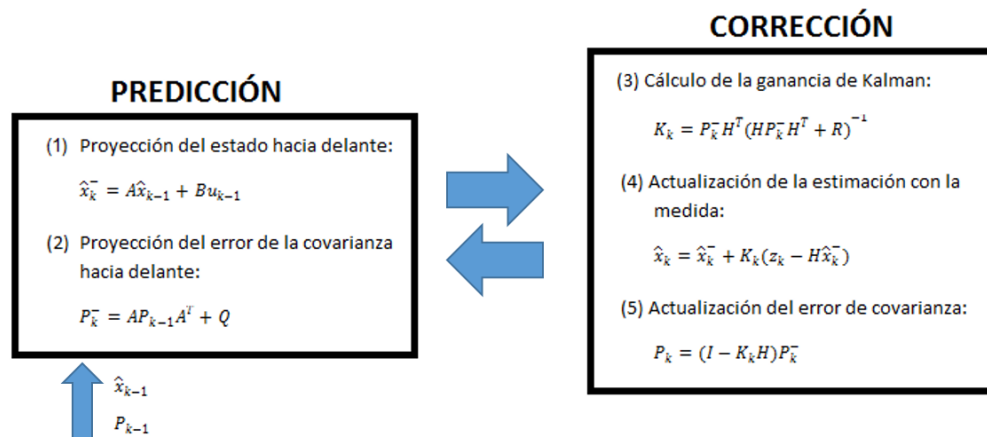


Figura 3. Esquema algoritmo filtro de Kalman

El objetivo del FK es estimar el estado x_k mediante el conocimiento de las dinámicas del sistema (modelo lineal), las características de las perturbaciones (matrices de covarianza) y la disponibilidad de la medida del sensor z_k . Este filtro minimiza los efectos de w_k y v_k en la estimación del estado. El esquema de trabajo del KF se muestra en la figura 3 donde se define $\hat{x}_k^- \in \mathfrak{R}^n$ como el estado estimado *a priori* en el periodo k (que se obtiene del proceso anterior al periodo k), y $\hat{x}_k \in \mathfrak{R}^n$ como el estado estimado *a posteriori* (en el periodo k usando la medición z_k). Los errores estimados *a priori* y *a posteriori* se definen como [2]:

$$e_k^- = x_k - \hat{x}_k^-; \quad e_k = x_k - \hat{x}_k \quad (2.8)$$

Las estimaciones de las covarianzas del error *a priori* y *a posteriori* son expresadas en (2.9) donde el operador E representa el valor promedio [2].

$$P_k^- = E[e_k^- e_k^{-T}] \quad P_k = E[e_k e_k^T] \quad (2.9)$$

La estimación del estado *a posteriori* \hat{x}_k se expresa como una combinación lineal de la estimación *a priori* \hat{x}_k^- y del error estimado en la medición, que es la diferencia entre la medida actual z_k y la predicción de la medición $H\hat{x}_k^-$ [2]:

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H \cdot \hat{x}_k^-) \quad (2.10)$$

El término $z_k - H \cdot \hat{x}_k^-$ se denomina valor residual y refleja la discrepancia entre la medición y la medición estimada para actualizar el estado estimado *a priori*. Aquí la matriz de ganancia K se escoge para minimizar la covarianza del error *a posteriori* P_k y se determina por una aproximación probabilística usando las covarianzas del error y del ruido. La minimización se lleva a cabo sustituyendo (2.10) en e_k (2.8), y entonces el resultado es sustituido en P_k (2.9). Como ya se ha mencionado, minimizando (2.10) con respecto a K la ganancia óptima del filtro K se obtiene como [2]:

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (2.11)$$

Esta ecuación necesita el valor de la covarianza *a priori* P_k^- que se obtiene de la definición en (2.9) y que a su vez requiere de la variable del sistema x_k . Desarrollando las ecuaciones, se obtiene que [2]:

$$P_k^- = A P_{k-1} A^T + Q \quad (2.12)$$

Para la definición de la matriz de covarianza *a posteriori*, se requiere del valor de la covarianza *a priori*, P_k^- , y se obtiene de su definición en (2.9) [2]:

$$P_k = (I - K_k H) P_k^- \quad (2.13)$$

En ciertos sistemas, puede considerarse el valor de la ganancia del filtro de Kalman constante, generando un ahorro computacional importante y haciendo al filtro útil en sistemas con recursos limitados. Sin embargo, cuando se requiere de una fusión sensorial, es necesario el cálculo de las matrices inversas dentro de la unidad de control, lo que aumenta los tiempos de computación de la misma.

Con las ecuaciones de (2.11) a (2.13) el KF puede ser implementado en un algoritmo. Dicho algoritmo puede ser planteado en dos etapas:

- Periodo de actualización (o Predictor): calcula la siguiente estimación del estado *a priori* a partir de la estimación previa del estado y del valor actual de la entrada; esto se utiliza para calcular la covarianza *a priori*.
- Actualización de la medida (o Corrector): utiliza la medición actual para mejorar el resultado dado en la parte anterior.

Para implementar el algoritmo del filtro de Kalman también se necesitan las matrices de covarianza del ruido del proceso Q y de las mediciones R. La matriz Q define la precisión del modelo para describir el proceso real, los términos de la diagonal son la varianza de cada estado y los términos de fuera de la diagonal son las covarianzas entre estados. Un valor más grande en la diagonal indica que se tiene menos en cuenta el modelo a la hora de hacer la

estimación del estado correspondiente. La matriz R se obtiene midiendo la covarianza del sensor. Valores bajos indican que el sensor tiene una alta resolución y el filtro tendrá que basar la estimación en estas mediciones (el modelo debe ser menos tenido en cuenta). Un valor grande hace que el filtro confíe más en el modelo de otros sensores (con valores más pequeños) que en la medición del sensor con el valor de covarianza grande. Otras matrices que también deben ser definidas son $\hat{x}_{k-1,ini}$ y $P_{k-1,ini}$. El estado inicial es conocido normalmente. Por ejemplo, en la inicialización de estas variables para el caso de la estimación de la posición de un robot móvil, se conoce si el robot parte o no del reposo y en qué punto del espacio estaba situado al comenzar el movimiento. La covarianza del error inicial se define como una matriz diagonal con términos positivos. Estos valores iniciales deben escogerse de manera cuidadosa para hacer que el filtro converja. Finalmente, para la definición de las matrices del proceso (A), de entrada (B) y de medición (H), se tendrá en cuenta el modelo y las matrices se diseñarán para hacer la conversión pertinente de las variables del sistema, de entrada y de medición, respectivamente, a las variables de estado.

Este tipo de filtro es la solución óptima en sistemas lineales cuyos ruidos w_k y n_k se caracterizan por tener una distribución probabilística independiente, blanca y Gaussiana con media cero. Sin embargo, si los ruidos no son Gaussianos, el filtro de Kalman es la mejor solución lineal pero no tiene por qué ser la más eficiente ya que un filtro no lineal, como por ejemplo el filtro extendido de Kalman, podría generar mejores estimaciones.

2.3.1.2. Filtro H^∞

Otro tipo de filtro empleado en sistemas lineales es el filtro H^∞ que tiene como finalidad minimizar el peor error de estimación mediante multiplicadores de Lagrange y teoría de juegos [1]. Además, su estructura es similar a la del KF ya que sigue las dos etapas de predicción-corrección. El filtro H^∞ se diseña para considerar posibles errores cometidos en el modelado (definición de A , B y H) y en la caracterización de los ruidos del proceso y de la instrumentación lo que supone que sean robustos ante cualquier variación del modelo o de los sensores empleados en el sistema.

Al no emplear hipótesis en la caracterización de los ruidos del proceso y de la medida, el filtro H^∞ se ajusta a diferentes tipos de perturbaciones y es más general que el KF. A pesar de ello, aumenta su complejidad en el cálculo computacional cuando se requiere de fusión sensorial debido a que hay que invertir gran cantidad de matrices y esto supone más tiempo de procesamiento por parte de la computadora y, por tanto, más memoria. Además, una de las mayores diferencias que tiene con el KF es la necesidad de una gran precisión en el ajuste de los parámetros [1], lo que conlleva a invertir más tiempo en la definición de los parámetros del filtro para obtener buenos resultados.

Por otro lado, este tipo de observador puede ser combinado con los KF para obtener un estimador con las ventajas de uno y otro, ya que combinan la minimización de la covarianza del error de estimación y del peor error de estimación mediante las ganancias lineales de los dos filtros. Pero tienen el inconveniente de una mayor necesidad de capacidad de cálculo en la computadora debido a la ejecución de ambos filtros de manera simultánea [1]. Asimismo, esto supone un mayor gasto computacional debido a que se ejecutan simultáneamente dos filtros.

2.3.2. Filtros para la estimación de sistemas no lineales

El desarrollo de los filtros para sistemas no lineales todavía se encuentra en proceso de investigación debido a su complejidad. Esto supone que solo sean aplicables a un rango muy acotado de sistemas ya que no convergen a una solución exacta y general que vaya describiendo la transformación de la densidad *pdf* en el sistema no lineal o la evolución de la matriz de covarianza del error de estimación P_k como ocurre en los sistemas lineales. Por tanto, en dichos sistemas se emplean soluciones aproximadas y subóptimas [1]. Un ejemplo de un sistema no lineal se da en la ecuación (2.2).

Al igual que sucede en los sistemas lineales, los filtros para sistemas no lineales requieren de diferentes variables estadísticas ya que estas se emplean en su implementación. Así pues, para la descripción de la evolución de la matriz de covarianzas del error, ya sea en la etapa de predicción como en la de corrección, se recurre a aproximaciones mediante la linealización del modelo y adaptándolo a las ecuaciones de los filtros empleados en sistemas lineales o mediante iteraciones de la función de densidad de probabilidad *pdf* del sistema no lineal que obtenga y actualice el valor de P_k [1].

A continuación, se detallan dos tipos de filtros para sistemas no lineales. El primero de ellos, el filtro extendido de Kalman, es la adaptación del filtro lineal de Kalman a sistemas no lineales. El segundo de ellos, el filtro de partículas, es un tipo de filtro muy utilizado actualmente en aplicaciones de localización [1].

2.3.2.1. Filtro extendido de Kalman (EKF)

Cuando se trabaja con procesos no lineales, tal y como se detalló en el apartado de introducción y notación, el modelo queda expresado de la forma:

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}); \quad z_k = h(x_k, v_k) \quad (2.14)$$

Ahora bien, el estado y la medición pueden ser aproximados considerando que tanto el ruido del proceso como el de la medición son nulos cuando el estado está próximo al punto de operación, quedando sus ecuaciones de la forma:

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_{k-1}, 0); \quad \hat{z}_k^- = h(\hat{x}_k^-, 0) \quad (2.15)$$

Donde \hat{x}_{k-1} es el estado estimado *a posteriori* en la iteración k-1. Con el fin de utilizar las ecuaciones del filtro ya desarrolladas para el KF lineal, el modelo (2.13) se linealiza entorno al punto de operación, así pues:

$$x_k \approx \hat{x}_k^- + A(x_{k-1} - \hat{x}_{k-1}) + Ww_{k-1} \quad (2.16)$$

$$z_k \approx \hat{z}_k^- + H(x_k - \hat{x}_k^-) + Vv_k$$

Donde \hat{x}_k^- , \hat{z}_k^- son las estimaciones de (2.15). Las matrices del sistema de (2.16) se linealizan por series de Taylor y se calculan en cada periodo k según muestran las ecuaciones de (2.17). Cabe notar que estas fórmulas consideran que el estado está próximo al punto de operación.

$$A_k = \left. \frac{\partial f}{\partial x} \right|_{\hat{x}_{k-1}, u_{k-1}, 0}$$
$$H_k = \left. \frac{\partial h}{\partial x} \right|_{\hat{x}_k, 0} \quad (2.17)$$

$$W_k = \frac{\partial f}{\partial w} \Big|_{\hat{x}_{k-1}, u_{k-1}, 0}$$

$$N_k = \frac{\partial h}{\partial n} \Big|_{\hat{x}_k, 0}$$

La ganancia del filtro de Kalman extendido se calcula de forma similar que en el KF (el desarrollo completo puede consultarse en Simon 2006 y Bozic 1994).

Por otro lado, hay que destacar también que el modelo puede ser linealizado hasta la primera o segunda derivada, dependiendo de la precisión requerida y de la capacidad computacional disponible.

En el filtro extendido de Kalman (EKF), se siguen considerando diferentes tipos de ruidos y se puede modificar para observar sistemas continuos. Asimismo, cuando la variable de estado está próxima al punto de operación, el ruido del proceso y el de los sensores son nulos. El esquema de trabajo del EKF se muestra en la figura 4.

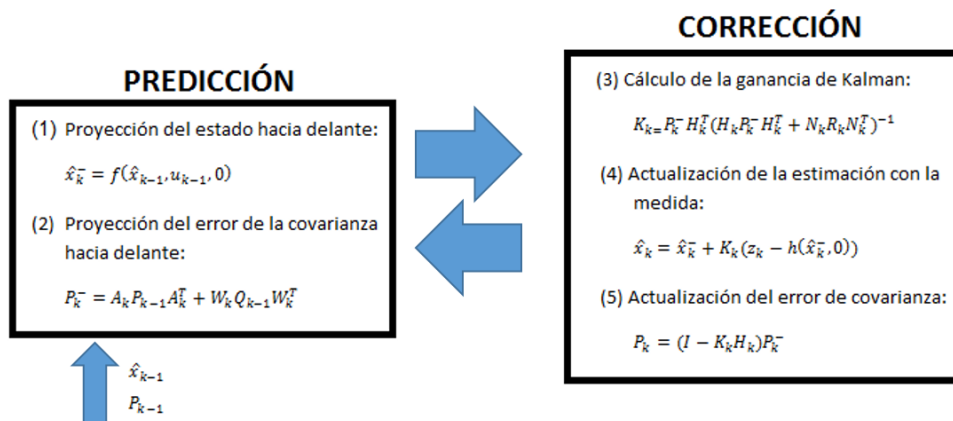


Figura 4. Esquema algoritmo filtro extendido de Kalman

2.3.2.2. Filtro de partículas (PF)

Cuando en los sistemas no lineales el filtro de Kalman no es capaz de generar una solución adecuada, la alternativa es el filtro de partículas el cual se basa en métodos de estimación exhaustiva para ir aproximando el valor de toda la pdf.

Para que este filtro converja a una solución exacta, se tiene que tener una cantidad de partículas o puntos (estados en nuestro caso) que tienda al infinito. Cada una de estas partículas tiene una probabilidad asociada y, conforme se van tomando medidas del sistema, dicha probabilidad se va modificando. Para generar la convergencia, en cada iteración se van eliminando aquellas partículas con una probabilidad baja y se duplican la cantidad de partículas con mayor probabilidad. Una iteración llegará a su fin cuando el número de partículas de la iteración k sea igual al número de puntos iniciales [1]. Así pues, cada vez se tienen más cantidad de partículas con una alta probabilidad asociada en torno al punto real de

trabajo o, en nuestro caso, mayor cantidad de estados muy similares entre sí que van convergiendo al estado real.

A pesar de que este tipo de filtro puede llegar a generar buenas soluciones en sistemas más complejos, requiere una capacidad de cálculo mucho más potente que no podrá ser aplicada en sistema con recursos limitados como es el caso del LEGO Mindstorm NXT.

En las figuras 6 se muestra un ejemplo de la implementación de este tipo de filtro en la localización de un robot móvil. En la iteración de (a) se puede observar que hay una alta cantidad de puntos donde puede situarse el robot mientras que solo unos pocos están en torno a la posición real (Posición del Robot). Unas iteraciones después, (b), el filtro ha ido eliminando aquellos puntos con baja probabilidad para duplicar los que tienen mayores posibilidades de situarse en torno al punto de localización real; esto supone que haya dos posibles posiciones del robot y una de ellas es la real. Finalmente, el filtro sigue haciendo iteraciones y todos los puntos están alrededor de la solución real, es decir, todos los puntos están muy próximos (alta probabilidad asociada) a la localización real del robot (c).

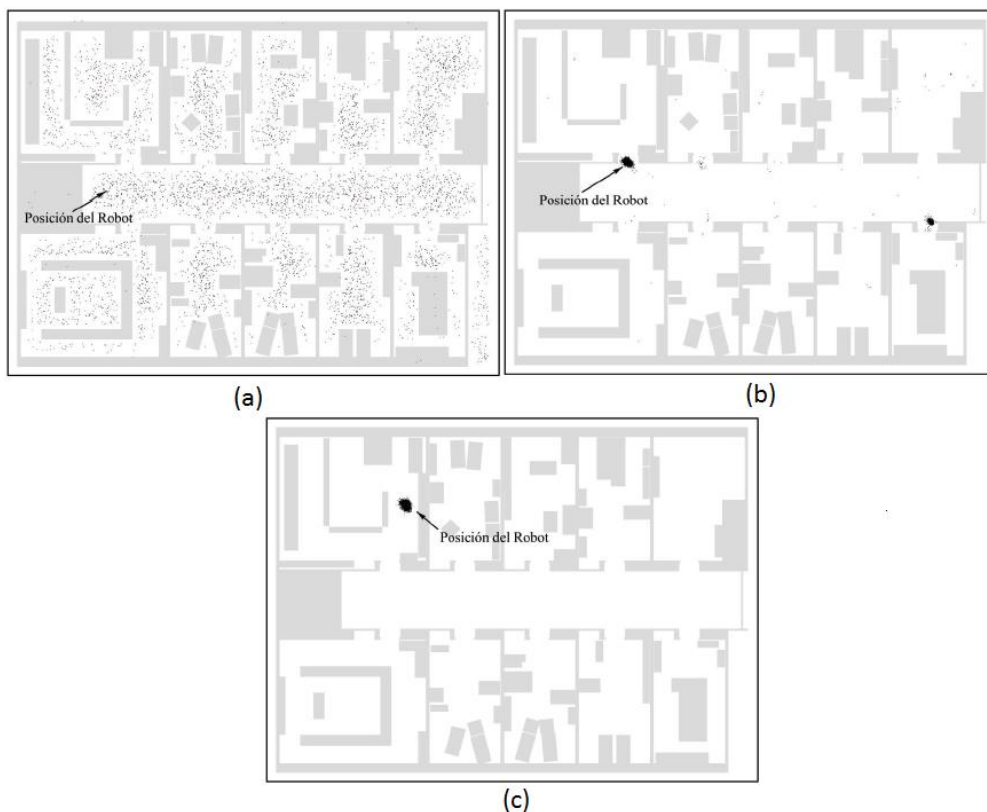


Figura 5. Localización de un robot con el filtro de partículas [1]

2.4. Conclusiones

Una vez introducidos los estimadores de estados, los observadores y algunos de los distintos tipos de filtros de estimación tanto para sistemas lineales como para sistemas no lineales, se debe seleccionar una de estas metodologías para la estimación de la velocidad de un robot móvil con la finalidad de eliminar los ruidos que aparecen en el sistema cuando sus ruedas derrapan además de estimar la posición de un segmento corporal mediante la fusión de la información procedente de una cámara de visión artificial y de un giróscopo.

Por un lado, como se ha visto previamente, los observadores pueden requerir menos recursos computacionales que los filtros. Sin embargo, no tienen en cuenta los ruidos del proceso ni de la medición además de que son difícilmente parametrizables ya que necesitan ser diseñados con modelos muy precisos. Asimismo, en el caso de sistemas no lineales, se recurre a las técnicas de Lyapunov para el cálculo de la ganancia del estimador por lo que se requiere de mayor tiempo para el diseño del observador. En cuanto a su aplicación para la fusión sensorial, no son técnicas flexibles ya que se debe modificar la implementación cuando se producen variaciones en el esquema, tales como la conexión o desconexión de algún sensor o la necesidad de darle mayor importancia a alguno de los sensores porque tiene mayor precisión.

Por otro lado, los filtros están basados en modelos estocásticos que consideran la existencia de ruidos en el proceso y en la medición. Tales filtros requieren de operaciones matriciales complejas, como es el cómputo con matrices inversas, con las cuales no se pueden trabajar en algunas de las plataformas de recursos limitados. Sin embargo, existe una gran variedad de filtros, tanto para sistemas lineales como no lineales, que pueden ser aplicados a la fusión sensorial debido a la facilidad que aportan tanto para la conexión o desconexión de sensores como para la asignación de mayor o menor importancia a alguno de esos sensores (modificación de la matriz R) mientras que el algoritmo está en ejecución.

Así pues, en el caso del robot móvil, van a existir ruidos en el proceso y en la medición cuando las ruedas derrapan. Por otro lado, en el caso de la estimación de la posición de un segmento corporal, se va a requerir fusionar los datos de dos sensores y además dotarles distintas importancias dependiendo de la situación. Asimismo, tanto la velocidad como la posición son variables que se pueden calcular linealmente, ya que la velocidad es la derivada de la posición.

En definitiva, se va a trabajar con sistemas lineales que van a requerir de un filtro de estimación. Dentro de los filtros de estimación para sistemas lineales, el filtro óptimo es el filtro de Kalman (KF) por lo que se escoge esta técnica para el desarrollo de los diferentes experimentos del TFG.

3. Objetivos

Una vez planteados los problemas que se van a afrontar en este TFG y encontrado un estimador de estado que se vaya a comportar de manera óptima en las soluciones desarrolladas, se busca establecer una serie de objetivos que marquen el camino para lograr el objetivo principal del TFG que es implementar el KF para estimar la posición de un segmento corporal de una persona mediante los datos de una cámara de visión artificial y de un Sistema Inercial de Medición (IMU).

Para lograr tal objetivo se deben cumplir previamente una serie de objetivos menores.

- Definir correctamente las matrices estadísticas del filtro de Kalman para que el sistema converja.
- Definir la trayectoria del segmento corporal mediante la estimación de la posición y de la velocidad.
- Dotar de mayor importancia a los datos procedentes de la cámara cuando esta detecte la posición del segmento corporal y de menor importancia cuando no sea así.

