



**UNIVERSIDAD DE CONCEPCIÓN
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA**



IMPLEMENTACIÓN DE UN FILTRO DE KALMAN EXTENDIDO EN TIEMPO REAL

POR

José Matías Hernández Ramírez

Memoria de Título presentada a la Facultad de Ingeniería de la Universidad de Concepción para optar al título profesional de Ingeniero(a) Civil Electrónico(a)

Profesor Guía
Juan Pablo Segovia Vera

Diciembre 2022
Concepción (Chile)

©2022 José Matías Hernández Ramírez

©2022 José Matías Hernández Ramírez

Ninguna parte de esta tesis puede reproducirse o transmitirse bajo ninguna forma o por ningún medio o procedimiento, sin permiso por escrito del autor.

A Dios, que me ha dado la oportunidad de llegar a este punto de mi vida.

Agradecimientos

Quiero agradecer primeramente a Dios, que me ha dado la gran oportunidad de poder estudiar esta carrera y colocar en mi camino a grandes compañeros, amigos y profesores que aportaron a mi desarrollo profesional. A todos mis familiares por apoyarme cuando más lo necesite y sentía que todo se volvía cuesta arriba, y, sobre todo, por creer en mí, incluso las veces cuando dudaba de mi capacidad para salir adelante.

A mis amigos John, Gustavo, Cristian y Cecil, por ayudar a desenvolverme en un ambiente tan nuevo en ese entonces, por las horas de estudio y simplemente por estar ahí para hablar de lo que nos interesara en el momento. Fueron una parte importante para mí al entrar a la universidad, les agradezco por resolver todas mis dudas y por esas tardes en donde jugábamos y charlábamos.

Al profesor Juan Pablo Segovia, por impartir todo el conocimiento que estuvo a su disposición, por sus consejos y apoyarme en el desarrollo de mi memoria. Gracias por ser exigente en sus ramos y laboratorios, por usted muchos de nosotros pudimos avanzar hasta donde estamos hoy.

A Christofer y al personal de Green, por darme la oportunidad de realizar mi práctica profesional en su empresa y enseñarme parte de su experiencia, por ayudarme a aplicar todo lo que me enseñaron en la universidad y explicarme cuando había cosas que no entendía.

A mi tío José Ángel, por ser siempre un gran apoyo para nuestra familia y acompañarnos en los momentos más importantes de nuestra vida. Aunque ya partiste y no puedo decirte lo mucho que significaste para mí, siempre te estaré agradecido por escucharme y enseñarme valores como persona. Muchas gracias por los consejos que me diste, desde que tengo memoria estuviste con nosotros, fuiste un segundo padre, muchas gracias por escucharme y estar conmigo cuando necesitaba hablar con alguien. Aunque ya no podemos vernos, espero llegue el día en que volvamos a reunirnos y poder darte las gracias por todo lo que significaste para mí.

A mis padres, no tengo palabras para agradecerles por todo lo que significaron en estos largos años de estudio. Muchas gracias por darme palabras de apoyo y “levantarme con pala” cuando me frustraba y todo se volvía oscuro. Gracias por hacerme ser quien soy, por enseñarme valores y hábitos que me sirvieron hasta el día de hoy. También a mis hermanos, por estar ahí cuando los necesité y darme palabras de apoyo.

A todas las otras personas que aportaron en mi desarrollo profesional y personal, por ayudarme a desenvolverme como persona y aportar su granito de arena para ayudarme a llegar hasta donde estoy actualmente, muchas gracias.

Sumario

En el presente documento se describe de forma detallada el proceso de diseño, simulación e implementación de un Filtro de Kalman Extendido para la planta piloto en los lazos de flujo, nivel y temperatura, ubicada en el laboratorio de control de procesos del DIE en la Universidad de Concepción.

Para realizar el presente proyecto se realizó una búsqueda bibliográfica sobre los temas a incluir en el diseño, donde se consideró el comportamiento de los lazos a distintos cambios de entrada para realizar un modelo que representara de forma precisa el comportamiento de los lazos a aproximar, además del diseño de un algoritmo basado en Mínimos Cuadrados Recursivos y el Filtro de Kalman Extendido para la estimación de parámetros y salidas del sistema.