Por otro lado, también se definen tres objetivos que podrán resolver la problemática planteada hasta este momento con los robots y vehículos móviles. Estas problemáticas son: la necesidad de inicializar correctamente las matrices estadísticas del filtro de Kalman para que la estimación converja, la necesidad de estimar la velocidad de un robot móvil para que cuando derrapen sus ruedas, el controlador no interprete un movimiento que no es real y, finalmente, la necesidad de fusionar los datos procedentes de varios sensores para mejorar la estimación de la velocidad. Así pues, se definen los siguientes objetivos:

- Estudiar el comportamiento de las matrices estadísticas (P, Q y R) del filtro de Kalman mediante su inicialización.
- Mejorar la trayectoria circular de un robot móvil de dos ruedas mediante la estimación de la velocidad.
- Llevar a cabo la fusión sensorial en un robot móvil.

Como puede observarse, los objetivos planteados para el robot móvil son muy similares a los objetivos menores definidos para el caso de la determinación de la posición del segmento corporal. Es por ello que, cumpliendo los objetivos planteados para el caso del robot móvil, se darán por cumplidos los objetivos menores para el caso del segmento corporal.

4. Metodología

4.1. Introducción

En el presente capítulo se introduce el equipamiento empleado para después exponer como se va a implementar el filtro de Kalman para la estimación de las variables de estado de los dos procesos estudiados. Para ello, se introducen los equipos utilizados y las plataformas de programación empleadas en cada parte. También se definen las variables de estado de los dos sistemas y los modelos empleados mediante la definición de las matrices del KF. Finalmente, se detalla tanto el desarrollo de la experimentación como la implementación del KF para cada uno de los experimentos llevados a cabo.

4.2. Equipamiento empleado

4.2.1. LEGO Mindstorm NXT

En este apartado del TFG se propone el empleo del LEGO Mindstorm como una plataforma para llevar a cabo el trabajo práctico del robot móvil con el KF. A continuación, se detalla por qué se ha decidido emplear este material.

A pesar de que normalmente el material educativo empleado para el control de procesos es muy caro, es posible comprar diferentes plataformas de robots más económicas (ActivMedia's Pioneer robot, MIT's HandyBoard and Cricket controller cards, The LEGO Group's LEGO Mindstorms, etc.). Estas plataformas normalmente constan de controladores, sensores electrónicos, sistemas mecánicos low cost y robots pequeños. Estos robots tienen recursos limitados ya que no permiten tanta precisión como un robot industrial, pero tienen la suficiente exactitud para ser empleados en el proceso educativo. Una de estas plataformas es el LEGO Mindstorms NXT que es la evolución de la primera versión desarrollada en 1998 (LEGO Mindstorm RCX) en una colaboración entre LEGO y el Massachusetts Institute of Technology (MIT).

En enero de 2006 la nueva generación, el LEGO Mindstorms NXT, fue introducida en la Feria Internacional de Electrónica de Consumo. Además de otros cambios minoritarios en los sensores electrónicos y las piezas de construcción, la nueva versión incorporaba una nueva unidad de control: la NXT, la cual se basa en un potente microcontrolador de 32 bits: ARM7, con 256 Kbytes de memoria FLASH y 64 Kbytes de RAM. Para la programación y la comunicación, NXT tiene un puerto USB 2.0 y bluetooth. Además, esta unidad de control tiene también cuatro entradas y tres salidas analógicas. Dicha versión de LEGO ofrecía cuatro tipos de sensores electrónicos: de contacto, láser, de sonido y de distancia. Además de estos sensores básicos, también se pueden incorporar una gran variedad de sensores opcionales, como por ejemplo cámaras de visión, brújulas magnéticas, acelerómetros, giroscopios, buscadores infrarrojos, etc. Los elementos más destacados de esta versión fueron los actuadores. En este caso, consisten en motores de corriente continua que incorporan encoders. Estos sensores tienen una resolución de un grado, lo cual mejora la resolución de la versión previa.

En esta parte del TFG se va a trabajar con el robot móvil de configuración diferencial en las dos ruedas usando el juego básico del LEGO NXT y dos acelerómetros de la marca Hitechnic. El prototipo empleado se muestra en la figura 6.



Figura 6. Robot LEGO empleado durante el TFG

4.2.1.1. Entornos de desarrollo y programación para el LEGO

Para poder implementar tanto el modelo cinemático como el KF en el robot de LEGO, se necesita de algún entorno de desarrollo y programación que sea compatible con el LEGO Mindstorm NXT. Es por esto que en este apartado se detallan las distintas opciones existentes en el mercado y se escoge la más destacable entre todas las opciones.

Los sistemas de LEGO Mindstorm han tenido una gran popularidad desde su puesta en el mercado, es por ello que se pueden encontrar una gran variedad de entornos de desarrollo y programación para dichos sistemas que permiten trabajar con Linux, Mac y Windows. Algunos ejemplos de dicha amplia amalgama de entornos de desarrollo son: BricxCC, NBC (Next Byte Codes), NXC, leJOS (NXJ), pbLua, NXT# o RobotC.

Entre todos ellos destaca la plataforma RobotC, diseñada por la Universidad Carnegie Mellon (Pittsburgh, Pennsylvania, EEUU) para los sistemas operativos de Windows XP y Windows VISTA. Debido a que dicha plataforma está basada en el lenguaje de programación C, la cual es muy interesante para el desarrollo de proyectos de ingeniería con el LEGO Mindstorm NXT ya que C es uno de los lenguajes de programación más empleados en la industria.

Por otro lado, la gran desventaja que presenta RobotC es que no permite el cálculo matricial, es decir, no se pueden inicializar variables con forma matricial para operar con ellas. Esto conlleva a desarrollar todo el cálculo con matrices en forma de ecuaciones y programar las

líneas de código del modelo cinemático con estas ecuaciones, es decir, con las entradas de las matrices como números decimales o enteros.

En definitiva, a pesar de no permitir el cálculo con matrices, RobotC es la plataforma de programación, dentro de las compatibles para trabajar con el LEGO Mindstorm NXT, más importante debido a que emplea el lenguaje de programación C.



Figura 7. Logo RobotC con el LEGO Mindstorm NXT

4.2.2. Dispositivo de tecnología vestible (wearable)

El desarrollo de modelos biomecánicos de cuerpo completo (MBCC) es una de las líneas de investigación más activas en el campo de la Biomecánica, por sus importantes aplicaciones en el ámbito de la Medicina. Los temas de mayor relevancia son los relacionados con la Rehabilitación y las Técnicas de Diagnóstico y Valoración Funcional, siendo en éstos campos los que han dado lugar a más publicaciones científicas y al desarrollo de las aplicaciones de mayor interés comercial.

Los MBCC se basan en las herramientas y técnicas de análisis de la Dinámica de Sistemas Multicuerpo. A pesar de los avances conseguidos en este campo, su capacidad predictiva es bastante reducida, presentando limitaciones importantes para describir la cinemática o la dinámica del cuerpo humano [3].

Las desviaciones más importantes entre los MBCC y el cuerpo humano se refieren a [4]:

- La naturaleza de las articulaciones que no pueden representarse como pares inferiores (rótulas o pares de revolución) sino que obedecen a modelos mucho más complejos [5,6].
- La deformación de los tejidos blandos sobre el hueso que introducen artefactos en el análisis cinemático [7,8] y dinámico [9,10].

- La indeterminación dinámica en las acciones en las articulaciones, debido a la redundancia de las estructuras corporales [4,11].
- Las dificultades para personalizar los modelos mediante la incorporación de características antropométricas y parámetros inerciales individualizados [12].

A estas desviaciones hay que añadir las dificultades para controlar adecuadamente las condiciones experimentales. Así, la mayoría de las aplicaciones clínicas de los MBCC se realizan en el contexto del análisis dinámico inverso, donde se intentan estimar los esfuerzos a partir de movimientos registrados mediante marcadores externos, de parámetros inerciales estimados y de las fuerzas externas medidas mediante una plataforma dinamométrica [13]. Este enfoque solo utiliza la información mínima necesaria para calcular la salida del modelo y supone la imposibilidad de su validación experimental in vivo [4].

Por otra parte, las aplicaciones médicas suelen orientarse al estudio de gestos y movimientos “naturales”. Así, la mayoría de estudios biomecánicos se centran en la marcha humana, el desarrollo de gestos de la vida diaria o en determinados movimientos articulares. Estos gestos presentan una gran variabilidad y no son los adecuados para desarrollar protocolos de medición reproducibles [14,15].

El control de las condiciones experimentales pasa por un cambio de enfoque de los MBCC para aplicaciones médicas. En particular, se plantea complementar las pruebas basadas en gestos naturales con otras en las que se controla y monitoriza el movimiento para mejorar la respuesta del sistema y el control de los errores. Esta solución implica usar sistemas mecánicos que asistan el movimiento durante las pruebas y monitoricen variables de interés. Se trata de incorporar a las pruebas tradicionales actuadores mecánicos o sistemas robóticos, como los que ya se han desarrollado en el ámbito de la Rehabilitación.

Para el experimento llevado a cabo en este TFG, se plantea el siguiente modo de empleo del dispositivo. El médico especialista colocará en un segmento corporal el brazalete e irá indicando al paciente los movimientos que ha de realizar en la etapa de análisis y/o diagnóstico. Este brazalete cuando se pone en modo operación empieza a enviar al dispositivo móvil los datos del giróscopo (aceleración, velocidad...) de dicha extremidad. Para que el especialista pueda empezar el diagnóstico tiene que ejecutar la herramienta software diseñada e implementada para este fin. En ella se procesa la información suministrada por el dispositivo IMU y además con la ayuda de la cámara es capaz de reconocer y localizar en todo instante de tiempo el marcador, del cual se extrae las orientaciones y las translaciones. Con esta información entra en funcionamiento el filtro de Kalman diseñado para esta aplicación médica.

Durante el análisis puede que haya posiciones en las cuales el sistema de visión falle, debido al movimiento rotacional de algún segmento corporal, cuando esto se produce el sistema diseñado es capaz de predecir la localización y la rotación que tendría en ese instante el señuelo visual.

4.2.2.1. Descripción de la arquitectura

En las figuras 8 y 9 se pueden observar los dispositivos que forman el brazalete y el sistema móvil. Se puede observar una batería, para que el sistema wearable sea portable, un Sistema de Medición Inercial y un microcontrolador que lleva integrado el sistema Bluetooth.



Figura 8. Brazalete

En la figura 9 se puede observar el ecosistema propuesto, donde se puede ver una de las posibles ubicaciones del sistema wearable y el sistema móvil empleado para capturar y procesar toda la información necesaria.

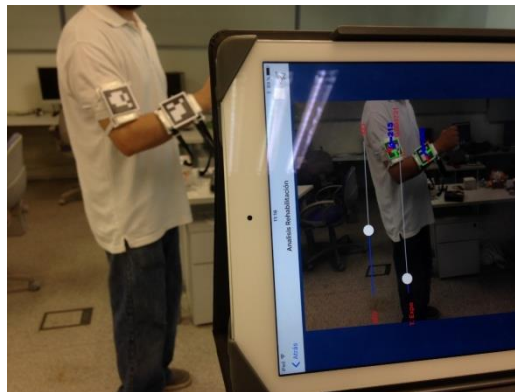


Figura 9. Sistema adquisición imagen, datos y fusión

El ecosistema propuesto para este sistema de análisis biométrico está compuesto por diferentes tecnologías electrónicas, figura 10. En primer lugar, se describe el sistema wearable. Este dispositivo está formado por un microcontrolador (ARM Cortex-M), un sistema de comunicaciones Bluetooth de bajo consumo (S110 nRF51 Bluetooth® low energy Peripheral SoftDevice) y un IMU de nueve ejes (BNO055 Intelligent 9-axis absolute orientation sensor), el cual internamente lleva implementado una serie de algoritmos de fusión de datos. Todo esto y un identificador de realidad virtual (Aruco) forman un brazalete. En segundo lugar, se describe el sistema móvil (Tablet), este está dotado de un microprocesador (ARM Cortex-A), de un sistema de comunicaciones BLE compatible con el que lleva el brazalete y una cámara para poder visualizar el señuelo que lleva el brazalete.

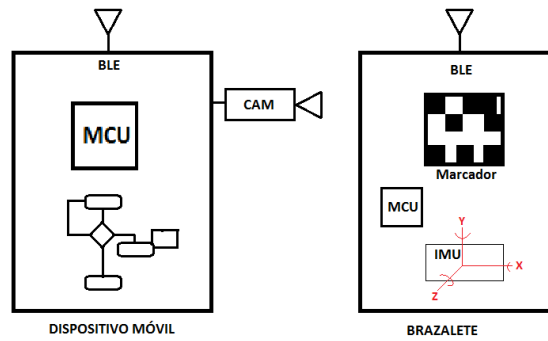


Figura 10. Esquema funcional del sistema

4.2.2.2. Entorno de desarrollo y programación empleado para el brazalete

En este TFG se va a trabajar con experimentos *offline*, es decir, se han tomado los datos del movimiento del brazo en un ensayo, se han guardado y se requiere procesarlos mediante algún entorno de programación en el ordenador. Para ello se recurre a una de las herramientas de software matemático más utilizada en todo el mundo por ingenieros y científicos, Matlab.



Figura 11. Logo de Matlab

Matlab fue desarrollada por MathWorks y está optimizada para resolver problemas de ingeniería. Algunos de los ámbitos donde ha sido utilizada son: sistemas de seguridad para automóviles, naves espaciales interplanetarias o redes eléctricas. El lenguaje de Matlab (lenguaje M) basado en el cálculo matricial es la forma más común para implementar códigos computacionales. Además, Matlab permite la representación de una gran cantidad de datos mediante diferentes tipos de gráficos lo cual lo hace muy útil para la visualización inmediata de los resultados obtenidos.

Hay que destacar que, en un futuro, si la tecnología wearable finalmente se emplea para diagnosticar pacientes, este entorno de desarrollo no podrá ser utilizado ya que se requerirá del tratamiento de los datos *online*.

4.3. Planteamiento de las problemáticas

4.3.1. Problemática con las estimaciones simples

El control automático de la trayectoria de un robot o vehículo móvil ha sido uno de los problemas más estudiados en el campo de la robótica. Dicho control puede llevarse a cabo midiendo la posición del motor y estimando la velocidad del mismo como la derivada de la posición respecto del tiempo. Así pues, se requiere de un modelo cinemático que permita calcular la velocidad y la posición del robot o vehículo móvil a partir de la velocidad de los motores de las ruedas.

Sin embargo, dicha aproximación, debido a su simplicidad, genera una serie de inconvenientes que cuando se necesita de una gran precisión en las trayectorias, no se puede emplear. Esto puede suceder, por ejemplo, debido a que las ruedas derrapan, por lo tanto, los sensores de los motores detectan un cierto giro y, en consecuencia, se estima la nueva posición suponiendo que ha habido un desplazamiento cuando, en realidad, la rueda ha girado y el robot no se ha desplazado. Otro de los inconvenientes que presenta controlar de forma automática la trayectoria de un robot móvil por aproximaciones simples es el error que se va cometiendo cuando la resolución de los sensores de las ruedas no es lo suficientemente precisa ya que, en este caso, se define una posición y una velocidad cometiendo un cierto error debido a las mediciones de los sensores.

En la asignatura de Laboratorio de Control y Automatización (LAC) se desarrollan unas prácticas con el LEGO Mindstorm NXT llevando a cabo justamente esto, es decir, el control de la posición de un robot móvil mediante la estimación de la velocidad como la derivada de la posición. El modelo cinemático empleado para estas prácticas se expone a continuación pero antes cabe notar que el prototipo que se desarrolla es un robot móvil de dos ruedas con configuración diferencial, por lo que se van a requerir, entre otras, las ecuaciones de la odometría para este tipo de configuración. A continuación, se detallan todas las ecuaciones que definen el modelo cinemático de un robot de dos ruedas con configuración diferencial.

El modelo cinemático de un robot móvil con configuración diferencial en las ruedas representa la evolución de las velocidades del robot en un sistema inercial fijo. La posición del robot se define por la posición de $P_0=(x,y)$ y el ángulo de cabeceo θ en el sistema de referencia global (ver figura 12).

Mediante el conocimiento de las velocidades locales tanto lineal como angular (v y ω), se puede definir la posición en el sistema de referencia global como:

$$\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (4.1)$$

De esta ecuación se puede estimar la posición y la orientación del robot usando la ecuación de la odometría [2]:

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \theta_{k-1} \end{bmatrix} + \begin{bmatrix} v_{k-1} T_s \cos(\theta_{k-1}) \\ v_{k-1} T_s \sin(\theta_{k-1}) \\ T_s \omega_{k-1} \end{bmatrix} \quad (4.2)$$

Donde T_s es el periodo de muestreo. La relación cinemática entre v y ω y las velocidades angulares en las ruedas ω_L y ω_R son [2]:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ -1/2 & 1/2 \end{bmatrix} \begin{bmatrix} \omega_L \\ \omega_R \end{bmatrix}, \quad \begin{bmatrix} \omega_L \\ \omega_R \end{bmatrix} = \frac{1}{T_s} \begin{bmatrix} \Delta s_L \\ \Delta s_R \end{bmatrix}, \quad \begin{bmatrix} v_L \\ v_R \end{bmatrix} = r \cdot \begin{bmatrix} \omega_L \\ \omega_R \end{bmatrix} \quad (4.3)$$

Donde $\Delta s_L = (\text{Enc}_{Lk} - \text{Enc}_{Lk-1}) \cdot \pi/180$ y $\Delta s_R = (\text{Enc}_{Rk} - \text{Enc}_{Rk-1}) \cdot \pi/180$ son los cambios en los valores de los encoders de las ruedas izquierda (Enc_{Lk}) y derecha (Enc_{Rk}) en radianes. Para incorporar los valores de las medidas de los acelerómetros y poder llevar a cabo la fusión sensorial de los dos encoders y los dos acelerómetros de Hitechnic, se debe diferenciar la ecuación (3.3) [2]:

$$\begin{bmatrix} a \\ \alpha \end{bmatrix} = \begin{bmatrix} \frac{dv}{dt} \\ \frac{d\omega}{dt} \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ -1/2 & 1/2 \end{bmatrix} \begin{bmatrix} \frac{d\omega_L}{dt} \\ \frac{d\omega_R}{dt} \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ -1/2 & 1/2 \end{bmatrix} \begin{bmatrix} a_L \\ a_R \end{bmatrix} \quad (4.4)$$

Donde a_L y a_R son las medidas tomadas por los acelerómetros de la rueda izquierda (a_L) y derecha (a_R).

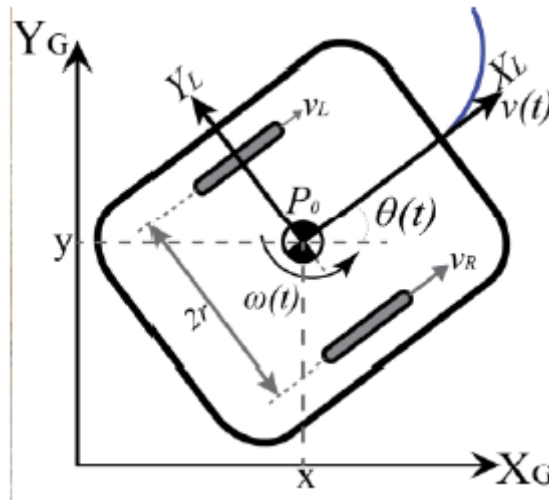


Figura 12. Sistemas referenciales del robot móvil [2]

Como puede observarse tanto la velocidad como la aceleración se obtienen diferenciando la posición y la velocidad, respectivamente, lo cual no siempre es una solución adecuada. Así pues, se plantea la modelización, implementación y experimentación con el KF para emplearlo como estimador de estado que estime tanto la posición del robot como la velocidad del mismo a partir de la velocidad de las ruedas. Esto supone que las ecuaciones (3.1), (3.2), (3.3) se pueden emplear para la estimación de la posición y de la velocidad de las ruedas del robot que se requieren para implementar correctamente el filtro de Kalman. Varios artículos científicos se han desarrollado acerca de esto, pero en este TFG se plantea utilizar estas estimaciones para trazar una trayectoria circular de 335mm de radio con el robot de LEGO y comparar las

variables estimadas por el filtro con las variables obtenidas sin la aplicación del estimador, es decir, en la asignatura de LAC.

Así pues, teniendo en cuenta las ecuaciones del modelo cinemático del robot con configuración diferencial en las dos ruedas, se debe dar paso a la modelización de las ecuaciones del filtro de Kalman.

4.2.2. Problemática con la toma de datos de los sensores del brazalete

Los experimentos de este apartado se plantean a partir de los datos suministrados por los compañeros del ai2 donde se proporcionan los ángulos muestreados por la cámara, las velocidades angulares medidas por la IMU y los tiempos de muestreo de cada uno de estos datos. Así pues, surgen dos problemáticas principales en la toma de datos de la cámara que se deben solucionar para que el filtro pueda generar las estimaciones correctamente.

La primera de ellas es la aparición de *outliers* en las medidas de la cámara. Los outliers son valores muy diferentes al resto de valores dentro de un conjunto de datos. Generalmente los outliers se deben a un error en la medición de los sensores. En la figura 13, donde se muestran los valores originales de la cámara facilitados por los compañeros del ai2 antes de ser procesados. Se puede observar como existen distintos valores atípicos dentro del conjunto de datos de la cámara. Estos valores son procesados para eliminarlos ya que no muestran un valor real de la medición de la posición del brazo.

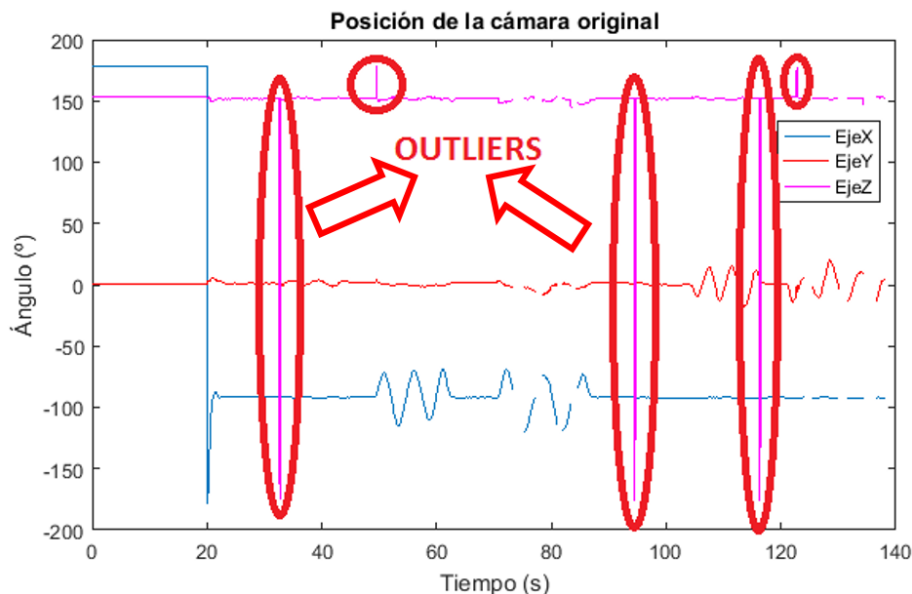


Figura 13. Outliers de la cámara

El segundo inconveniente que surge es debido a que la instrumentación trabaja con frecuencias distintas. Así pues, la cámara muestrea a una frecuencia de 8'9061Hz mientras que la IMU lo hace a 201'1937Hz. Esto supone que mientras la cámara ha tomado un dato, la IMU toma de 22 a 23 datos. Cuando se afronte esta problemática, se opta por trabajar a la frecuencia de la IMU para más tarde, por medio de un correcto diseño del filtro, se estimen los valores de la cámara tanto cuando no se detecta el brazalete como cuando no se tiene el dato debido a que la frecuencia de la IMU es más rápida.

En la figura 14 se han integrado los valores de la velocidad obtenidos con el IMU para obtener la posición angular del brazo y se han comparado con los datos de la cámara, se puede observar que el número de datos de la cámara (1232) es muy inferior al número de datos del IMU (27810).

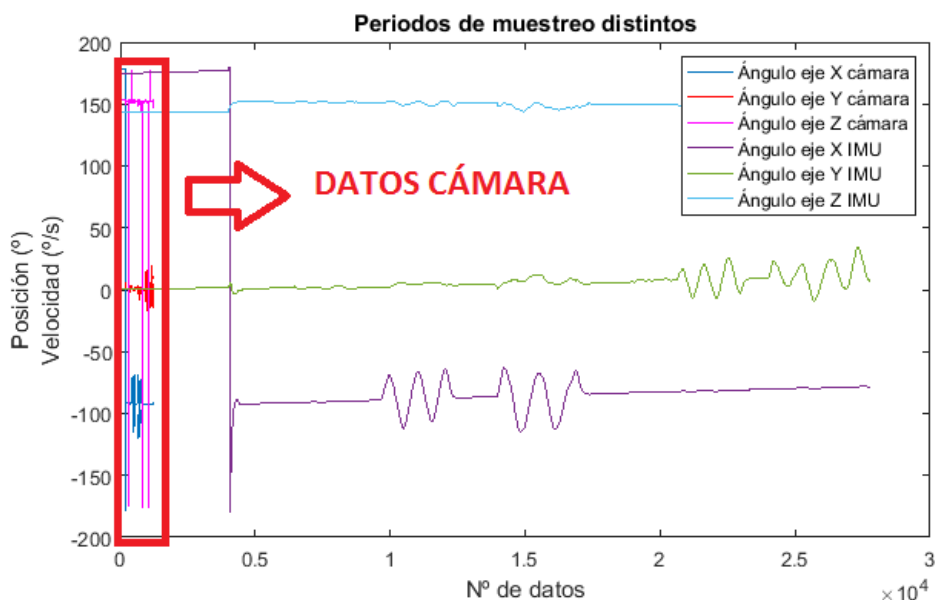


Figura 14. Datos de la cámara y de la IMU

4.3. Modelización del KF en el LEGO Mindstorm NXT

En este apartado se lleva a cabo la definición de las variables de estado, de entrada y de salida empleadas en la experimentación con el KF en el LEGO Mindstorm NXT además de las matrices del proceso (A), de entrada (B) y de medición (H). En cuanto a la modelización de las matrices de las covarianzas del error de estimación (P) y de los ruidos del proceso y de la medición (Q y R) se lleva a cabo en el capítulo de experimentación.

Cabe notar que en la parte de experimentación con el LEGO se llevan a cabo tres experimentos: estudio de la inicialización de las matrices de covarianzas del filtro de Kalman (P, Q y R), mejora de las medidas de los encoders de los motores de las ruedas y estimación de la velocidad lineal y angular del robot móvil mediante fusión sensorial.

4.3.1. Modelización para la inicialización de P, Q y R

En este primer experimento se llevan a cabo 10 ensayos donde, aplicándole a la rueda derecha del robot una velocidad de referencia de tipo escalón de una vuelta por segundo, se modifican los valores de las matrices características hasta llegar a un buen comportamiento de la velocidad de giro del motor. Tal comportamiento no es óptimo ya que las matrices se obtienen de manera experimental, es decir, sin seguir ninguna regla de optimización para las mismas.

Para ello, en primer lugar, se deben definir las variables de estado del filtro para este experimento que serán las variables de salida del proceso, es decir, serán la posición angular del motor (θ_m) y la velocidad angular del mismo (ω_m). Esto supone que el vector de estado sea:

$$x_k = \begin{bmatrix} \theta_m \\ \omega_m \end{bmatrix} \quad (4.5)$$

Cabe destacar que el robot parte del reposo por lo que:

$$x_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (4.6)$$

En cuanto a las variables de entrada, en este caso, a pesar de haber variables de salida del controlador, no se consideran que entren al estimador en el lazo de control del proceso (figura 2), por lo tanto, $u=0_{2 \times 1}$. Finalmente, queda definir cuáles serán las mediciones de los sensores y en este primer experimento solo se toman los datos del valor del encóder de la rueda derecha (EncR). Por tanto, el vector de medición queda definido como:

$$z_k = [EncR_k] \quad (4.7)$$

En cuanto a las matrices A, B y H, se deben ajustar para que las unidades de las variables de estado, de entrada y de medición estén bien ajustadas en las ecuaciones del filtro de Kalman. En este primer caso hay que considerar que no hay variables de entrada y por tanto la matriz B no será necesario definirla. Así pues, para el ajuste de las unidades de la proyección del estado hacia delante se considera que:

$$x_k = \begin{bmatrix} \theta_{m,k} \\ \omega_{m,k} \end{bmatrix} = A * \begin{bmatrix} \theta_{m,k-1} \\ \omega_{m,k-1} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} * \begin{bmatrix} \theta_{m,k-1} \\ \omega_{m,k-1} \end{bmatrix} = \begin{bmatrix} A_{11} * \theta_{m,k-1} + A_{12} * \omega_{m,k-1} \\ A_{21} * \theta_{m,k-1} + A_{22} * \omega_{m,k-1} \end{bmatrix} \quad (4.8)$$

Siendo A fácilmente ajustable como:

$$A = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \quad (4.9)$$

Para el caso del ajuste de las unidades de la matriz de medición, se considera que:

$$z_k - Hx_k = [EncR] - H * \begin{bmatrix} \theta_{m,k} \\ \omega_{m,k} \end{bmatrix} = [EncR] - [H_{11} \quad H_{12}] * \begin{bmatrix} \theta_{m,k} \\ \omega_{m,k} \end{bmatrix} = [EncR] - [H_{11} * \theta_{m,k} + H_{12} * \omega_{m,k}] \quad (4.10)$$

De donde se deduce que:

$$H = [1 \quad 0] \quad (4.11)$$

4.3.2. Modelización para la mejora de las medidas de los encóders

En este caso el filtro de Kalman se aplica al encóder de las ruedas (posición, $\theta_{L,R}$) para obtener una buena estimación de la velocidad de cada una de ellas ($\omega_{L,R}$). Como se necesitan los valores de la velocidad de la rueda izquierda y derecha para estimar la posición del robot en el sistema global de referencia y estas velocidades se calculan mediante las ecuaciones de la odometría (4.3), el error en la medición de la velocidad afecta a la estimación de la posición; lo que genera un comportamiento indeseado en el control de la posición y el movimiento del robot. Para evitar este problema se utiliza el filtro de Kalman (FK) con la finalidad de estimar la velocidad de las ruedas mediante la información de la medida de los encóders.

En este modelo, se va a aplicar dos veces el KF, una vez en cada rueda, por tanto hay que definir todas las variables y matrices que se emplean en el filtro dos veces. Sin embargo, se considera que el modelo de la rueda izquierda es el mismo que el de la rueda derecha con lo

que ambos modelos serán iguales con la excepción que en uno se meterá la medición del encóder de la rueda derecha y en el otro el de la rueda izquierda.

Las variables de estado de estos modelos serán la posición angular y la velocidad angular de los motores de las ruedas, es decir:

$$x_{k,R} = \begin{bmatrix} \theta_R \\ \omega_R \end{bmatrix}, \quad x_{k,L} = \begin{bmatrix} \theta_L \\ \omega_L \end{bmatrix} \quad (4.12)$$

Además, ambas variables serán inicializadas a cero ya que el robot parte del reposo, es decir:

$$x_{0,R} = x_{0,L} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (4.13)$$

Al igual que en el primer experimento, en este caso, no se considera que las variables de salida del controlador entren al estimador en el lazo de control del proceso (figura 2), por lo tanto $u_{R,L}=0_{2 \times 1}$ y la matriz de entrada no se debe definir. En cuanto a las mediciones, ahora se tomarán las medidas de los encóders de las dos ruedas, quedando los vectores de medición definidos como:

$$z_{k,R} = [EncR_k], \quad z_{k,L} = [EncL_k] \quad (4.14)$$

Para definir los valores de las matrices A y H, se siguen los mismos razonamientos seguidos en las ecuaciones (4.7) y (4.8), por lo que las matrices quedan definidas según (4.9) y (4.11), es decir, quedan definidas exactamente igual que en el experimento anterior.

4.3.3. Modelización para la estimación de las velocidades lineal y angular

En este caso el filtro de Kalman se aplica para la estimación de la velocidad lineal y angular del robot (v, ω) utilizando la ecuación (4.3) como medición e incorporando las medidas de los dos acelerómetros como *inputs* al sistema mediante la ecuación (4.4). Además, los valores de los acelerómetros se pre-procesan para ajustar las unidades al SI antes de introducirlos al KF.

En este modelo, las variables de salida a controlar son las velocidades lineal y angular del robot móvil y coincidirán con las variables estimadas, así pues:

$$x_k = \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (4.15)$$

Debido a que el robot parte del reposo, $x_0=0_{2 \times 1}$. Además, en este modelo se toman los valores de la aceleración como *inputs*, por lo que el vector de entrada está compuesto por la aceleración lineal y angular del robot, con lo que:

$$u_k = \begin{bmatrix} a_k \\ \alpha_k \end{bmatrix} \quad (4.16)$$

Finalmente, las mediciones son las velocidades lineal y angular calculadas a partir de los datos de los encóders según la ecuación de la odometría (4.3). Con lo que el vector de medición es:

$$z_k = \begin{bmatrix} v_{enc} \\ \omega_{enc} \end{bmatrix} \quad (4.17)$$

En este caso las matrices A y H se diseñan fácilmente ya que no hay que hacer ningún ajuste de unidades y ambas son:

$$A = H = I_{2 \times 2} \quad (4.18)$$

Sin embargo, la matriz de entrada pasa las unidades de aceleración a unidades de velocidad según (4.19):

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = B * \begin{bmatrix} a_k \\ \alpha_k \end{bmatrix} = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} * \begin{bmatrix} a_k \\ \alpha_k \end{bmatrix} = \begin{bmatrix} B_{11} * a_k + B_{12} * \alpha_k \\ B_{21} * a_k + B_{22} * \alpha_k \end{bmatrix} \quad (4.19)$$

De (3.19) se deduce que:

$$B = \begin{bmatrix} T_s & 0 \\ 0 & T_s \end{bmatrix} \quad (4.20)$$

4.4. Modelización del KF en el dispositivo de tecnología vestible

En este apartado se aplica el filtro de Kalman para llevar a cabo la fusión sensorial de los datos procedentes de la cámara de visión artificial y del dispositivo IMU. Para ello se pasan a definir los valores de las variables de estado, de entrada y de medición del KF para este caso además de las matrices del proceso (A), de entrada (B) y de la medición (H). Debido a que se han desarrollado dos hipótesis de partida distintas para la sincronización de los datos de los sensores, se han desarrollado dos modelos distintos.

Las variables de estado de este sistema son la posición angular que se obtiene de la cámara y la velocidad angular a la que se mueve el brazo por la cámara. Debido a que se trabaja en las tres dimensiones del espacio habrá tres posiciones angulares ($\theta_x, \theta_y, \theta_z$) y tres velocidades angular ($\omega_x, \omega_y, \omega_z$). Con lo que el vector de estado quedará definido como:

$$x_k = \begin{bmatrix} \theta \\ \omega \end{bmatrix} = \begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (4.21)$$

El movimiento del brazo no puede ser controlado por ninguna variable binaria, es por esto que en este modelo el vector de entrada es nulo ($u=0_{6 \times 1}$) y por tanto no hace falta diseñar la matriz de entrada B. El vector de medición está formado por el valor de las posiciones en las tres dimensiones del espacio que genera la cámara y el valor de la velocidad angular que proporciona el dispositivo IMU para cada una de las coordenadas, es decir:

$$z_k = \begin{bmatrix} \theta_c \\ \omega_{IMU} \end{bmatrix} = \begin{bmatrix} \theta_{c,x} \\ \theta_{c,y} \\ \theta_{c,z} \\ \omega_{IMU,x} \\ \omega_{IMU,y} \\ \omega_{IMU,z} \end{bmatrix} \quad (4.22)$$

Por otro lado, la definición de la matriz del proceso (A) viene dada por:

$$x_k = A * \begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} & A_{16} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} & A_{26} \\ A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & A_{36} \\ A_{41} & A_{42} & A_{43} & A_{44} & A_{45} & A_{46} \\ A_{51} & A_{52} & A_{53} & A_{54} & A_{55} & A_{56} \\ A_{61} & A_{62} & A_{63} & A_{64} & A_{65} & A_{66} \end{bmatrix} * \begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (4.23)$$

Razonando de la misma forma que en (3.8) y (3.19) pero ahora con matrices con seis variables de estado, se obtiene que:

$$A = \begin{bmatrix} 1 & 0 & 0 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 & T_s & 0 \\ 0 & 0 & 1 & 0 & 0 & T_s \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.24)$$

Para definir el valor de la matriz de medición (H) se considera que el vector de estado y el vector de medición tienen las mismas unidades, por tanto, $H=I_{6 \times 6}$.

4.5. Implementación de los códigos de programación

4.5.1. Implementación en el LEGO

Debido a que para la implementación de los códigos en el LEGO se utiliza el entorno de programación RobotC basado en lenguaje C y este no permite el cálculo matricial necesario para ejecutar el KF, en el Anejo II se desarrollan todos los cálculos para implementar el filtro de Kalman en forma de ecuación. Los ensayos 1 y 2 son las implementaciones para la inicialización de P, Q y R y para la mejora de las medidas de los encoders mientras que el ensayo 3 es la implementación para la estimación de las velocidades lineal y angular. Además, para llevar a cabo estos desarrollos, se han tenido en cuenta las matrices de modelización detalladas anteriormente.

Para llevar a cabo el control de las distintas variables de los experimentos desarrollados con el LEGO Mindstorm NXT, se utiliza el algoritmo del punto descentralizado. Este algoritmo se conoce que se comporta bien para estos fines ya que fue utilizado en la asignatura Laboratorio de Control y Automatización (LAC) para la determinación de la posición de un robot móvil mediante la estimación simple de su velocidad.

Por tanto, el filtro de Kalman es solo una parte del código implementado además de que los códigos se van modificando a lo largo de los experimentos. Así pues, se pasa a desarrollar los pasos seguidos para la implementación de los códigos empleados con el LEGO Mindstorm NXT en los diferentes experimentos que se han ensayado.

4.5.1.1. Implementación para la inicialización de P, Q y R

En este experimento, en primer lugar, se implementa el KF teniendo en cuenta que únicamente va a ser aplicado a uno de los motores del LEGO. Para ello, se escribe las ecuaciones del Anejo II y se inicializan sus variables para poder ejecutar el código.

A continuación, se inicializa una referencia de una vuelta por segundo para la referencia de la velocidad angular y se define un tiempo de ejecución de 2 segundos para que el motor

estabilice su velocidad. Además, se establece un periodo de muestreo de 20ms y se inicializan los valores de los encóders a cero. Una vez ya se tiene esto, se lee el valor de los encóders y se hace el cambio de unidades de grados a radianes.

Finalmente, se añaden las ecuaciones y las variables del controlador al código y se genera un fichero de datos donde se archivan los valores de la referencia, de la velocidad de la rueda sin aplicar el KF, de la estimación de la velocidad de la rueda calculada con el KF y de la acción de control que se le va aplicando al motor.

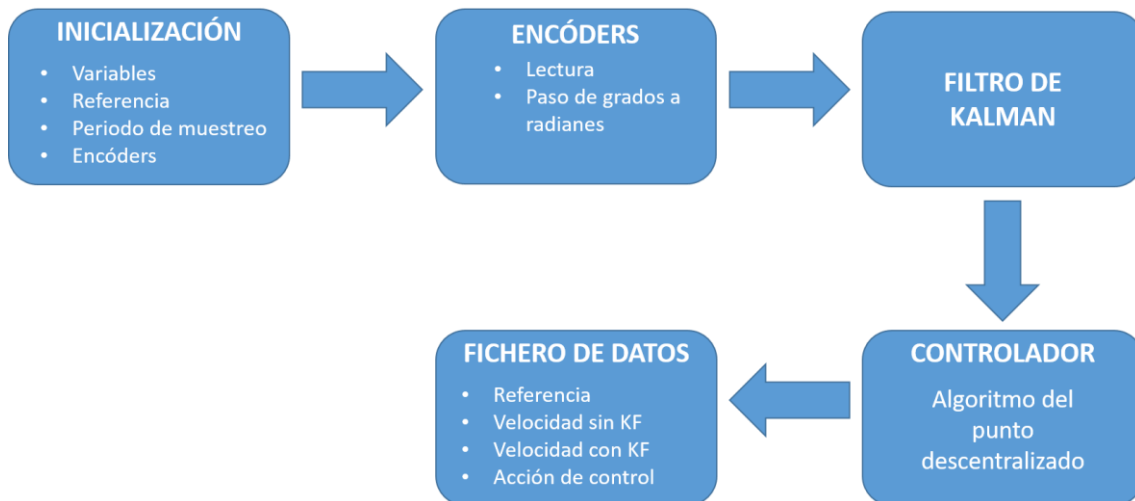


Figura 15. Esquema implementación para la inicialización de las matrices estadísticas

En la figura 15 se puede observar un pequeño esquema del código implementado.

4.5.1.2. Implementación para la mejora de las medidas de los encóders

Para la mejora de las medidas de los encóders, se parte del código desarrollado en la asignatura de LAC para controlar la trayectoria circular de radio 335mm de un robot móvil mediante la estimación de la velocidad de manera simple. Además, se tiene en cuenta lo previamente explicado para la implementación del filtro de Kalman en RobotC y la modelización del KF para esta parte.

Así pues, al código de LAC se le añaden las ecuaciones del anexo II del filtro de Kalman tanto para el motor de la rueda derecha como para el motor el de la rueda izquierda y se declaran todas las variables empleadas para ello. A continuación, a la lectura de los encóders se le hace el cambio de unidades de grados a radianes.

Todo seguido, se sustituyen la posición de los encóders y la velocidad estimada de manera simple por la posición y la velocidad estimada con el KF, respectivamente, en las ecuaciones del controlador. De esta forma, al dirigir el comportamiento del robot, se tendrá en cuenta la estimación de la velocidad y de la posición y se evitara los posibles ruidos que aparecen porque las ruedas derrapan.

Finalmente, se archiva en el fichero de datos los valores de la referencia de la trayectoria circular y de la trayectoria circular descrita por el robot en las dos coordenadas del plano, así como, las velocidades angulares estimadas tanto de forma simple como por el filtro de Kalman y las acciones de control de cada rueda.

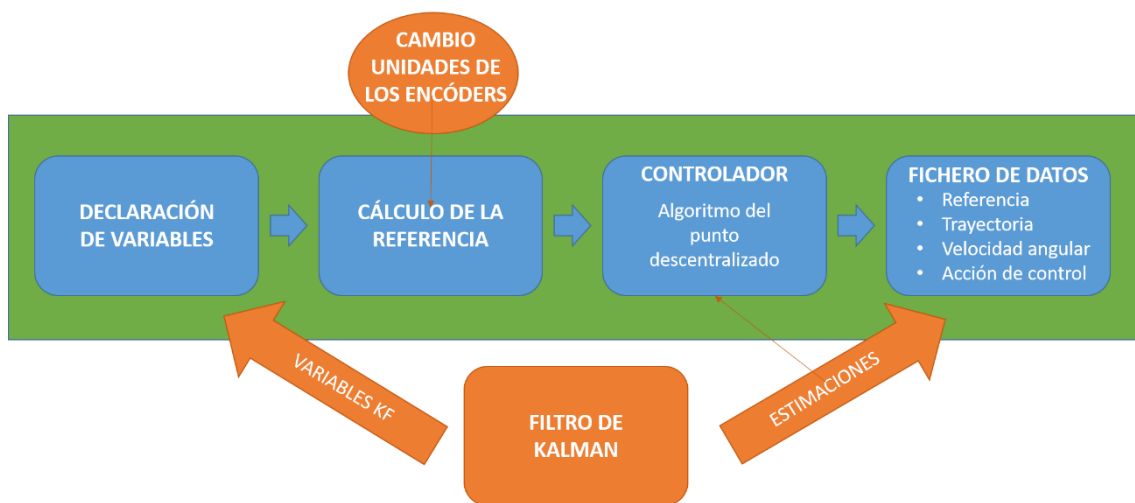


Figura 16. Esquema de la implementación para la mejora de medidas de los encoders

En la figura 16, recuadrado en verde, se muestra el esquema del código implementado en la asignatura de LAC y en naranja se detallan las modificaciones que se le han hecho al mismo para la incorporación del filtro de Kalman.