Se estudiaron los comportamientos de los lazos de la planta a través de pruebas y posterior aproximación mediante un modelo discreto basado en espacios de estado, esto con la finalidad de ajustarse a la respuesta de la planta, en la cual los tres lazos mencionados tienen tiempos muertos diferentes. Con el modelo, se desarrolla un método recursivo para obtener los valores de los parámetros del modelo en cada instante de tiempo. Una vez hecho esto, se realiza la estimación de la salida con el Filtro de Kalman Extendido, analizando en una fase inicial las ecuaciones que componen al filtro y su impacto, para luego realizar una estimación simultánea de parámetros y estados con el filtro, esto tratando los parámetros del modelo como nuevos estados extendidos, logrando ajustarse al tiempo muerto del sistema y reduciendo el ruido añadido en la medición tomada por los sensores de la planta.

Para evaluar el Filtro de Kalman Extendido en tiempo real, se realizan además controladores difusos con la finalidad de controlar el flujo, nivel y temperatura de la planta a través de las señales estimadas por el filtro, las cuales tienen ruido reducido y son capaces de ajustarse al tiempo muerto del sistema. Las entradas consideradas fueron las válvulas etiquetadas como FV-01, FV-02, FV-03 y FV-04 según el lazo a estimar, y los sensores FT-01, LT-01 y TT-01 que toman las mediciones.

La implementación del algoritmo desarrollado se realiza en el PLC de la planta piloto, en formato de texto estructurado en Rslogix 5000 una vez que la simulación de Matlab es correcta. Se analizan el error antes y después de la aplicación del filtro, la correlación entre la señal real y la señal estimada, así como la evolución de los parámetros correspondientes al lazo estimado, para luego realizar cambios en el sistema con el controlador y reevaluar la evolución del algoritmo en tiempo real para flujo, nivel y temperatura.

Summary

This document describes the detailed form of design process, simulation and implementation of an Extended Kalman Filter for Pilot Plant for flow, level and temperature loops, located in Control Process Laboratory in DIE, University of Concepción.

To realize the present project, it had been made a bibliograph search about topics included in the design, where the behavior of the loops at different inputs was considered to make a model that accurately represent the behavior of the loops to be approximated, in addition to the design of an algorithm based in Recursive Least Squares and Extended Kalman Filter to Parameters and Outputs estimation.

The behavior of the plant loops was studied through test and subsequent approximation by a discrete model based in space states, this in order to adjust to the plant response, in which the three loops mentioned have different dead time. With the model, a recursive method is developed to obtain the values of model parameters at each instant of time. Once this is done, the output is estimated with the Extended Kalman Filter, analyzing in an initial phase the equations that make up the filter and its impact, then perform a simultaneous parameter and state estimation with the filter, this redefined the model parameters as new extended states, managing to adjust to the dead time of the system and reducing the added noise in the measurement taken by the plant sensors.

To evaluate the Extended Kalman Filter in real time, fuzzy controllers are realized in order to control the flow, level and temperature of the plant through the signal estimated by the filter, which have reduced noise and are capable of adjust to the dead time of the system. The inputs considered were the vales labeled FV-01, FV-02, FV-03 and FV-04 according to the loop to be estimated, and the sensors FT-01, LT-01 and TT-01 that take the measurements.

The implementation of the developed algorithm is programmed in the PLC of the pilot plant in structured text format in Rslogix 500 once the Matlab Simulation is correct. The error before and after the application of the filter, the correlation between the estimated signal and real signal, as well the evolution of the parameters corresponding to the estimated loop is analyzed to make changes in the system with the controller and reappraise the evolution of the algorithm in real time for flow, level and temperature.