4.5.1.3. Implementación para la estimación de las velocidades lineal y angular

Para la estimación de las velocidades lineal y angular del robot, se parte del código implementado anteriormente para la mejora de las medidas de los encoders y se le realizan las modificaciones pertinentes.

En primer lugar, se ajustan las matrices A, B y H al nuevo modelo y a las nuevas unidades de medición y se desarrollan las líneas de código del KF teniendo en cuenta lo detallado en el apartado de modelización y adjuntadas en el Anexo II.

A continuación, se modifica la localización en el código de las estimaciones simples de la velocidad (4.3) ya que en este caso van a ser incorporadas al KF como mediciones. Detrás de estas ecuaciones se escriben las ecuaciones de la estimación simple de la aceleración (3.4) que serán incorporadas como *inputs* al sistema.

Finalmente, en las ecuaciones del controlador, se sustituyen las variables de la velocidad lineal y angular por las nuevas estimaciones calculadas con el filtro de Kalman. Asimismo, se archivan en el fichero de datos creado la velocidad lineal y angular tanto estimada de forma simple como calculada por el filtro de Kalman.

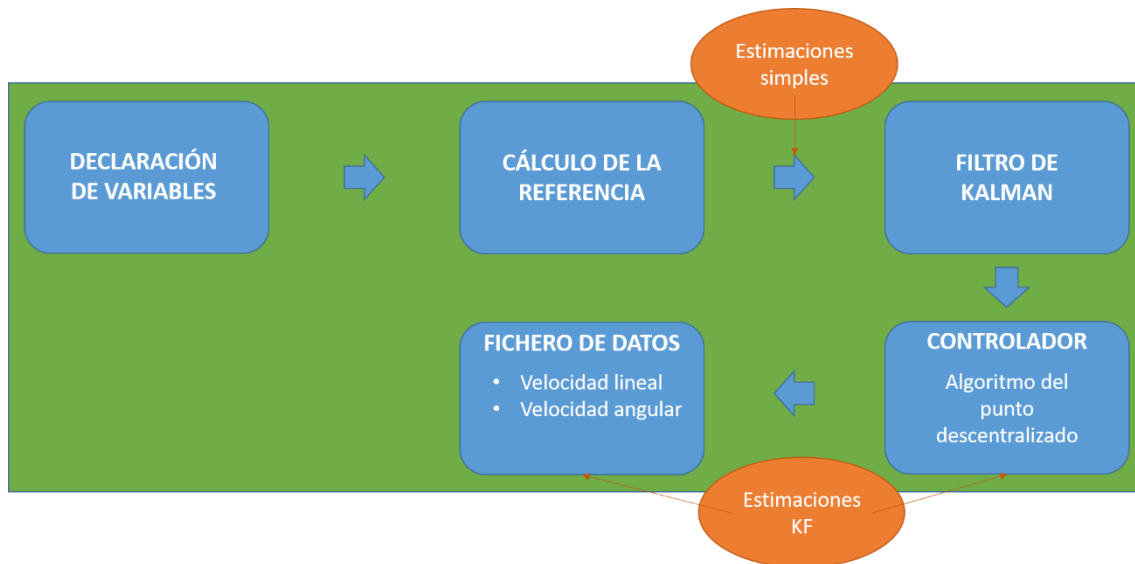


Figura 17. Esquema de implementación para la estimación de la velocidad lineal y angular

En la figura 17 se recuadra en verde el código implementado para el apartado anterior y en naranja se esquematizan las modificaciones que se le han realizado para estimar la velocidad lineal y angular en lugar de la posición y velocidad angular.

4.5.1. Implementación del filtro de Kalman en el brazalete

El entorno de programación empleado con el brazalete es Matlab. Esta herramienta de software matemático utiliza lenguaje de programación basado en el cálculo matricial por lo que no se requiere de un desarrollo muy complejo de las ecuaciones. En este caso, se implementa el código a partir del conocimiento previamente adquirido con RobotC.

Para la implementar el código, en primer lugar, se escriben las cinco ecuaciones del filtro de Kalman y se inicializan sus variables para poder ejecutar el código. A continuación, se inicializa la frecuencia de muestreo para poder calcular la matriz del proceso A.

Finalmente, se diseña un código que identifique cuando el dato de la cámara no es un valor de posición y es que la cámara no ha detectado el brazalete para poder ajustar tanto el vector de medición z_k como la matriz de covarianza del ruido de la medición R.

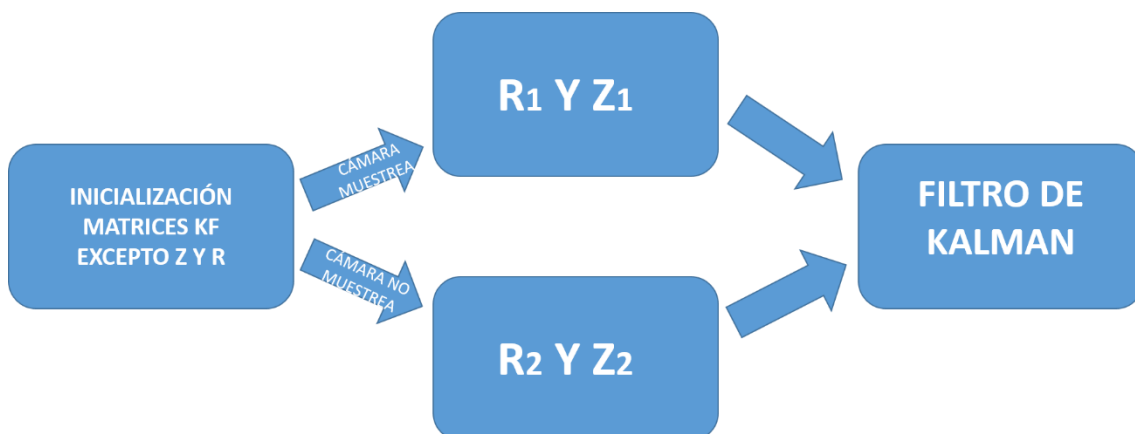


Figura 18. Esquema de implementación para el brazalete

En la figura 18 se puede observar el esquema seguido por el código de programación para dotar de mayor a la cámara cuando muestrea.

4.6. Experimentación con LEGO Mindstorm NXT

En este apartado se definen los valores de las matrices P, Q y R del KF además de que se detallan los pasos seguidos en cada uno de los experimentos para la ejecución del programa implementado en el apartado anterior con el filtro de Kalman.

4.6.1. Estudio de la inicialización de las matrices características del KF

En este primer ensayo se inicializan todas las matrices (P_0 , Q y R) a cero, pero debido a que para el cálculo de K_k se requiere de una división, se produce un error de compilación. Es por ello que se modifica el valor de la matriz de ruido de la medición a $R=0'1$ (ensayo 1).

A continuación, se analiza el comportamiento de la matriz P_0 sobre el sistema. Para esto, se comienza dándole un valor al azar, el valor de la matriz identidad, y se mantienen el resto de valores como en el primer ensayo (ensayo 2). Después se siguen aumentando los valores de la diagonal de la matriz de covarianza del error de manera muy brusca (ensayo 3). Una vez alcanzadas las primeras conclusiones, se pasa a darle valores positivos y distintos de cero (100) a las entradas no diagonales de la matriz de covarianza del error quedando el comportamiento de la matriz P_0 totalmente definido (ensayo 4).

ENSAYO	P_0	Q	R
1	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$	0'1
2	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$	0'1
3	$\begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$	0'1
4	$\begin{bmatrix} 100 & 100 \\ 100 & 100 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$	0'1
5	$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	0'1
6	$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$	0'1
7	$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$	0'1
8	$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 10 \end{bmatrix}$	0'1
9	$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 10 \end{bmatrix}$	10
10	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 3 \end{bmatrix}$	0'1

Tabla 1. Ensayos para la caracterización de las matrices del KF

El siguiente paso es estudiar el comportamiento de la matriz de ruido del proceso (Q). Para ello, a partir de un valor de P_0 que se comporta adecuadamente, se modifican los valores de Q. Cabe destacar que la matriz de ruido del proceso tiene que ser diagonal para que no afecte al

comportamiento de P_0 en las ecuaciones del FK; esto se debe a que para el cálculo de K_k ambas matrices se suman.

Al igual que en el caso anterior, se comienza con el valor identidad (ensayo 5) para la matriz Q y se prueba qué resultados tenemos si hacemos nulos los distintos valores de la diagonal de la matriz. En el ensayo 6 se anula la entrada Q_{22} , en el ensayo 7 la entrada Q_{11} y en el ensayo 8 se aumenta el valor de la entrada Q_{22} .

Por último, se aumenta el valor de R a 10 (ensayo 9) y se escogen los valores de las matrices P_0 , Q y R que mejor estiman la velocidad del motor de la rueda (ensayo 10).

En la tabla 1 se especifican los valores de cada matriz en los diferentes ensayos que se llevan a cabo.

4.6.2. El filtro de Kalman para mejorar la medida de los encoders

En este segundo experimento con el LEGO Mindstorm NXT, se le aplica a cada rueda del robot móvil los valores de las matrices P , Q y R del ensayo 10 (tabla 1) del primer experimento con el LEGO.

Sin embargo, debido a la limitación de la memoria del LEGO y a la gran cantidad de datos empleados, la trayectoria trazada por el robot no llegaba a completarse, es decir, no se generaba un círculo cerrado. Así pues, se ejecuta hasta tres veces el programa para obtener: la trayectoria circular completa trazada por el robot móvil, la velocidad de cada rueda estimada por el filtro y la acción de control que se ejerce en cada motor de manera separada. De esta forma, el robot no tenía que almacenar tanta cantidad de datos y podía describir la trayectoria completa.

Asimismo, se ejecuta también el programa que se ha desarrollado en la asignatura de LAC, sin el filtro de Kalman, para obtener la trayectoria de referencia y la trayectoria que describe en este caso el robot, además de la velocidad del robot calculada a partir de los datos de medición y las acciones de control que se ejercen sobre cada motor sin el KF. En este caso como el número de operaciones es mucho menor, con ejecutar únicamente una vez el programa es suficiente para que el robot trace la trayectoria circular completa.

Cabe notar que no se modifican los valores de las matrices estadísticas del filtro de Kalman debido a que los resultados que se obtienen al trazar la trayectoria circular completa tienen una precisión acorde a la búsqueda. Aunque para ver tal precisión se requiere de hacer zoom sobre la trayectoria ya que la referencia está en unidades milimétricas mientras que a gráfica que se genera está en unidades métricas.

4.6.3. Estimación de la velocidad lineal y angular del LEGO Mindstorm NXT

Para llevar a cabo este experimento, se parte de la implementación del código previamente descrita y se inicializan los valores de las matrices estadísticas del filtro de Kalman con los valores de los ensayos anteriormente detallados para la experimentación con el LEGO. Sin embargo, estos valores no generan una estimación que mejore la velocidad lineal y angular del robot debido a que los resultados de ambos casos son muy similares a los reales, como más tarde se puede observar.

Así pues, con la finalidad de que la estimación fuera mejor que los valores reales de las velocidades, se comienza modificando la matriz de covarianza del ruido de la medición (R) para estudiar cómo cambia el funcionamiento del sistema ante otros valores. Una vez queda definido este comportamiento, se van modificando las entradas de la matriz de covarianza del ruido del proceso (Q) para ver como afectaba cada una de sus entradas a los resultados de la estimación de las velocidades (lineal y angular). Finalmente, se definen los valores de las entradas de la matriz de covarianza del error de estimación manteniendo constantes los valores de Q y R que se habían comportado adecuadamente en sus respectivos estudios.

Ensayo	P_0	Q	R
1	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 3 \end{bmatrix}$	$\begin{bmatrix} 0'1 & 0 \\ 0 & 0'1 \end{bmatrix}$
2	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 0'1 & 0 \\ 0 & 0'1 \end{bmatrix}$
3	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 10 & 0 \\ 0 & 0'1 \end{bmatrix}$
4	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix}$
5	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$	$\begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$
6	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$	$\begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$
7	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$	$\begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$
8	$\begin{bmatrix} 300 & 0 \\ 0 & 300 \end{bmatrix}$	$\begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$	$\begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$
9	$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$	$\begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$	$\begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$
10	$\begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}$	$\begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$	$\begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$

Tabla 2. Ensayos para la estimación de las velocidades lineal y angular

En la tabla 2 se muestran los valores probados durante los 10 ensayos que se llevan a cabo en este apartado, pero hay que destacar que únicamente el ensayo 10 genera unas soluciones adecuadas para las dos variables de estado (velocidades lineal y angular), es decir, las estimaciones de las dos velocidades son mejores que los valores originales de las mismas. Es por ello que únicamente se mostrarán los resultados de la primera y la última prueba que se llevan a cabo ya que el ajuste de las matrices ya se desarrolla en el primer experimento con el LEGO Mindstorm NXT.

4.7. Experimentación con el brazalete

En este apartado se plantea la eliminación de los outliers y la estimación de los valores no medidos por la cámara mediante la correcta definición e inicialización de P_0 , Q y R. Debido a que se han realizado dos modelos para afrontar el problema, hay que considerar cada uno de los modelos por separado tanto para la eliminación de los outliers como para la estimación de los datos de la cámara.

Previamente a la ejecución de las cinco ecuaciones del esquema del KF (figura 3), se plantean tres casuísticas. La primera de ellas se da cuando la cámara no toma ningún dato debido a que no detecta el marcador del brazalete y, por tanto, se guarda el valor "Not a Number" (NaN) como valor de la posición angular de la cámara en el fichero de datos de la aplicación. El segundo caso que se puede dar es que en el periodo de muestreo k, debido a que el

dispositivo IMU tiene una frecuencia mucho mayor que la de la cámara, se guarda el dato de la IMU, pero al valor de la cámara se le asigna el número NaN. La última posibilidad es que la cámara detecte el marcador y tome el dato de la posición angular de este, en tal caso, es posible que el dato sea correcto o se produzca un error y el dato sea un outlier.

Otra consideración que también se debe tener en cuenta es que el dispositivo IMU tiene peor precisión que la cámara. Así pues, debido a que el filtro de Kalman permite darles mayor o menor importancia a los sensores, al dispositivo IMU no se le debe dar demasiada relevancia comparándolo con la cámara, es decir, se le debe asignar un valor grande a las entradas de la matriz R correspondientes a los valores de este sensor.

Así pues, en la ecuación (4.25), se definen los valores de partida empleados en las matrices P_0 y Q para este experimento. Dichos valores también se modifican una vez definida la matriz R para ver qué cambios generan en el comportamiento del sistema.

$$P_0 = Q = 60 * I_{6 \times 6} \quad (4.25)$$

Para definir el valor de R, se debe considerar si el valor leído en el fichero de datos es NaN o no. En el caso de que sea NaN, independientemente de cual sea la causa (si porque la cámara no detecta el marcador o porque es un valor para sincronizar los datos de la cámara y el IMU), al valor de la posición de la cámara se le asignará, en las tres coordenadas, el valor cero mientras que el valor de la matriz de covarianza del ruido de los sensores vendrá dado por:

$$R_1 = \begin{bmatrix} \infty & 0 & 0 & 0 & 0 & 0 \\ 0 & \infty & 0 & 0 & 0 & 0 \\ 0 & 0 & \infty & 0 & 0 & 0 \\ 0 & 0 & 0 & S & 0 & 0 \\ 0 & 0 & 0 & 0 & S & 0 \\ 0 & 0 & 0 & 0 & 0 & S \end{bmatrix} \quad (4.26)$$

Donde habrá que ir dándole valores a S hasta observar un buen comportamiento del KF. Este valor debe ser grande para que se considere la precisión del IMU. En cuanto al valor de las posiciones de la cámara, que son infinito, se les da este valor ya que el valor muestreado no se corresponde en ningún caso con la realidad por lo que tiene que ser lo mayor posible para que el filtro, cuando haga la estimación, no tenga en cuenta este valor y haga la estimación mediante la primera aproximación (etapa de predicción) sin incluir las medidas de los sensores (etapa de corrección).

Por otro lado, en el caso de que el valor almacenado en el fichero de datos durante el ensayo sea un número, es decir, no sea NaN, se debe considerar la posibilidad de que el valor sea un outlier y que es posible que este afecte de manera diferente a cada una de las tres coordenadas, es decir, que sea un valor mucho más falso en una coordenada que en otra. En cuanto a los datos tomados por el IMU, deben proponer menor precisión que los de la cámara con lo que los valores de la matriz R correspondientes a las entradas del dispositivo IMU deben ser más grandes que los valores de las entradas correspondientes a la cámara e iguales a los valores anteriormente dados en la ecuación (4.26). Con todo esto, la matriz R en este caso se define como:

$$R_2 = \begin{bmatrix} X & 0 & 0 & 0 & 0 & 0 \\ 0 & Y & 0 & 0 & 0 & 0 \\ 0 & 0 & Z & 0 & 0 & 0 \\ 0 & 0 & 0 & S & 0 & 0 \\ 0 & 0 & 0 & 0 & S & 0 \\ 0 & 0 & 0 & 0 & 0 & S \end{bmatrix} \quad (4.27)$$

En la tabla 3 se muestran los valores probados para cada una de las variables X, Y, Z y S que forman la matriz R de las ecuaciones (4.26) y (4.27).

Ensayo	X	Y	Z	S	P ₀	Q
1	1	1	20000	20000	60	60
2	1	100	100	20000	60	60
3	1	1	20000	20000	60	60
4	1	1	20000	100	60	60
5	1	1	20000	20000	60	0'1

Tabla 3. Ensayos con el brazaete

En los ensayos 1 y 2 se lleva a cabo la eliminación de los outliers mediante la modificación de las posiciones Y-Z de la matriz R ya que son la que presentan valores más atípicos. En los ensayos 3 y 4 se hace una comparación de la velocidad angular estimada para diferentes valores de la variable S de las matrices R. Finalmente, en el ensayo 5, se modifica el valor de la matriz Q para ver si el comportamiento del sistema se ve afectado. Una vez se posee la información de estos 5 ensayos, se definen cuáles son los valores de las matrices estadísticas del filtro de Kalman que mejor estiman el comportamiento del sistema.

Cabe notar que además de este experimento se llevan a cabo otros dos experimentos con modelos distintos que no resultan convenientes ni desde el punto de vista teórico ni desde el punto de vista de resultados.

El primer de estos dos experimentos, considera que los valores de la velocidad angular de la IMU son entradas de control al proceso. Esto, desde un punto de vista teórico, no es correcto debido a que el proceso del movimiento de un segmento corporal por una persona no es un sistema controlable, por lo cual es inapropiado considerar cualquier variable de salida de un controlador.

El segundo de los modelos considera que el valor del muestreo de la cámara se mantiene constante mientras no se toma otro valor. Este modelo no es correcto debido a que, realmente, el brazo se está moviendo y la cámara no está tomando ningún dato, es decir, el valor del ángulo que debería leer la cámara no es el mismo durante 5ms.

5. Resultados

En el presente capítulo se exponen todos los resultados obtenidos a partir de las hipótesis teóricas y la metodología seguida que se han expuesto anteriormente. Además, dichos resultados serán analizados y expuestos de la forma más visual posible.

5.1. Resultados con el LEGO Mindstorm NXT

5.1.1. Estudio de la inicialización de las matrices características del KF

Con el fin de cumplir el objetivo de estudiar el comportamiento de las matrices estadísticas (P, Q y R) del filtro de Kalman mediante su inicialización, en este apartado se realizan 10 experimentos modificando diversos valores de las matrices según muestra la tabla 1. Los resultados obtenidos son los expuestos en la figura 19.

En primer lugar, hay que destacar que la referencia es la velocidad a la que queremos llegar en el régimen permanente, el ensayo 0 es la velocidad del motor de LEGO sin la aplicación del KF y el ensayo 1 está sobre el eje de abscisas debido a que la estimación del KF era nula.