Tabla de contenidos

Tabla de contenido

Agradecimientos.....	iv
Sumario	v
Summary	vi
Tabla de contenidos.....	vii
Lista de tablas.....	x
Lista de figuras	xi
Abreviaciones y símbolos	xiii
Capítulo 1. Introducción	1
1.1. Introducción general	1
1.2. Trabajos Previos.....	3
1.2.1. Teoría y Técnicas Actuales	3
1.2.2. Simulación y Experimentación	6
1.2.3. Teoría y Aplicación de Controladores	9
1.3. Discusión.....	11
1.4. Hipótesis de Trabajo	11
1.5. Objetivos	12
1.5.1. Objetivo General	12
1.5.2. Objetivo Específicos	12
1.6. Alcances y Limitaciones	12
1.7. Temario y Metodología.....	12
Capítulo 2. Modelación de la planta piloto	14
Capítulo 3. Selección estructural del filtro.....	16
3.1. Introducción al Filtro de Kalman	16
3.2. Modelo en espacios de estado	16

3.3. Filtro de Kalman Extendido	17
Capítulo 4. Identificación estructural	22
4.1. Identificación de parámetros del sistema	22
4.2. Identificación simultánea de parámetros y estados con el Filtro de Kalman Extendido	23
4.3. Aplicación del Filtro de Kalman al Sistema en Modo Manual	25
Capítulo 5. Controladores	32
5.1. Introducción a Lógica Difusa	32
5.2. Fuzzificación y Funciones de Inferencia.....	32
5.3. Reglas Difusas y Matriz de Inferencia	35
5.4. Defuzzificación	36
Capítulo 6. Resultados Experimentales.....	38
6.1. Generalidades.....	38
6.2. Lazo de Flujo.....	42
6.3. Lazo de Nivel	43
6.4. Lazo de Temperatura.....	45
Capítulo 7. Conclusiones.....	47
7.1. Sumario	47
7.2. Conclusiones	48
7.3. Trabajos futuros.....	50
Referencias	51
Anexo A. Obtención del Modelo Discreto.....	54
Anexo B. Filtro de Kalman Suavizado y Filtro Partícula de Kalman Extendido.....	58
B.1. Filtro de Kalman Suavizado	58
B.2. Filtro Partícula de Kalman Extendido	61
Anexo C. Diagramas de flujo.....	65

Anexo D. Códigos implementados.....	66
D.1. Código Filtro de Kalman Extendido	66
D.2. Código Controlador Flujo	73
D.3. Código Controlador Nivel	78
D.4. Código Controlador Temperatura	81

Lista de tablas

Tabla 2.1. Errores cuadráticos extraídos de [9].....	15
Tabla 5.3.1. Matriz de Inferencia.....	35
Tabla 5.4.1. Términos utilizados para Sintonización de los controladores.....	37

Lista de figuras

Figura 1.1.1. Diagrama en bloques del Filtro de Kalman, [14].....	1
Figura 1.1.2. Filtro de Kalman aplicado a un PLC Siemens, [13].....	2
Figura 3.3.1. Salida simulada con ruido v/s Salida aproximada mediante EKF.....	20
Figura 3.3.2. Error del Filtro de Kalman Extendido.....	20
Figura 4.1.1. Respuesta del sistema ante entradas escalón.....	22
Figura 4.3.1. Comparación entre Sensor de Flujo FT-01 y Salida de Flujo Estimada por EFK.....	25
Figura 4.3.2. Error de Estimación de Flujo por EKF.....	26
Figura 4.3.3. Error de Predicción de Flujo en Etapa de Identificación.....	26
Figura 4.3.4. Parámetros del Lazo de Flujo.....	26
Figura 4.3.5. Correlación cruzada entre FT-01 y Flujo estimado.....	27
Figura 4.3.6. Comparación entre Sensor de Nivel LT-01 y Salida de Nivel Estimada por EKF.....	27
Figura 4.3.7. Error de Estimación de Nivel por EKF.....	28
Figura 4.3.8. Error de Predicción de Nivel en Etapa de Identificación.....	28
Figura 4.3.9. Parámetros del Lazo de Nivel.....	28
Figura 4.3.10. Correlación cruzada entre LT-01 y Nivel estimado.....	29
Figura 4.3.11. Comparación entre Sensor de Temperatura TT-01 y Temperatura estimada por EKF.....	29
Figura 4.3.12. Error de estimación de Temperatura por EKF.....	30
Figura 4.3.13. Error de predicción de Temperatura en Etapa de identificación.....	30
Figura 4.3.14. Parámetros del Lazo de Temperatura.....	30
Figura 4.3.15. Correlación cruzada entre TT-01 y Temperatura estimada.....	31
Figura 5.2.1. Composición de un controlador difuso, [23].....	33
Figura 5.2.2. Funciones de membresía para error.....	34
Figura 5.2.3. Funciones de membresía para derivada del error.....	34
Figura 5.4.1. Funciones de membresía para cambio de la acción de control.....	36
Figura 6.1.1. Vista general de la planta piloto y P&ID.....	38
Figura 6.1.2. Bomba B-01 y Válvula FV-02.....	39
Figura 6.1.3. Calefones para el lazo de temperatura.....	39
Figura 6.1.4. PLC Allen Bradley 1756 de la planta piloto.....	40