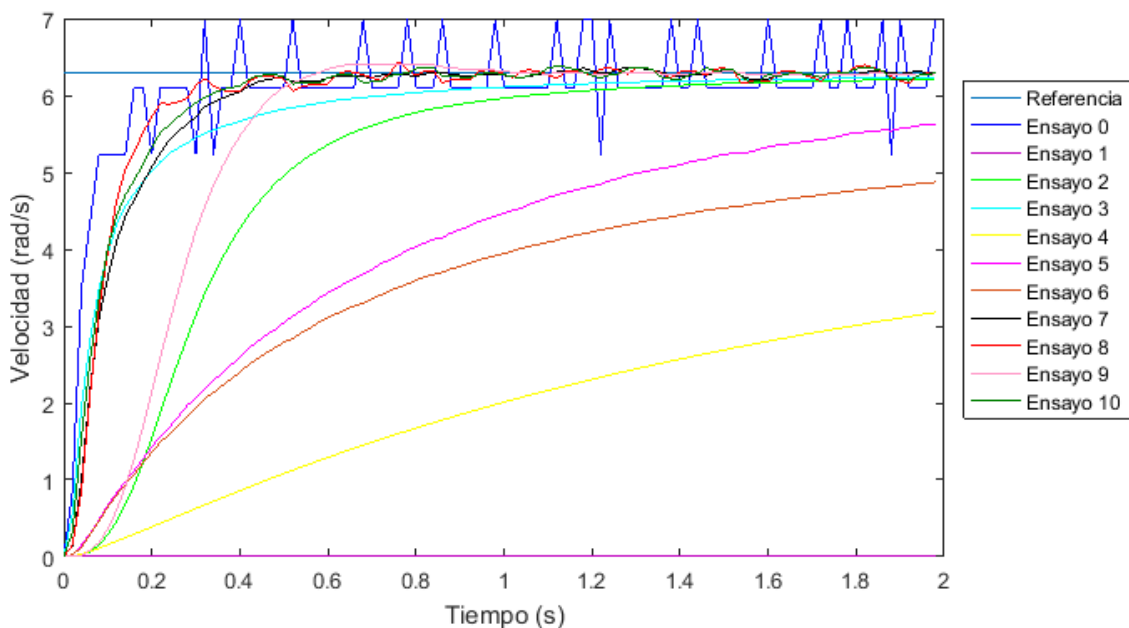


Figura 19. Ensayos para la inicialización de las matrices estadísticas

En segundo lugar, se estudia cómo afecta a la estimación de la velocidad del motor la inicialización de la matriz de covarianza del error. Se puede observar que conforme aumentan los valores de la diagonal de la matriz (ensayos 2 y 3), el tiempo de establecimiento y el error de posición se ven notablemente reducidos. Sin embargo, al introducir valores fuera de la diagonal, ambas variables empeoran. Debido a que la matriz de covarianzas del error se define como una matriz diagonal positiva, todos los resultados obtenidos hasta ahora son correctos.

Por otro lado, esto nos obliga a que la matriz del ruido del proceso (Q) sea diagonal ya que se suma a P_0 y genera el mismo efecto que al introducir valores fuera de la diagonal en P_0 .

Una vez conocido el comportamiento de P_0 , se pasa a estudiar cómo afecta al sistema la matriz de covarianzas del ruido del proceso. En este caso hacen falta cuatro ensayos (ensayos 5, 6, 7 y 8). En el ensayo 5, donde las dos entradas diagonales de la matriz son valores no nulos, se observa que la estimación obtenida es peor a la de los ensayos 2 y 3 ya que el error de posición y el tiempo de establecimiento aumentan. En el ensayo 6, donde se le da valor no nulo únicamente a la entrada Q_{11} , la estimación empeora en comparación con el ensayo 5 debido a que el error de posición todavía aumenta más. Así pues, solo cabe una última posibilidad y es que la entrada Q_{22} sea no nula (ensayo 7). Como puede verse en la figura 19, el ensayo 7 reduce el tiempo de establecimiento y el error de posición, además de que aparece una pequeña oscilación que aproxima más la estimación a los valores reales de la velocidad del motor (ensayo 0). En el ensayo 8, al aumentar el valor de la entrada Q_{22} , se reduce más el tiempo de establecimiento y aumenta la oscilación en el transitorio. Con todo esto el comportamiento de la matriz de covarianzas del ruido del proceso queda definido. En este sistema tienen que ser todas las entradas nulas excepto la entrada Q_{22} para que la estimación sea correcta. Esto tiene sentido ya que el objetivo es mejorar la medida de los sensores por lo que se tiene poco en cuenta el valor del modelo de la posición que será similar al valor medido por los encoders y se tiene más en cuenta el valor estimado por el modelo de la velocidad.

Finalmente queda estudiar cómo se comporta la matriz de covarianzas del ruido de la instrumentación. Debido a que en estos ensayos no es una matriz sino un valor numérico, solo hay que aumentarlo para ver el comportamiento del sistema ante variaciones de dicho valor. Además, se puede partir de la hipótesis que, al aumentar el valor de R , el filtro tiene más en cuenta los valores de las medidas de los encoders y el modelo empeora. En el ensayo 9 se observa que, al aumentar el valor de R , pese a que el error de posición se mantiene, la oscilación desaparece y el tiempo de establecimiento aumenta. Así pues, valores pequeños de R mejoran la estimación de la velocidad del motor.

En definitiva, para la inicialización de las matrices estadísticas del KF hay que tener en cuenta tres cosas. Las matrices P_0 y Q deben ser matrices diagonales. Además, dichas matrices deben tener valores grandes en la diagonal en el caso de P_0 para que el modelo pueda converger a los valores reales y en la entrada Q_{22} , en el caso de la matriz de covarianzas del ruido del proceso, para reducir el tiempo de establecimiento y el error de posición, además de generar una pequeña oscilación tanto en régimen permanente como en régimen transitorio. Por último, el valor de la matriz R debe ser pequeño para que no elimine los efectos de P_0 y Q , debido a que el filtro tiene menos en cuenta los valores de la medición.

5.1.2. El filtro de Kalman para mejorar la medida de los encoders

Con la finalidad de cumplir el objetivo de mejorar la trayectoria circular de un robot móvil con configuración diferencial en las dos ruedas mediante la estimación de la velocidad, en el presente apartado se aplica el KF para la estimación de las velocidades angulares del motor izquierdo y derecho del robot móvil. Para poder evaluar los resultados obtenidos, se compararán los valores estimados por el KF y los valores obtenidos mediante la estimación

simple de tres variables: la trayectoria descrita por el robot, la velocidad angular de los dos motores y las acciones de control que se llevan a cabo en cada motor.

Asimismo, como se mencionó previamente, la mejora de las medidas de los encoders se lleva a cabo mediante la ejecución de una trayectoria circular de 335mm por el LEGO Mindstorm NXT aplicando el KF y después se comparan los resultados obtenidos para las tres variables analizadas con los resultados que se obtuvieron en la asignatura de LAC para ver si el filtro mejora el comportamiento del sistema.

Trayectoria

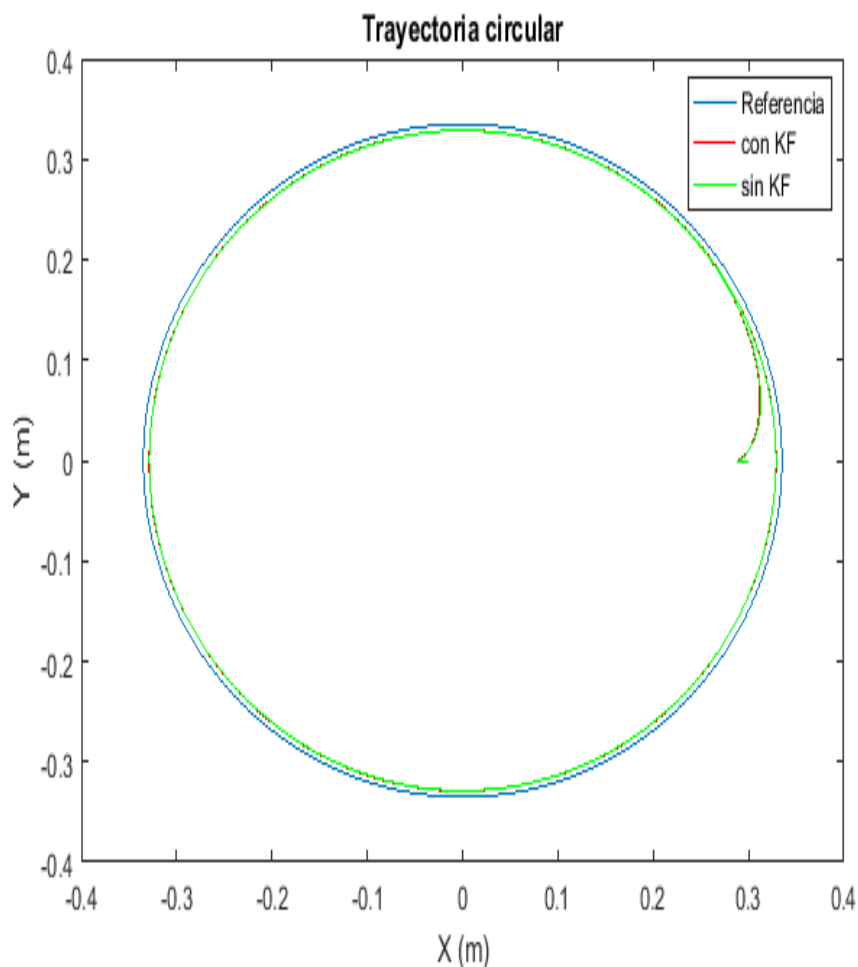


Figura 20. Trayectorias circulares descritas por el robot móvil

En la comparación de esta primera variable no se observa una clara ventaja en la aplicación del KF a la estimación de trayectorias de robots o vehículos móviles. Esto se puede ver en la figura 20 donde se comparan la referencia, la trayectoria descrita por el LEGO Mindstorm NXT sin el KF y la trayectoria obtenida al aplicar el filtro con la misma plataforma.

Se observa que ambas trayectorias están prácticamente sobrepuestas, es decir, que son iguales. Ahora bien, si aumentamos la imagen para ver más de cerca las trazadas, sí que se observa una pequeña mejora en la circunferencia descrita por el robot con la aplicación del KF debido a que la traza roja está más próxima a la referencia. Esto puede ser debido a que la

estimación de la velocidad angular mediante el KF es mejor que la obtenida por medio de la lectura de los encoders.

En la figura 21 se muestran las dos zonas donde se ha realizado el zoom. Estas zonas son cuando el robot ha dado media vuelta, zoom de la parte izquierda de la figura 21, y cuando el robot ya ha dado la vuelta completa, zoom de la parte derecha de la circunferencia de la figura 21. Cabe destacar que, pese a que solo se muestran dos zonas, a lo largo de toda la circunferencia, la línea roja correspondiente a la trazada del robot cuando se le aplica el filtro de Kalman (línea roja) está más próxima a la trayectoria de referencia (línea azul) que la trazada del robot cuando no se le aplica el KF (línea verde).

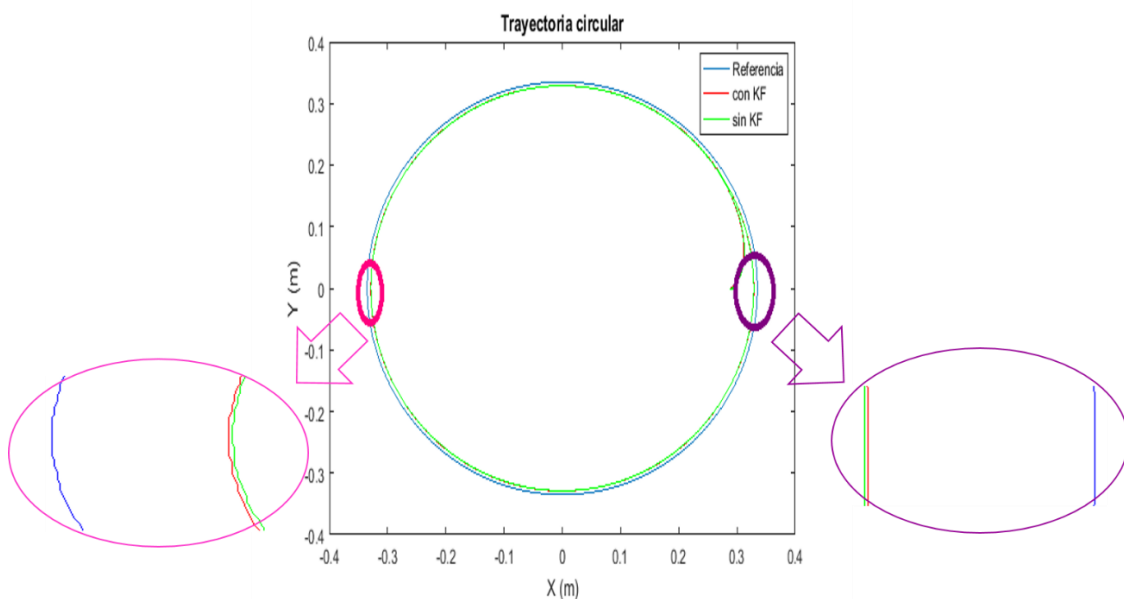


Figura 21. Zoom sobre las trayectorias

Velocidad angular

En la figura 22 se presentan los resultados obtenidos con esta variable. Como puede observarse, con el filtro de Kalman se obtienen unos valores de velocidad angular con menos ruido tanto en la rueda derecha como en la izquierda. Esto puede apreciarse con mejor claridad en la zona aumentada de la gráfica. En dicha zona, las velocidades reales tienen unos picos de sobreoscilación con mayor amplitud que las ondas que aparecen en los valores estimados.

Además, la estimación de la velocidad angular puede calificarse de buena ya que el tiempo de establecimiento y el error de posición de las velocidades estimadas son semejantes a los de las velocidades reales. Sin embargo, pese a que la estimación es buena, existen muchas sobreoscilaciones lo que lleva a que la trazada del robot no sea mucho mejor con el filtro que sin él.

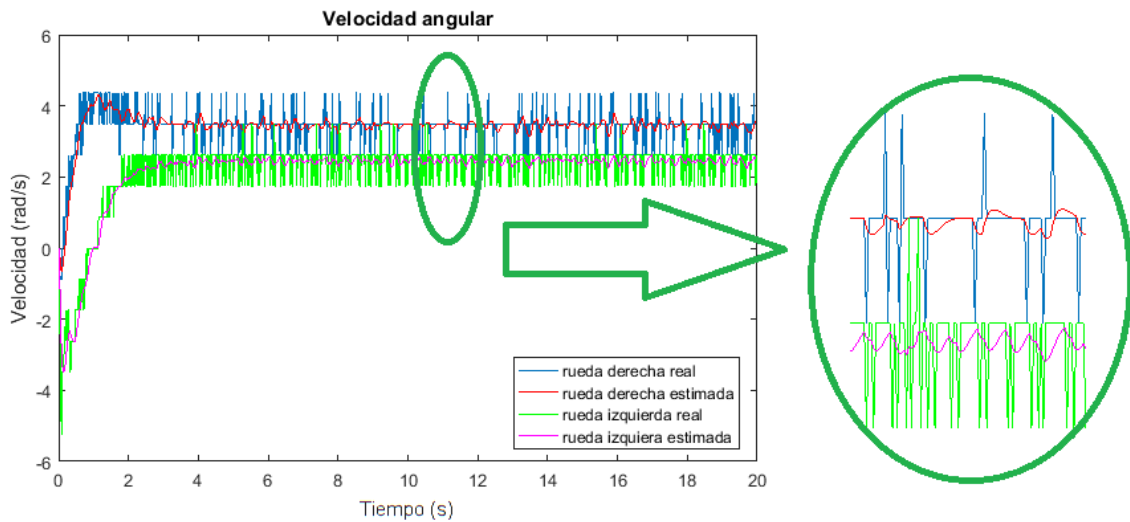


Figura 22. Velocidad angular de las ruedas izquierda y derecha

En definitiva, la variable de la velocidad angular del robot o vehículo móvil sí que se mejora en el caso de la aplicación del filtro de Kalman para su estimación ya que se reduce el ruido de los motores de las ruedas, se mantienen tanto el error de posición como el tiempo de establecimiento del sistema constantes y, aunque sigue habiendo sobreoscilaciones, el valor medio de estas está más centrado en torno a la posición del régimen permanente de la velocidad angular.

Acción de control

Finalmente, la variable de la acción de control presenta unos resultados muy similares a los obtenidos con la velocidad angular. Dichos resultados se presentan en la figura 23.

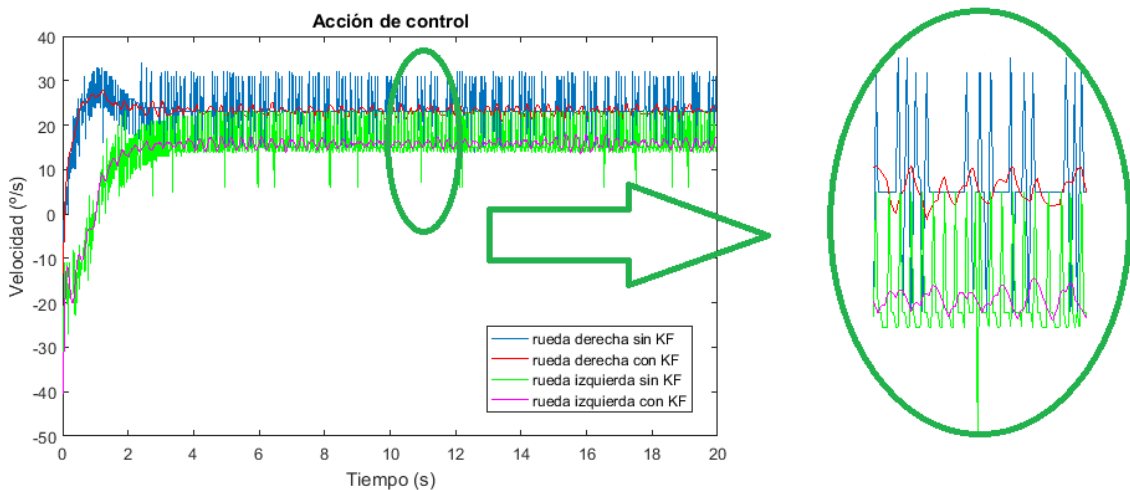


Figura 23. Acción de control sobre los motores izquierdo y derecho

Como puede apreciarse, los motores reciben señales menos ruidosas, lo que supone un aumento de la vida de los componentes electrónicos que forman el motor debido a la aplicación del filtro de Kalman. Además, únicamente mediante la toma de datos de la posición angular de las ruedas, se puede estimar la velocidad angular de las mismas por lo que no se requiere de instrumentación que mida esta variable

5.1.3. Estimación de la velocidad lineal y angular del LEGO Mindstorm NXT

Con la finalidad de llevar a cabo la fusión sensorial en un robot móvil, en este apartado se fusionan los datos de los encoders de los motores, que se emplean para calcular mediante una estimación simple la velocidad lineal y angular del robot, con los datos de los acelerómetros de la marca Hitechnic, que se emplean para calcular la aceleración lineal y angular del robot también mediante una estimación simple.

Como se puede observar en la figura 24, los resultados obtenidos con los valores de las matrices de covarianza empleadas para los casos anteriores ajustan totalmente la estimación de la velocidad lineal y angular del robot mediante el filtro de Kalman a estas mismas velocidad obtenida sin aplicarlo y calculándolas mediante estimaciones simples. Esto no es una buena solución ya que implementado el KF estamos aumentando el tiempo de computación y no generamos ninguna mejora en el comportamiento el sistema, simplemente lo estaríamos igualando.

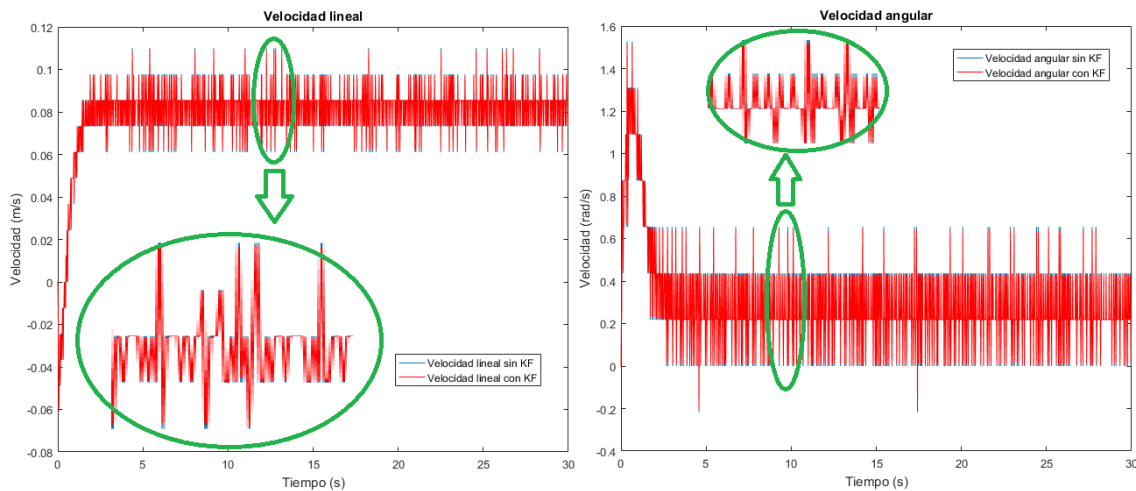


Figura 24. Velocidad lineal y angular prueba 1

Así pues, se van modificando los valores de las matrices P_0 , Q y R según muestra la tabla 2 en diferentes ensayos hasta que se obtienen los resultados mostrados en las figuras 25 y 26. Por un lado, en la figura 25, se muestra la estimación de la velocidad lineal del robot en rojo y la velocidad lineal del robot sin aplicar el filtro de Kalman en azul. Como se puede observar en la zona aumentada y remarcada en verde, en la estimación de la velocidad lineal mediante la aplicación del filtro de Kalman se reduce la amplitud de la oscilación en comparación con el cálculo de la misma variable mediante aproximaciones simples.

Por otro lado, en la figura 26 se muestra la estimación de la velocidad angular del robot en rojo y la velocidad angular del robot sin aplicar el KF en azul. En la zona remarcada en verde se puede observar la reducción de la amplitud de la señal estimada mediante el filtro de Kalman respecto de la señal calculada mediante estimaciones simples.