Figura 6.1.5. Diagrama en bloques del conjunto control, planta y algoritmo.....	41
Figura 6.2.1. Control de Flujo de descarga con controlador y Filtro de Kalman Extendido..	42
Figura 6.2.2. Error de predicción durante el control de flujo.....	43
Figura 6.2.3. Error de estimación durante el control de flujo.....	43
Figura 6.3.1. Control del Nivel del Estanque con controlador y Filtro de Kalman Extendido.....	44
Figura 6.3.2. Error de predicción durante el control de nivel.....	44
Figura 6.3.3. Error de estimación durante el control de nivel.....	45
Figura 6.4.1. Control de temperatura con controlador y Filtro de Kalman Extendido.....	45
Figura 6.4.2. Error de predicción durante el control de temperatura.....	46
Figura 6.4.3. Error de estimación durante el control de temperatura.....	46
Figura C.1. Diagrama de Flujo del algoritmo.....	65

Abreviaciones y símbolos

Abreviaciones

L.A.	: lazo abierto.
L.I.T.	: lineal invariante en el tiempo.
F. de T.	: función de transferencia.
E.S.	: entrada/salida.
S.S.	: estado estacionario.
SISO	: sistema de una entrada y una salida (single input single output).
MISO	: sistema de múltiples entradas y una salida (multiple input single output)
T.L.	: Transformada de Laplace.
K.F.	: Filtro de Kalman
E.K.F.	: Filtro de Kalman Extendido (Extended Kalman Filter).
U.K.F.	: Filtro de Kalman Suavizado (Unscented Kalman Filter)
E.K.P.F.	: Filtro Partícula de Kalman Extendido (Extended Kalman Particle Filter)
R.L.S.	: Mínimos Cuadrados Recursivos con Factor de Olvido (Recursive Least Squares).
SSE	: Error Cuadrático
FT-01	: Sensor de Flujo
LT-01	: Sensor de Nivel
TT-01	: Sensor de Temperatura
FV-01	: Válvula de carga de agua
FV-02	: Válvula de descarga de agua
FV-03	: Válvula de agua caliente
FV-04	: Válvula de agua fría

Símbolos

$h(s)$: Función de transferencia en tiempo continuo.
$H(z)$: Función de transferencia en tiempo discreto
θ	: Retardo
τ	: Constante de tiempo
k_m	: Ganancia de la F. de T.
K_x	: Parámetro “x” del sistema
x_k	: Estados del modelo en espacios de estado
y_k	: Salida del modelo en espacios de estado
u_k	: Entrada del modelo en espacios de estado
f	: Función de proceso
h	: Función de medida
w_k	: Ruido blanco de proceso
v_k	: Ruido blanco de medida
A	: Matriz jacobiana de función de proceso
H	: Matriz jacobiana de función de medida
\hat{x}_k	: Estado a posteriori
\hat{x}_k^-	: Estado a priori
Q_k	: Matriz de covarianza del ruido de proceso

R_k	: Matriz de covarianza del ruido de medida
P_k^-	: Matriz de covarianza del error a priori
P_k	: Matriz de covarianza del error a posteriori
K_k	: Matriz de Ganancia de corrección de Kalman
φ_k	: Matriz de regresores
θ_k	: Vector de parámetros del sistema
Pi_k	: Matriz de covarianza de identificación de parámetros
Ki_k	: Matriz de ganancia de corrección de identificación de parámetros
nu	: Total de parámetros a identificar
z_k	: Estado extendido de modelo en espacios de estado
a_k	: Parámetros desconocidos en función de proceso
b_k	: Parámetros desconocidos en función de medida
Es	: Error normalizado del controlador
Eds	: Derivada del error normalizada del controlador
du_k	: Cambio en la señal del actuador
Se	: Factor de escalamiento del error normalizado
Sde	: Factor de escalamiento de la derivada del error normalizada
Sdu	: Factor de escalamiento del cambio en la señal del actuador
T_o	: Tiempo de muestreo

Capítulo 1. Introducción

1.1. Introducción general

Las mediciones siempre han jugado un papel fundamental en la identificación y control de procesos, permitiéndonos acceder a las variables que en la mayoría de los casos no son intuitivas. A simple vista, para quién esté operando el proceso, desde voltaje presente en convertidores hasta cantidad de material fluyendo por una cañería, son de vital importancia para lograr un correcto funcionamiento del sistema.