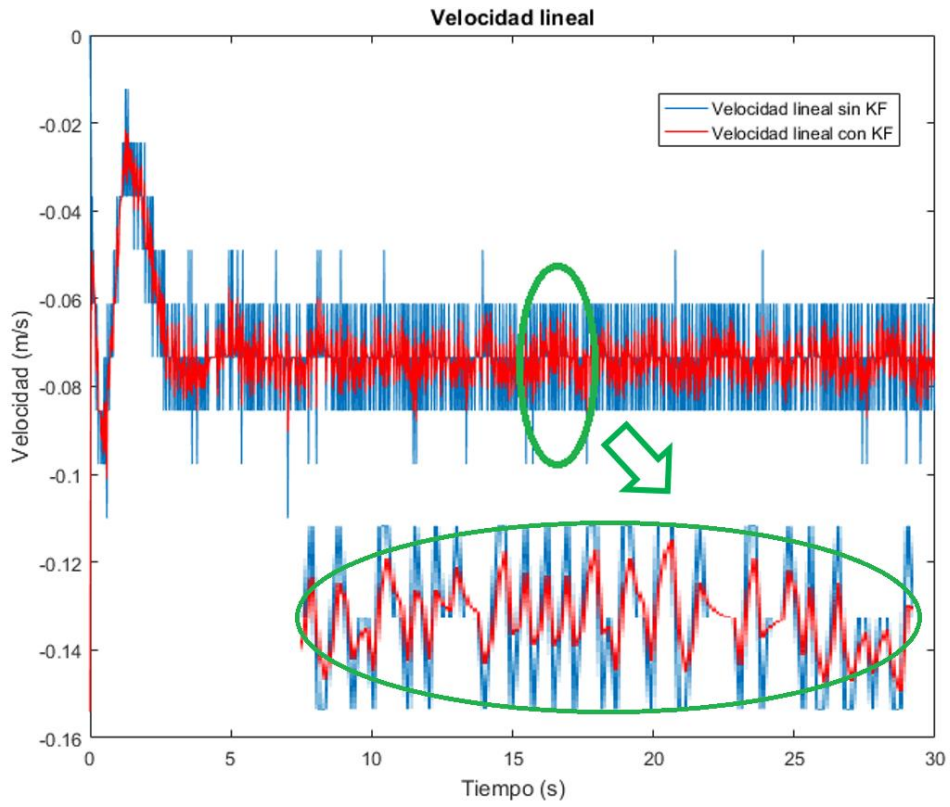


Figura 25. Velocidad lineal prueba 10

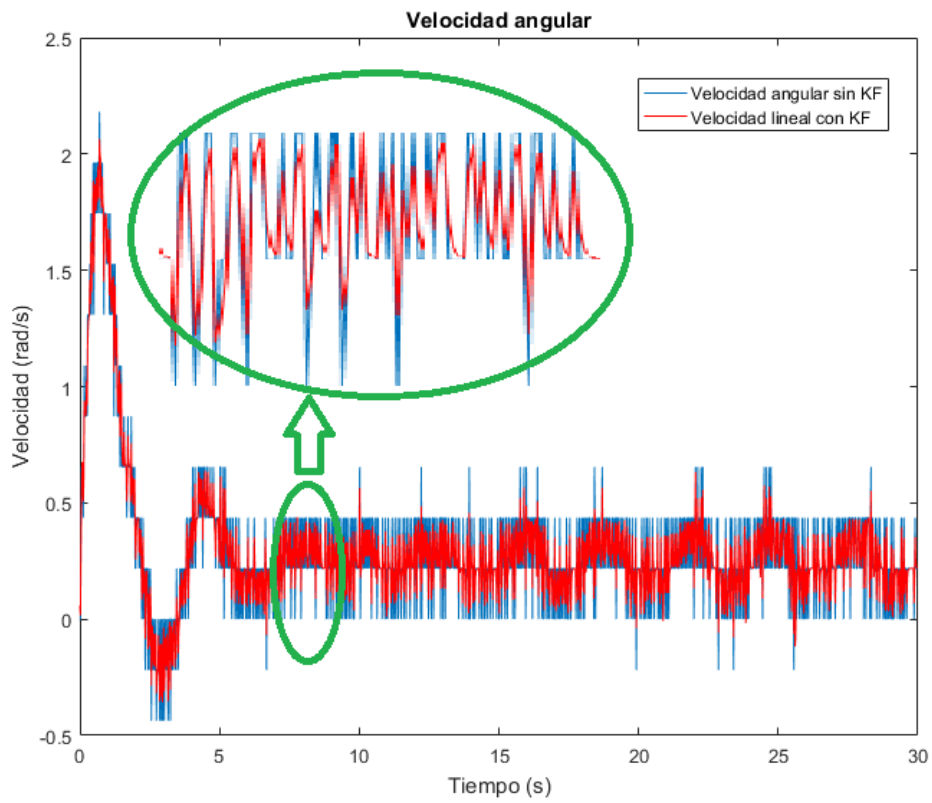


Figura 26. Velocidad angular prueba 10

Con este ensayo sí que tiene sentido aplicar el filtro de Kalman para estimar tanto la velocidad lineal como la velocidad angular de un robot móvil ya que se conseguirá tener una velocidad más similar a la de referencia, la cual no tiene ruidos, y lograr una trayectoria más precisa.

5.2. Resultados con el brazalet

Durante la ejecución del programa implementado para esta parte se realizan distintas pruebas para ajustar lo mejor posible los valores de las entradas de las matrices estadísticas del KF al comportamiento de este sistema. En este caso es necesario ajustar con mucha precisión los valores de las entradas de la matriz de covarianza del ruido de la medición tanto para estimar bien la posición del marcador, dándole mayor importancia a los valores muestreados por la cámara, como para eliminar los outliers de la misma cuando no muestrea. Al final del apartado también se modifica el valor de la matriz de covarianza del ruido del proceso para ver cómo afecta al comportamiento del sistema.

Tal y como se puede observar en la figura 13, los outliers son mucho más atípicos en la tercera coordenada (coordenada Z) que en las otras dos. Por ello, el valor de la entrada de la matriz R correspondiente a los datos de la cámara en la tercera coordenada (valor denotado por Z) debe ser mucho mayor que los valores X e Y para que el filtro considere más las mediciones de la cámara en las coordenadas de X e Y que las mediciones de la coordenada de Z. Los resultados para el ensayo 1 de la tabla 3 se muestran en la figura 27, en esta figura únicamente se representan los valores de la posición para tener una mejor percepción de los resultados. Asimismo, también se representan los valores estimados para la posición angular de la cámara aplicando el filtro de Kalman y los valores de “ground truth” facilitados por los compañeros del ai2. Cabe destacar que los datos de “ground truth” no han sido tomados generando que la cámara muestree a lo largo de todo el movimiento del brazo y por tanto se pueden definir como los datos del movimiento real del brazo. Sin embargo, sí que sirven para saber si la estimación se está aproximando a los valores próximos a los reales.

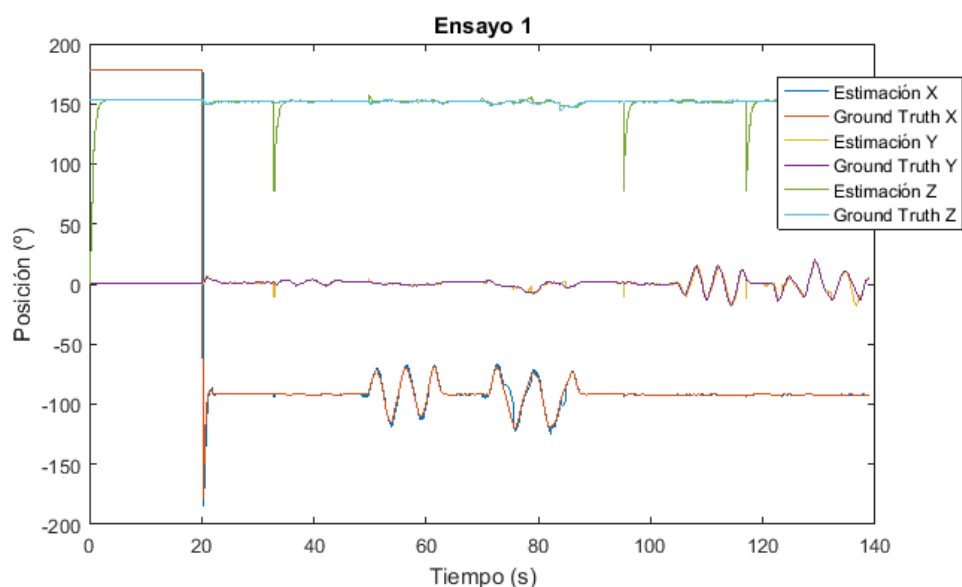


Figura 27. Ensayo 1 con el brazalet

Como se puede observar, los valores de los outliers se ven reducidos considerablemente pese a que no se eliminan del todo. Además, la posición de la cámara estimada y la posición real (“ground truth”) están sobrepuestas a lo largo de todo el movimiento excepto en las zonas donde la cámara no muestrea, ver el eje X de la figura 13.

En el ensayo 2 donde se aumenta el valor de Y y se reduce el valor de Z, cabe esperar que los outliers de la estimación en Y se vean reducidos ya que tenemos menos en cuenta los valores tomados por la cámara a la hora de hacer la estimación mientras que los valores de los outliers de la coordenada correspondiente a Z sean mucho mayores ya que reduciendo el valor de Z tomamos más en cuenta los valores medidos por la cámara a la hora de hacer la estimación. En la figura 28 se pueden ver los resultados obtenidos.

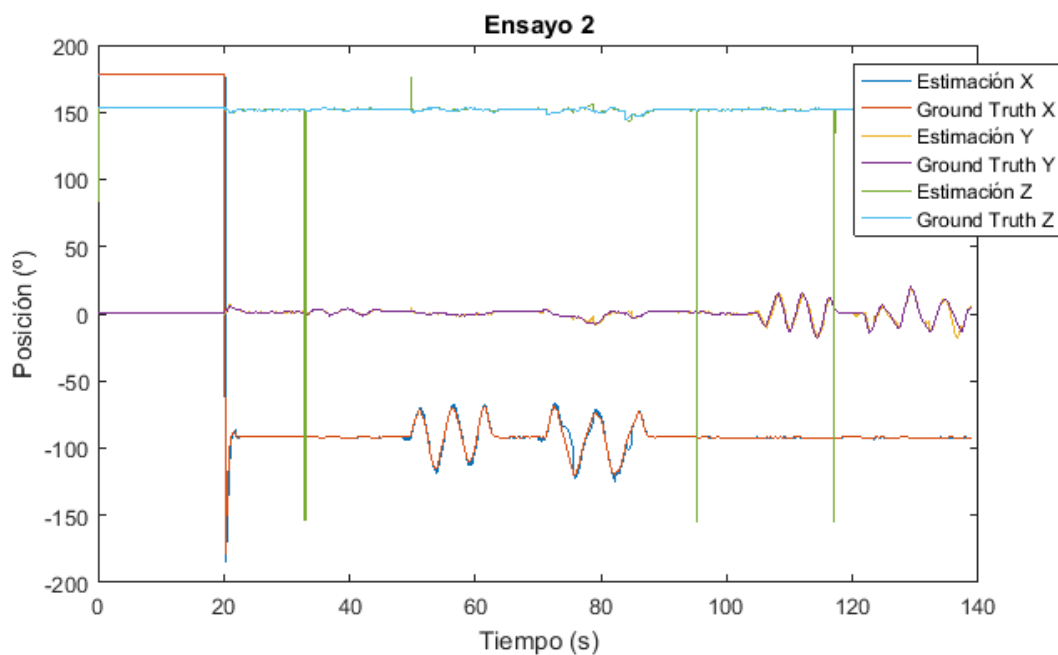


Figura 28. Ensayo 2 con el brazaletes

De acuerdo a la hipótesis hecha previamente, los valores de los outliers en la coordenada de Z aumentan, por lo que hay que darle valores a Z lo más grandes posibles. En cuanto a los valores de la posición en Y, se observa una reducción considerable de los valores de los outliers mostrados en la figura 27 por lo que hay que darle valores grandes a Y pero no tan grandes como a Z.

Una vez estudiada la eliminación de los outliers se pasa a estudiar la estimación de las velocidades angulares. Para ello se, en el ensayo 3, se utilizan los mismos valores que en el ensayo 1.

En el ensayo 3 se parte de la hipótesis de que debido a que el valor de S será muy grande, el estimador debe reducir los ruidos de la señal del dispositivo IMU ya que tiene menos en cuenta el valor de la medición hecha por este sensor a la hora de hacer la estimación. En la figura 29 se muestran los resultados para la coordenada X del IMU.

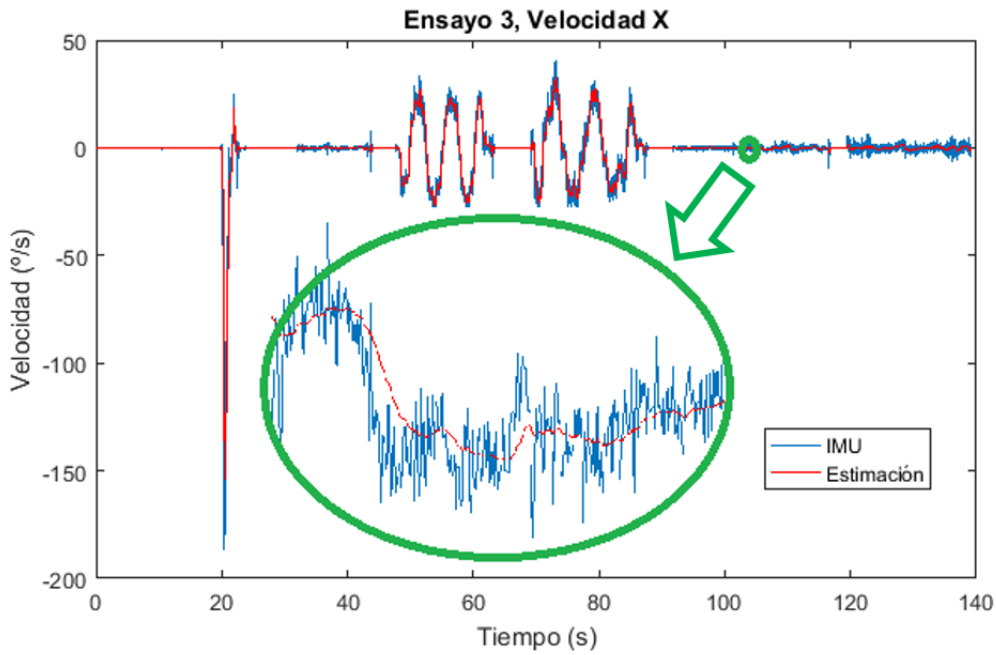


Figura 29. Ensayo 3 con el brazalete, coordenada X

En la figura 29 se amplía una zona de la señal para que se pueda apreciar mejor que la estimación, obtenida fusionando los datos de la cámara y del IMU mediante el KF, elimina gran parte de los ruidos de la señal original del IMU. En las figuras 30 y 31 se muestran los resultados obtenidos para la estimación de la velocidad angular en las coordenadas Y-Z mediante la fusión de los datos de los dos sensores. También se hace un zoom a las señales para poder apreciar mejor la eliminación de los ruidos de la señal original del dispositivo IMU.

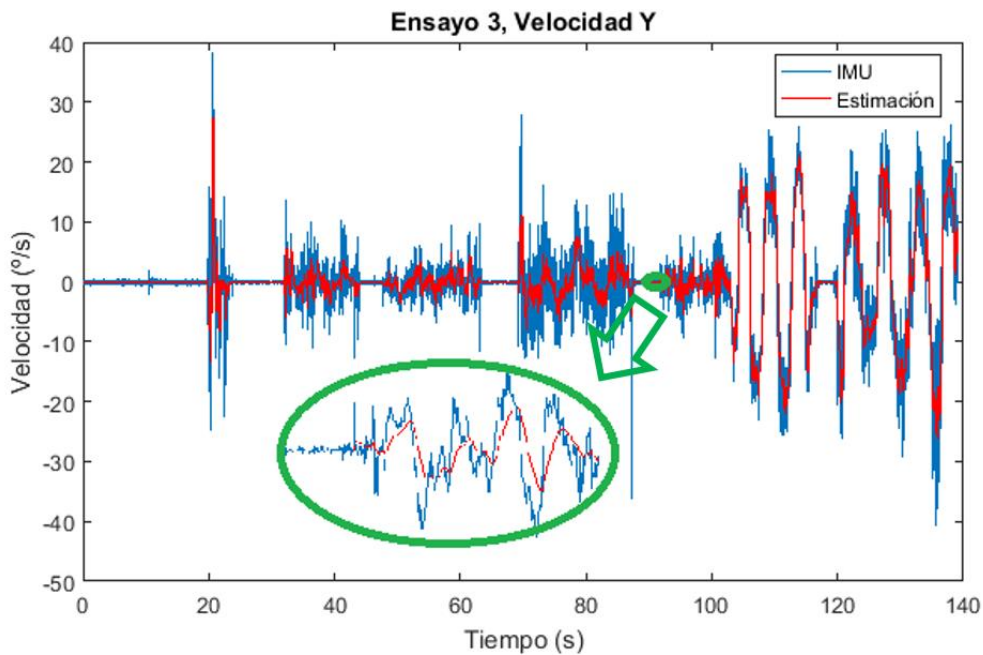


Figura 30. Ensayo 3 con el brazalete, coordenada Y

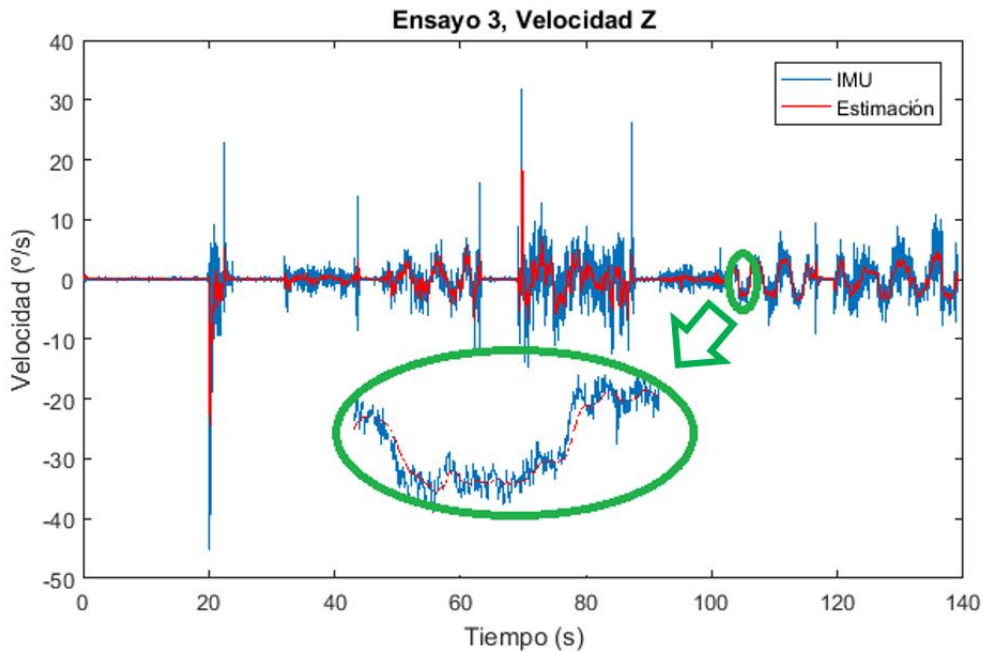


Figura 31. Ensayo 3 con el brazalete, coordenada Z

En el ensayo 4 se reduce el valor de S , correspondiente a las entradas del IMU, con la finalidad de ver que los ruidos de la señal del IMU en la estimación no se eliminan. En las figuras 32, 33 y 34 se pueden observar los resultados obtenidos para las coordenadas X, Y, Z, respectivamente. En las tres figuras se hace un zoom sobre una zona de las dos señales para ver más de cerca los resultados. En estas zonas aumentadas se puede observar que el valor estimado mediante la fusión de los datos de los sensores de la cámara y el IMU tiene una menor amplitud en la oscilación pero, comparando los resultados con los obtenidos en la figura 29, esta oscilación se puede reducir mucho más.

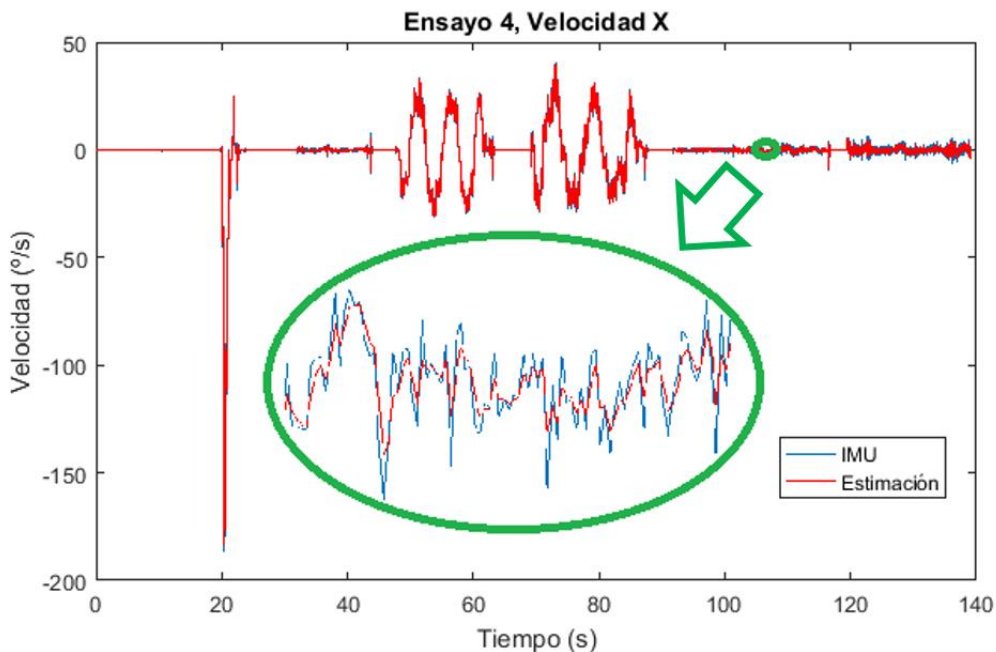


Figura 32. Ensayo 4 con el brazalete, coordenada X

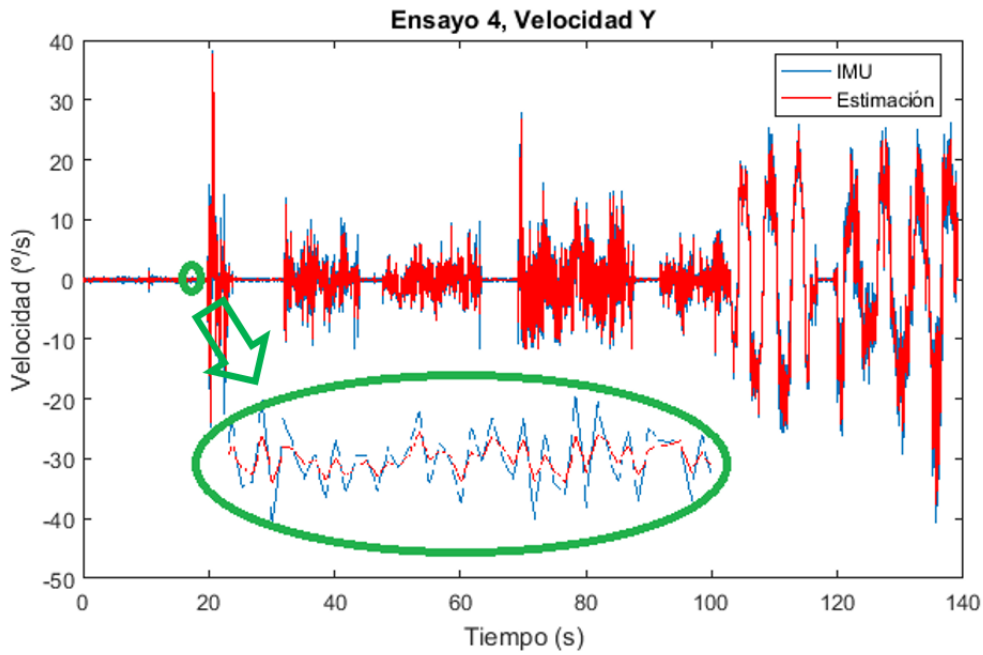


Figura 33. Ensayo 4 con el brazalete, coordenada Y

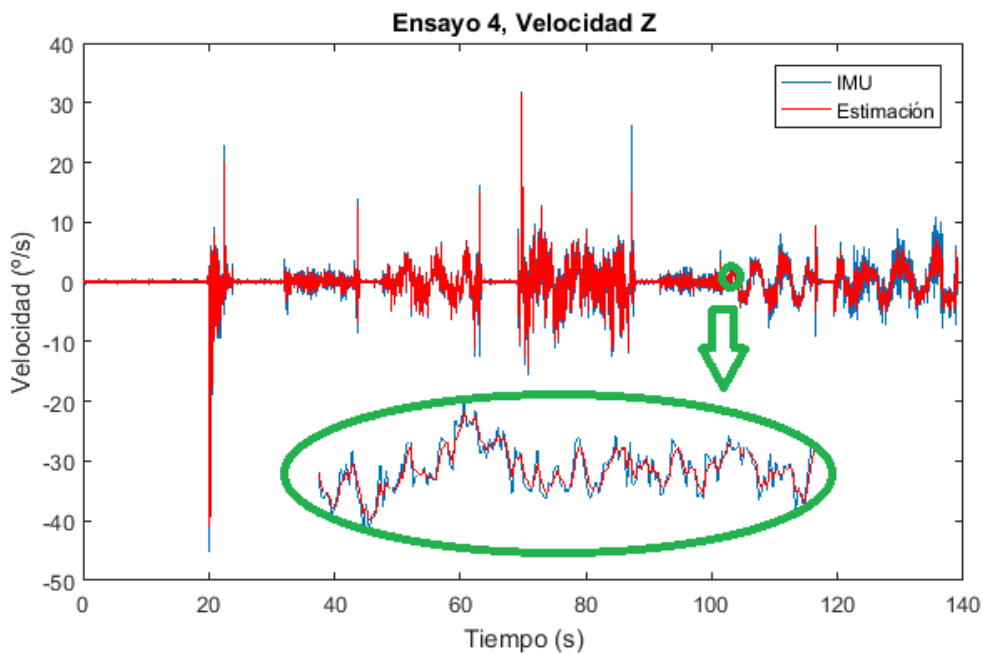


Figura 34. Ensayo 4 con el brazalete, coordenada Z

Cabe destacar también que hay ciertos valores que pueden ser outliers de la toma de datos hecha por el IMU y que con los valores del ensayo 4 estos outliers no se ven tan reducidos como ocurre con los valores de la matriz R del ensayo3.

Por todo ello, la hipótesis de partida en la cual se afirmaba que el valor de S tenía que ser lo más grande posible para que los valores del IMU fueran menos tenidos en cuenta en la estimación de las velocidades es cierta.

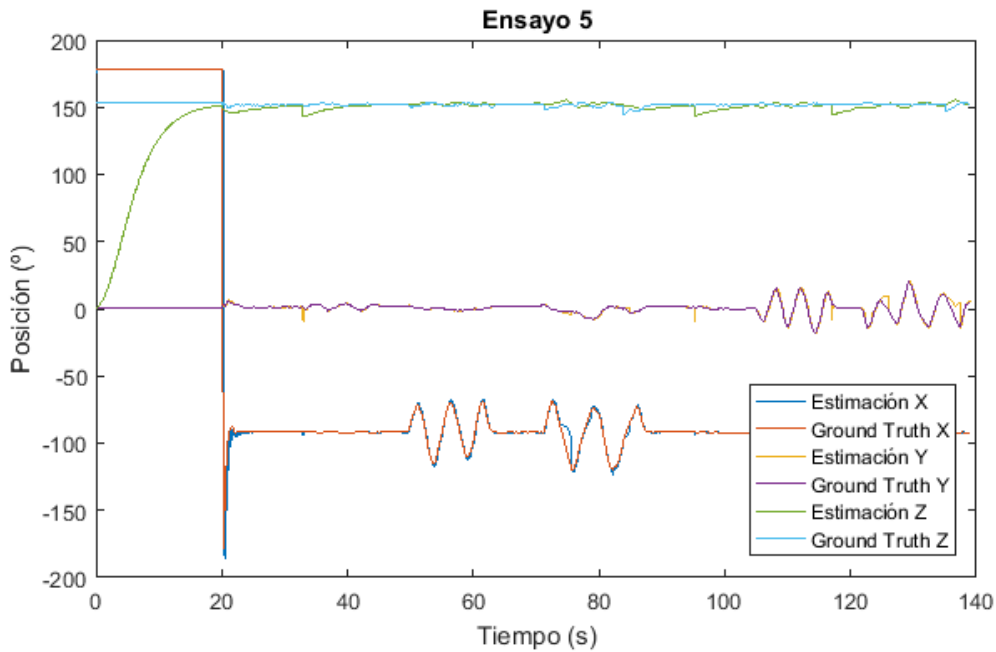


Figura 35. Ensayo 5 con el brazaletes, posición

Finalmente, en la figura 35 se muestran los resultados del ensayo 5 para la estimación de la posición angular del segmento corporal. En este caso cada coordenada se comporta de una manera diferente. La coordenada X mejora la estimación de la posición ya que se aproxima más a la señal de “ground truth”. La coordenada Y empeora en la estimación de ciertos puntos y además no se eliminan los outliers. Por último, la coordenada Z genera una peor estimación a pesar de que se eliminan prácticamente del todo los outliers. En el caso de los resultados obtenidos para las velocidades angulares en las tres coordenadas del espacio con los valores de las matrices estadísticas del filtro de Kalman empleados en el ensayo 5, se obtienen los resultados de las imágenes 36, 37 y 38.

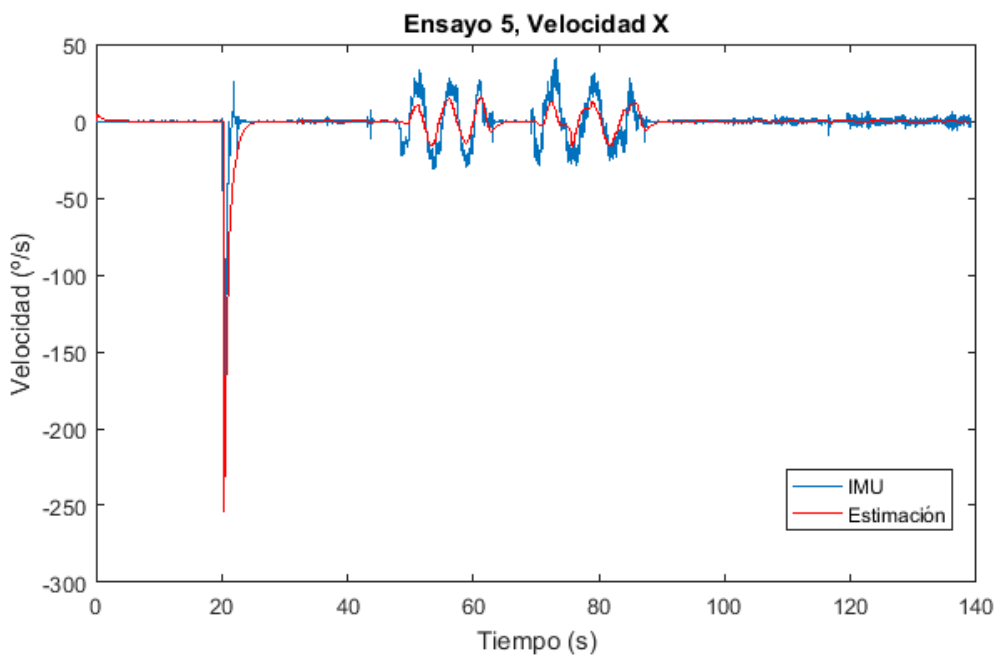


Figura 36. Ensayo 5 con el brazaletes, velocidad X

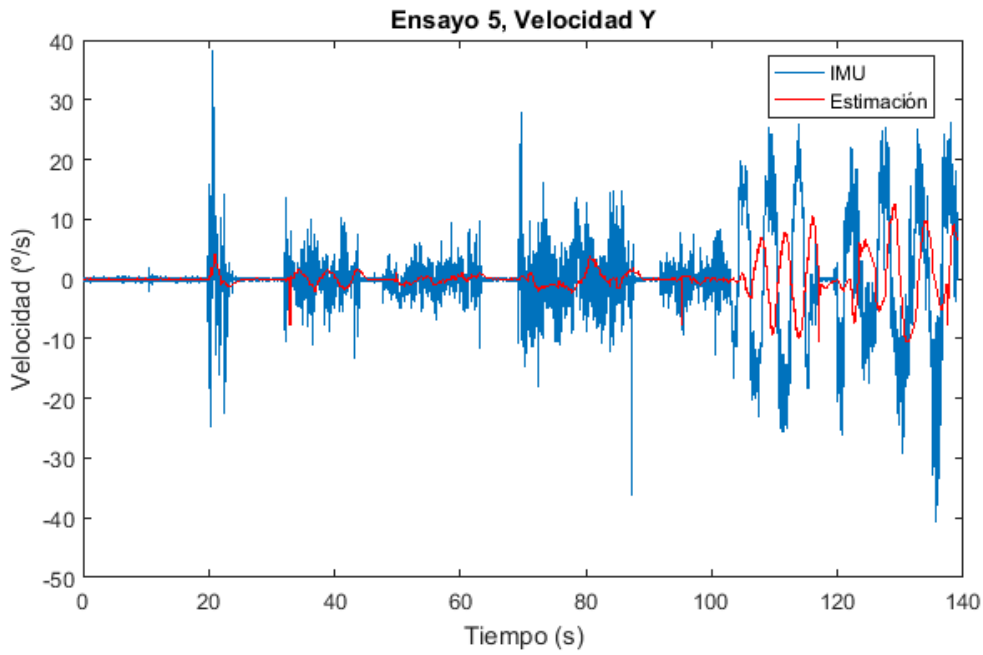


Figura 37. Ensayo 5 con el brazaletes, velocidad Y

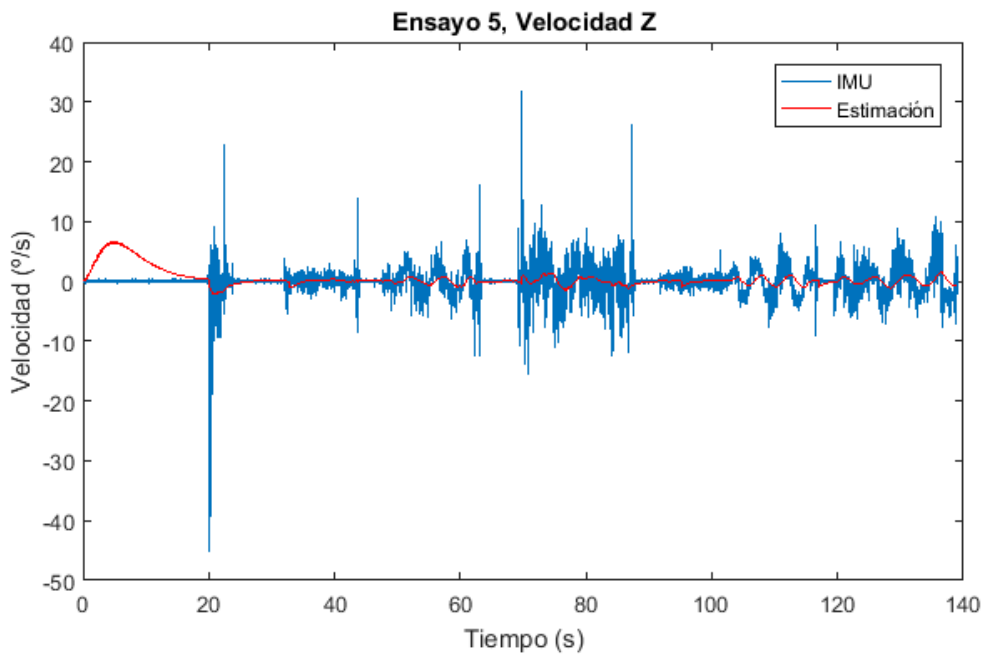


Figura 38. Ensayo 5 con el brazaletes, velocidad Z

Como puede observarse en todas las gráficas, la estimación de la velocidad no llega a tener los valores de amplitud necesarios cuando el brazo gira. Así pues, los resultados obtenidos en el ensayo 3 para la velocidad son más acordes al comportamiento del sistema.

En definitiva, los valores que se deben escoger ya que son los más acordes al comportamiento del movimiento del brazo son los empleados en el ensayo 1 y 3.

6. Conclusiones

Para establecer unas conclusiones se hace un repaso a los objetivos planteados y se comprueba si se han logrado cumplir:

- El primer objetivo se ha logrado mediante la ligera mejora de la traza circular de un robot móvil mediante la implementación del filtro de Kalman. También se ha logrado reducir considerablemente los ruidos de la señal original de la velocidad angular y de las acciones de control efectuadas por los motores. De esta forma, el filtro de Kalman, que es un estimador de estado, no solo mejora las variables de estado reales sino que preserva los componentes electrónicos del sistema mediante la reducción de los ruidos existentes. Este objetivo también se ha visto cumplido en la estimación de las velocidades lineal y angular de un robot móvil ya que también se han eliminado los ruidos de las variables de estado. Otro de los beneficios aportados por un estimador de estado es que no requiere de una gran cantidad de sensores para poder obtener una gran variedad de variables, es decir, mediante la toma de datos de la posición del motor (encoders) se ha conseguido estimar la velocidad.
- El estudio del filtro de Kalman se ha llevado a cabo mediante la inicialización de las matrices: de la covarianza del error de medición P , la cual se ha observado que a medida que se hace más grande el error de estimación se reduce y la variable estimada tiende a ser igual que la variable real; de la covarianza del ruido del proceso, la cual evalúa los ruidos aportados por las variables de control a la predicción de la estimación en la primera ecuación del esquema de cálculo del KF; de la covarianza del ruido de los sensores, la cual aporta mayor o menor importancia a los sensores empleados a la hora de hacer la corrección de la estimación del estado.
- Finalmente se pretendía fusionar los datos de varios sensores mediante la aplicación del filtro de Kalman. Este objetivo se ha cumplido ya que se ha implementado un modelo que unía los datos de los encoders de los motores de las ruedas con los de los acelerómetros de la marca Hitechnic para lograr una estimación de las velocidades del motor.

Así pues, una vez cumplidos estos objetivos se estuvo preparado para afrontar el objetivo principal que era estimar la posición de un segmento corporal de una persona mediante los datos de una cámara de visión artificial y de un Sistema Inercial de Medición (IMU).

Para evaluar el cumplimiento del objetivo principal, se evalúa si se han cumplido los tres objetivos menores que se plantearon en el capítulo de objetivos.

- Definir correctamente las matrices estadísticas del filtro de Kalman para que el sistema converja. Este objetivo se ha cumplido analizando el comportamiento del sistema ante los cambios correspondientes en las matrices estadísticas del filtro de Kalman. Además, al final de los resultados se ha concluido con los valores que mejor resultado han dado ante el sistema.
- Definir la trayectoria del segmento corporal mediante la estimación de la posición y de la velocidad. En el ensayo 1 (figura 27) se han obtenido las estimaciones de la

trayectoria descrita por el brazo en cada una de las coordenadas del espacio y en el ensayo 3 (figuras 29, 30 y 31) se han obtenido las estimaciones de las velocidades angulares del brazo para cada una de las coordenadas X-Y-Z.

- Dotar de mayor importancia a los datos procedentes de la cámara cuando esta detecte la posición del segmento corporal y de menor importancia cuando no sea así. Este objetivo se ha visto cumplido mediante la correcta definición de R, es decir, en el apartado de modelización.

Por otro lado, pese a que los objetivos de este TFG se han cumplido, hay una serie de motivaciones que llevarán al alumno a seguir trabajando en el desarrollo de este proyecto.

Dichas motivaciones son:

- Llevar a cabo nuevos ensayos con el brazalete donde se mida la posición de la cámara en todo momento, a continuación eliminar ciertos valores de la posición de la cámara y aplicar el filtro de Kalman para estimar esos valores eliminados. De esta forma se podrá ver si las estimaciones se aproximan a los valores reales de la posición.
- Añadir al filtro de Kalman los valores de un acelerómetro que lleva incluido el dispositivo IMU para mejorar las estimaciones de la posición y velocidad de la cámara.
- Estimar con el filtro de Kalman, además de las variables ya estimadas, la posición y velocidad lineal del segmento corporal para tener una información muy precisa del movimiento.
- Implementar e ejecutar el código del KF para su cálculo online.

Todo esto supondrá poder aplicar de manera experimental con pacientes la tecnología wearable.

Referencias

- [1] Leonardo J. Marín Paniaguda. Localización de robots móviles de recursos limitados basada en fusión sensorial por eventos. Tesis doctoral, UPV, abril, 2014.
- [2] Ángel Valera, Marina Vallés, Leonardo Marín, Pedro Albertos (2011). Design and Implementation of Kalman Filters applied to LEGO NXT based Robots.
- [3] Sherman MA., et al. (2011). Simbody: multibody dynamics for biomedical research. IUTAM 2:241–261.
- [4] [2] Hatze H. (2002) The fundamental problem of myoskeletal inverse dynamics and its implications. *Journal of Biomechanics*, 5, 109-115.
- [5] Page A., et al. (2011). Representation of planar motion of complex joints by means of rolling pairs. Application to neck motion. *Journal of Biomechanics*, 44, 747-750.
- [6] Yang J., et al. (2010). Review of biomechanical models for human shoulder complex. *Int. Human Factors Modelling and Simulation*, 1,271-293.
- [7] Leardini A., et al. (2005) Human movement analysis using stereophotogrammetry. part 3. soft tissue artefact assessment and compensation. *Gait Posture*, 21, 212-25.
- [8] Peters, A., Galna, B., Sangeux, M., Morris, M., Baker, R. (2010) Quantification of soft tissue artifact in lower limb human motion analysis: A systematic review. *Gait Posture*, 31, 1-8.
- [9] Gunther M., et al. (2003) Dealing with skin motion and wobbling masses in inverse dynamics. *Journal Mechanics in Medicine and Biology*, 3, 309-36.
- [10] Alonso, F., Del Castillo, J., Pintado, P. (2007) Motion data processing and wobbling mass modelling in the inverse dynamics of skeletal models. *Mechanism and Machine Theory*, 42, 1153-1169
- [11] Erdemir, A., McLean, S., Herzog, W., van den Bog, A.J. (2007). Model-based estimation of muscle forces exerted during movements. *Clinical Biomechanics*, 22, 131-154.
- [12] Anderson A., et al. (2007). Verification, validation and sensitivity studies in computational biomechanics. *Computer Methods in Biomechanics and Biomedical Engineering*, 10, 171-184
- [13] Nigg, B. M., Herzog, W. (1999). *Biomechanics of the musculo-skeletal system* (2nd ed.). Chichester: Wiley.
- [14] Riemer R, Hsiao-Wecksler ET, Zhang X. (2008) Uncertainties in inverse dynamics solutions: A comprehensive analysis and an application to gait. *Gait Posture*, 27(4), 578-588.
- [15] Deschamps, K., et al. (2011). Body of evidence supporting the clinical use of 3D multisegment foot models: A systematic review. *Gait and Posture*, 33(3), 338-349.

PRESUPUESTO

Presupuesto

En el presupuesto de este TFG hay que considerar que la parte relacionada con el LEGO Mindstorm NXT es meramente de estudio, a la cual no se le pretende dar una aplicación más haya que la del estudio que pueda hacer un alumno con ella. Aun así, se detalla la cantidad que ha podido suponer esta parte.

Por otro lado, la parte relacionada con el brazalete sí que tiene una aplicación, ya sea como proyecto de investigación o como proyecto que se utilice en algún centro médico. Por ello también se detalla el coste de esta parte.

Unidades de obra

Antes de entrar en análisis económicos, se lleva a cabo la definición de las dos unidades de obra que forman el proyecto. Estas dos unidades de obra ya han sido distinguidas previamente:

- Unidad de Obra 1 (UD1): Estudio del filtro de Kalman con la plataforma de LEGO Mindstorms NXT.
- Unidad de Obra 2 (UD2): Estudio del sistema de medida de la posición de un segmento corporal mediante tecnología wearable.

Lista de materiales

También es necesario conocer los materiales que se van a emplear en cada una de las unidades de obra. En el caso de la Unidad de Obra 1 los materiales empleados son:

- 1.1. LEGO Mindstorms NXT 8547-NXT 2.0 V24
- 1.2. RobotC-Licencia de centro v2.0
- 1.3. Sensor de aceleración Hitechnic para LEGO Mindstorms

En el caso de la Unidad de Obra 2 los materiales empleados son:

- 2.1. Bobina de impresión para el brazalete
- 2.2. BNO055 Intelligent 9-axis absolute orientation sensor (IMU)
- 2.3. S110 nRF51 Bluetooth® low energy Peripheral SoftDevice
- 2.4. MATLAB
- 2.5. Kit de cables
- 2.6. Cinta de goma elástica blanca

Amortización

Algunos de los materiales de las listas anteriores no se han empleado únicamente para este proyecto por lo que se procede a calcular su precio de amortización por hora asumiendo que la amortización se lleva a cabo en cinco años.

- LEGO Mindstorms NXT 8547-NXT 2.0 V24 (799'95€+14'90€ gastos de envío=814'85€)

$$\frac{814'85 \text{ €}}{5 \text{ años}} = 162'97 \frac{\text{€}}{\text{año}} \times \frac{1 \text{ año}}{365 \text{ días}} \times \frac{1 \text{ día}}{24 \text{ h}} = 0'02 \frac{\text{€}}{\text{h}}$$

- RobotC-Licencia de centro v2.0 (299€/unidad)

$$\frac{299 \text{ €}}{5 \text{ años}} = 59'8 \frac{\text{€}}{\text{año}} \times \frac{1 \text{ año}}{365 \text{ días}} \times \frac{1 \text{ día}}{24 \text{ h}} = 0'01 \frac{\text{€}}{\text{h}}$$

- MATLAB (2000€/unidad)

$$\frac{2000 \text{ €}}{5 \text{ años}} = 400 \frac{\text{€}}{\text{año}} \times \frac{1 \text{ año}}{365 \text{ días}} \times \frac{1 \text{ día}}{24 \text{ h}} = 0'05 \frac{\text{€}}{\text{h}}$$

Gastos de material

La siguiente tabla muestra los gastos de los materiales empleados:

Material	1.1	1.2	1.3	2.1	2.2	2.3	2.4	2.5	2.6
Precio	0'02 €/h	0'01 €/h	55 €/unidad	30 €/unidad	30 €/unidad	75 €/unidad	0'05 €/h	3'36 €/unidad	1'5 €/unidad

Gastos de personal

El personal destinado para el desarrollo de este proyecto va a ser un Ingeniero Industrial al que se le atribuye un sueldo mensual de 3100€ al mes. Así pues, el coste por hora de este trabajador será:

$$3100 \frac{\text{€}}{\text{mes}} \times \frac{1 \text{ mes}}{4 \text{ semanas}} \times \frac{1 \text{ semana}}{5 \text{ días laborables}} \times \frac{1 \text{ día laboral}}{8 \text{ horas}} = 19'38 \frac{\text{€}}{\text{h}}$$

El contenido del TFG tiene que ser de 300 horas trabajo, por tanto, se asignan 100 horas de trabajo al estudio con el LEGO Mindstorm NXT y 200 horas a los trabajos realizados con el brazaletе ya que supusieron mayor complejidad.

Rendimientos y mediciones

Finalmente, solo queda asignar el número de horas o de unidades de material que se requieren para llevar a cabo una unidad de obra y el número de unidades de obra necesarias.

Material	1.1	1.2	1.3	2.1	2.2	2.3	2.4	2.5	2.6
Rendimiento	100 h/ud	100 h/ud	2 uds/ud	1 ud/ud	1 ud/ud	1 ud/ud	200 h/ud	1 ud/ud	1 ud/ud

Unidad de obra	1	2
Medición	1	1

Cuadro de precios descompuestos

Debido a que se ha dividido el presupuesto en dos unidades de obra, se desarrollan dos cuadros de precios descompuestos, uno por unidad de obra. Cabe notar que no se establece ningún porcentaje en concepto de costes directos complementarios para ninguna de las dos unidades de obras.

Unidad de Obra 1:

Unidades	Descripción	Rendimiento	Precio	Importe
h	LEGO Mindstorms NXT 8547-NXT 2.0 V24	100	0,02	2 €
h	RobotC-Licencia de centro v2.0	100	0,01	1 €
ud	Sensor de aceleración Hitechnic para LEGO Mindstorms	2	55	110 €
h	Ingeniero Industrial	100	19,38	1938 €
			Total	2051 €

Unidad de Obra 2:

Unidades	Descripción	Rendimiento	Precio	Importe
ud	Bobina de impresión para el brazaletes	1	30	30 €
ud	BNO055 Intelligent 9-axis absolute orientation sensor (IMU)	1	30	30 €
ud	S110 nRF51 Bluetooth® low energy Peripheral SoftDeviceMATLAB	1	75	75 €
h	MATLAB	200	0,05	10 €
ud	Kit de cables	1	3,36	3,36 €
ud	Cinta de goma elástica blanca	1	1,5	1,5 €
h	Ingeniero Industrial	200	19,38	3876 €
			Total	4025,86 €

Presupuesto de Ejecución Material (PEM)

Unidad	Descripción de la unidad de obra	Medición	Precio	Importe
ud	Estudio del filtro de Kalman con la plataforma de LEGO Mindstorms NXT.	1	2051	2051 €
ud	Estudio del sistema de medida de la posición de un segmento corporal mediante tecnología wearable.	1	4025,86	4025,86 €
			PEM	6076,86 €

Presupuesto base de licitación

Debido a que no se considera, entre otros gastos, la energía eléctrica consumida por los distintos dispositivos electrónicos empleados o el consumo de agua en el ai2, se establece un porcentaje del 13% en concepto de Gastos Generales que cubran estos consumos.

PEM	6076,86 €
Gastos Generales (13%)	789,99 €
Presupuesto	6866,85 €

ANEJOS

Anejo I

x_k	Vector de estado del proceso
w_k	Vector de ruido del proceso
y_k	Vector de salida del proceso
z_k	Vector de medida del sensor
n_k	Vector de ruido del sensor
e_k	Vector de error del proceso
r_k	Vector referencia
u_k	Vector de entrada al proceso
\hat{x}_k	Vector de estimación del estado del proceso
A_k	Matriz de estado del proceso
B_k	Matriz de entrada al proceso
H_k	Matriz de medición
K_k	Ganancia del estimador
C	Matriz de salida

Tabla. Anejo 4

Anejo II

Ensayos 1 y 2

- (1) Proyección del estado hacia delante.

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1}$$

$$\hat{x}_k^- = \begin{bmatrix} \theta_k^- \\ \omega_k^- \end{bmatrix} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\theta}_{k-1} \\ \hat{\omega}_{k-1} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} u_{k-1} = \begin{bmatrix} \hat{\theta}_{k-1} + T_s \hat{\omega}_{k-1} \\ \hat{\omega}_{k-1} \end{bmatrix}$$

Línea 1: $\theta_k^- = \hat{\theta}_{k-1} + T_s \hat{\omega}_{k-1}$

Línea 2: $\omega_k^- = \hat{\omega}_{k-1}$

- (2) Proyección del error de la covarianza hacia delante.

$$P_k^- = AP_{k-1}A^T + Q$$

$$AP_{k-1}A^T = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ T_s & 1 \end{bmatrix} = \begin{bmatrix} P_{11} + T_s P_{21} & P_{12} + T_s P_{22} \\ P_{21} & P_{22} \end{bmatrix}$$

Línea 3: $a = P_{11} + T_s P_{21}$

Línea 4: $b = P_{12} + T_s P_{22}$

Línea 5: $c = P_{21}$

Línea 6: $d = P_{22}$

$$P_k^- = \begin{bmatrix} P_{11}^- & P_{12}^- \\ P_{21}^- & P_{22}^- \end{bmatrix} = AP_{k-1}A^T + Q = \begin{bmatrix} P_{11} + T_s P_{21} & P_{12} + T_s P_{22} \\ P_{21} & P_{22} \end{bmatrix} + \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} =$$

$$= \begin{bmatrix} P_{11} + T_s P_{21} + Q_{11} & P_{12} + T_s P_{22} + Q_{12} \\ P_{21} + Q_{21} & P_{22} + Q_{22} \end{bmatrix} = \begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix}$$

Línea 7: $P_{11}^- = a + Q_{11}$

Línea 8: $P_{12}^- = b + Q_{12}$

Línea 9: $P_{21}^- = c + Q_{21}$

Línea 10: $P_{22}^- = d + Q_{22}$

- (3) Cálculo de la ganancia de Kalman.

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1}$$

$$\begin{bmatrix} K_{11} \\ K_{21} \end{bmatrix} = \frac{\begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix}^T}{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix}^T + R} = \frac{\begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}}{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + R} =$$

$$= \frac{\begin{bmatrix} a + Q_{11} \\ c + Q_{21} \end{bmatrix}}{\begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + R} = \frac{\begin{bmatrix} a + Q_{11} \\ c + Q_{21} \end{bmatrix}}{a + Q_{11} + R} = \frac{\begin{bmatrix} P_{11}^- \\ P_{21}^- \end{bmatrix}}{P_{11}^- + R}$$

Línea 11: $K_{11} = \frac{P_{11}^-}{P_{11}^- + R}$

Línea 12: $K_{21} = \frac{P_{21}^-}{P_{11}^- + R}$

- (4) Actualización de la estimación con la medida z_k .

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$$

$$\begin{aligned}\hat{x}_k &= \begin{bmatrix} \hat{\theta}_k \\ \hat{\omega}_k \end{bmatrix} = \begin{bmatrix} \theta_k^- \\ \omega_k^- \end{bmatrix} + \begin{bmatrix} \frac{P_{11}^-}{P_{11}^- + R} \\ \frac{P_{21}^-}{P_{11}^- + R} \end{bmatrix} \left(z_k - \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \theta_k^- \\ \omega_k^- \end{bmatrix} \right) = \begin{bmatrix} \theta_k^- \\ \omega_k^- \end{bmatrix} + \begin{bmatrix} \frac{P_{11}^-}{P_{11}^- + R} \\ \frac{P_{21}^-}{P_{11}^- + R} \end{bmatrix} (z_k - \theta_k^-) \\ &= \begin{bmatrix} \theta_k^- + \frac{P_{11}^-}{P_{11}^- + R} (z_k - \theta_k^-) \\ \omega_k^- + \frac{P_{21}^-}{P_{11}^- + R} (z_k - \theta_k^-) \end{bmatrix} = \begin{bmatrix} \theta_k^- + K_{11}(z_k - \theta_k^-) \\ \omega_k^- + K_{21}(z_k - \theta_k^-) \end{bmatrix}\end{aligned}$$

Línea 13: $\hat{\theta}_k = \theta_k^- + K_{11}(z_k - \theta_k^-)$

Línea 14: $\hat{\omega}_k = \omega_k^- + K_{21}(z_k - \theta_k^-)$

(5) Actualización del error de covarianza.

$$P_k = (I - K_k H) P_k^- = P_k^- - K_k H P_k^-$$

$$\begin{aligned}K_k H P_k^- &= \begin{bmatrix} K_{11} \\ K_{21} \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} = \begin{bmatrix} K_{11} & 0 \\ K_{21} & 0 \end{bmatrix} \begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} \\ &= \begin{bmatrix} K_{11}(a + Q_{11}) & K_{11}(b + Q_{12}) \\ K_{21}(a + Q_{11}) & K_{21}(b + Q_{12}) \end{bmatrix}\end{aligned}$$

Línea 15: $a' = K_{11}(a + Q_{11})$

Línea 16: $b' = K_{11}(b + Q_{12})$

Línea 17: $c' = K_{21}(a + Q_{11})$

Línea 18: $d' = K_{21}(b + Q_{12})$

$$P_k = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} = \begin{bmatrix} P_{11}^- & P_{12}^- \\ P_{21}^- & P_{22}^- \end{bmatrix} - \begin{bmatrix} K_{11}(a + Q_{11}) & K_{11}(b + Q_{12}) \\ K_{21}(a + Q_{11}) & K_{21}(b + Q_{12}) \end{bmatrix} = \begin{bmatrix} P_{11}^- & P_{12}^- \\ P_{21}^- & P_{22}^- \end{bmatrix} - \begin{bmatrix} a' & b' \\ c' & d' \end{bmatrix}$$

Línea 19: $P_{11} = P_{11}^- - a'$

Línea 20: $P_{12} = P_{12}^- - b'$

Línea 21: $P_{21} = P_{21}^- - c'$

Línea 22: $P_{22} = P_{22}^- - d'$

Ensayo 3

(1) Proyección del estado hacia delante.

$$\hat{x}_k^- = A \hat{x}_{k-1} + B u_{k-1}$$

$$\hat{x}_k^- = \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_{k-1} \\ \omega_{k-1} \end{bmatrix} + \begin{bmatrix} T_s & 0 \\ 0 & T_s \end{bmatrix} \begin{bmatrix} a_{k-1} \\ \alpha_{k-1} \end{bmatrix} = \begin{bmatrix} v_{k-1} + T_s a_{k-1} \\ \omega_{k-1} + T_s \alpha_{k-1} \end{bmatrix}$$

Línea 1: $\theta_k^- = v_{k-1} + T_s a_{k-1}$

Línea 2: $\omega_k^- = \omega_{k-1} + T_s \alpha_{k-1}$

(2) Proyección del error de la covarianza hacia delante.

$$P_k^- = A P_{k-1} A^T + Q$$

$$A P_{k-1} A^T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_{11}^- & P_{12}^- \\ P_{21}^- & P_{22}^- \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} P_{11}^- & P_{12}^- \\ P_{21}^- & P_{22}^- \end{bmatrix}$$

Línea 3: $a = P_{11}$

Línea 4: $b = P_{12}$

Línea 5: $c = P_{21}$

Línea 6: $d = P_{22}$

$$\begin{aligned}P_k^- &= \begin{bmatrix} P_{11}^- & P_{12}^- \\ P_{21}^- & P_{22}^- \end{bmatrix} = A P_{k-1} A^T + Q = \begin{bmatrix} P_{11}^- & P_{12}^- \\ P_{21}^- & P_{22}^- \end{bmatrix} + \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} = \begin{bmatrix} P_{11}^- + Q_{11} & P_{12}^- + Q_{12} \\ P_{21}^- + Q_{21} & P_{22}^- + Q_{22} \end{bmatrix} = \\ &= \begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix}\end{aligned}$$

Línea 7: $P_{11}^- = a + Q_{11}$

Línea 8: $P_{12}^- = b + Q_{12}$

Línea 9: $P_{21}^- = c + Q_{21}$

Línea 10: $P_{22}^- = d + Q_{22}$

(3) Cálculo de la ganancia de Kalman.

Para el desarrollo del cálculo de la ganancia de Kalman se ha de tener en cuenta una serie de hipótesis:

En primer lugar se tiene en cuenta la hipótesis de que las matrices P_k^- y R tienen que ser diagonales y que la matriz H (o también denominada C) es la identidad, por lo que

$P_k^- = \begin{bmatrix} P_{11}^- + Q_{11} & 0 \\ 0 & P_{22}^- + Q_{22} \end{bmatrix}$ y $R = \begin{bmatrix} R_{11} & 0 \\ 0 & R_{22} \end{bmatrix}$ además de que la ecuación de la

ganancia de Kalman quedará de la forma $K_k = P_k^- (P_k^- + R)^{-1}$. A continuación se hace

un pequeño cambio de variables en la matriz P_k^- , $P_k^- = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix}$. Así pues:

$$\begin{aligned} K_k &= \frac{P_k^-}{P_k^- + R} = \frac{\begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix}}{\begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} + \begin{bmatrix} R_{11} & 0 \\ 0 & R_{22} \end{bmatrix}} = \frac{\begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix}}{\begin{bmatrix} A_{11} + R_{11} & 0 \\ 0 & A_{22} + R_{22} \end{bmatrix}} = \\ &= \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} A_{11} + R_{11} & 0 \\ 0 & A_{22} + R_{22} \end{bmatrix}^{-1} \end{aligned}$$

Ahora se procede a hacer el cálculo de la matriz inversa sabiendo que:

$$A^{-1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{\det(A)} \left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}^* \right)^T = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Por lo que:

$$\begin{aligned} K_k &= \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \frac{1}{(A_{11} + R_{11})(A_{22} + R_{22}) - 0} \begin{bmatrix} A_{22} + R_{22} & 0 \\ 0 & A_{11} + R_{11} \end{bmatrix} = \\ &= \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} \frac{A_{22} + R_{22}}{(A_{11} + R_{11})(A_{22} + R_{22})} & 0 \\ 0 & \frac{A_{11} + R_{11}}{(A_{11} + R_{11})(A_{22} + R_{22})} \end{bmatrix} = \\ &= \begin{bmatrix} A_{11} \frac{A_{22} + R_{22}}{(A_{11} + R_{11})(A_{22} + R_{22})} + 0 & 0 \\ 0 & A_{22} \frac{A_{11} + R_{11}}{(A_{11} + R_{11})(A_{22} + R_{22})} + 0 \end{bmatrix} = \\ &= \begin{bmatrix} \frac{A_{11}A_{22} + A_{11}R_{22}}{A_{11}A_{22} + A_{11}R_{22} + R_{11}A_{22} + R_{11}R_{22}} & 0 \\ 0 & \frac{A_{22}A_{11} + A_{22}R_{11}}{A_{11}A_{22} + A_{11}R_{22} + R_{11}A_{22} + R_{11}R_{22}} \end{bmatrix} \end{aligned}$$

Deshaciendo el cambio de variable K_k es igual a:

$$\begin{bmatrix} \frac{(P_{11} + Q_{11})(P_{22} + Q_{22}) + (P_{11} + Q_{11})R_{22}}{(P_{11} + Q_{11})(P_{22} + Q_{22}) + (P_{11} + Q_{11})R_{22} + R_{11}(P_{22} + Q_{22}) + R_{11}R_{22}} & 0 \\ 0 & \frac{(P_{22} + Q_{22})(P_{11} + Q_{11}) + (P_{22} + Q_{22})R_{11}}{(P_{11} + Q_{11})(P_{22} + Q_{22}) + (P_{11} + Q_{11})R_{22} + R_{11}(P_{22} + Q_{22}) + R_{11}R_{22}} \end{bmatrix}$$

Siendo K_{11} la primera entrada de la diagonal (línea 11) y K_{22} (línea 12) la segunda entrada de la diagonal mientras que K_{12} Y K_{21} son nulas.

(4) Actualización de la estimación con la medida z_k .

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$$

$$\begin{aligned}\hat{x}_k = \begin{bmatrix} \hat{v}_k \\ \hat{\omega}_k \end{bmatrix} &= \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} + K_k \left(z_k - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} \right) = \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} + K_k \left(z_k - \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} \right) = \begin{bmatrix} v_k^- + K_k \left(z_k - \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} \right) \\ \omega_k^- + K_k \left(z_k - \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} \right) \end{bmatrix} \\ &= \begin{bmatrix} v_k^- + K_{11} \left(z_k - \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} \right) \\ \omega_k^- + K_{22} \left(z_k - \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} \right) \end{bmatrix}\end{aligned}$$

$$\text{Línea 13: } \hat{v}_k = v_k^- + K_{11} \left(z_k - \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} \right)$$

$$\text{Línea 14: } \hat{\omega}_k = \omega_k^- + K_{22} \left(z_k - \begin{bmatrix} v_k^- \\ \omega_k^- \end{bmatrix} \right)$$

(5) Actualización del error de covarianza.

$$P_k = (I - K_k H) P_k^- = P_k^- - K_k H P_k^-$$

$$\begin{aligned}K_k H P_k^- &= \begin{bmatrix} K_{11} \\ K_{21} \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} = \begin{bmatrix} K_{11} & 0 \\ 0 & K_{22} \end{bmatrix} \begin{bmatrix} a + Q_{11} & b + Q_{12} \\ c + Q_{21} & d + Q_{22} \end{bmatrix} \\ &= \begin{bmatrix} K_{11}(a + Q_{11}) & K_{11}(b + Q_{12}) \\ K_{22}(c + Q_{21}) & K_{22}(d + Q_{22}) \end{bmatrix}\end{aligned}$$

$$\text{Línea 15: } a' = K_{11}(a + Q_{11})$$

$$\text{Línea 16: } b' = K_{11}(b + Q_{12})$$

$$\text{Línea 17: } c' = K_{22}(c + Q_{21})$$

$$\text{Línea 18: } d' = K_{22}(d + Q_{22})$$

$$P_k = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} - \begin{bmatrix} K_{11}(a + Q_{11}) & K_{11}(b + Q_{12}) \\ K_{22}(c + Q_{21}) & K_{22}(d + Q_{22}) \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} - \begin{bmatrix} a' & b' \\ c' & d' \end{bmatrix}$$

$$\text{Línea 19: } P_{11} = P_{11} - a'$$

$$\text{Línea 20: } P_{12} = P_{12} - b'$$

$$\text{Línea 21: } P_{21} = P_{21} - c'$$

$$\text{Línea 22: } P_{22} = P_{22} - d'$$