Además, durante la obtención de las medidas requeridas, a veces nos podemos encontrar con desafíos tales como parámetros variables en el tiempo, retardos del sistema que poseen cierto grado de variación y otros tipos de no linealidades inherentes al proceso. A esto se suma que, al tomar los datos en tiempo real, éstos tienen ruido añadido por los sensores encargados de realizar las mediciones, lo que hace aún más difícil las tareas como la identificación del sistema y control de las secuencias que debiera realizar la planta durante sus rutinas diarias.

En 1960, R. Kalman propuso un método basado en modelos de espacios de estado para predecir la salida del sistema y realizar un filtro de ruido para la medición tomada, actualmente llamado Filtro de Kalman, [3], el cual a lo largo de los años ha sido objetivo de muchos estudios y aplicaciones, ya sea para estimación de señales lineales como no lineales. Con el paso del tiempo, se ha mejorado el Filtro de Kalman original, con el fin de ampliar el rango de aplicaciones y lograr mejores resultados durante la estimación del sistema, ya sea incluyendo los parámetros del sistema a las ecuaciones del filtro [4], como también disminuir el error al usar una base distinta a las matrices jacobianas, [5, 7, 18]. En el siguiente diagrama se observa el proceso iterativo del filtro, [14]:

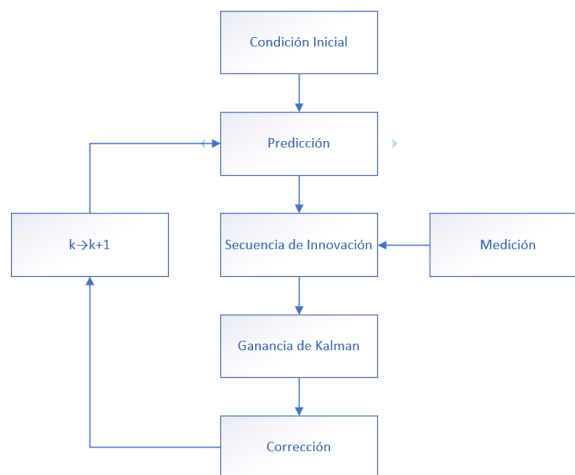


Figura 1.1.1. Diagrama en bloques del Filtro de Kalman, [14]

Entre las aplicaciones que ha tenido este método en la industria, se encuentran localización/rastreo de objetivos, eliminación y mejora de señales, desconvolución, redes de sensores inalámbricos y estimación distribuida, [17]. En estas aplicaciones mencionadas, se relaciona la necesidad de que la información encontrada en la señal detectada no se vea afectada por el ruido, el que puede añadirse durante la medición o el proceso, a fin de obtener resultados de confianza.

Con el objetivo de ser implementado en tiempo real, los principales desafíos que se enfrentan al implementar un Filtro de Kalman Extendido son: (I) Minimización del efecto computacional en el redondeo del error en la estabilidad del ruido al computar las matrices de covarianza, (II) Reducción de la carga computacional, (III) Minimizar el tiempo de ejecución. Estos problemas han intentado abordarse ampliamente en la literatura desde dos puntos de vista, las cuales son modificar el algoritmo a ser implementado y usar un sistema digital más eficiente para ejecutar el estimador, lo cual da como resultado un procesador o hardware dedicado para la implementación, [17].

Una aplicación bastante similar a la realizada en este proyecto se encuentra en [13], donde se aplicó un Filtro de Kalman en un PLC de la marca Siemens S7-1200 en un proceso de medición de temperatura realizado por un sensor tipo Pt100, donde los resultados se observan a continuación.



Figura 1.1.2. Filtro de Kalman aplicado a un PLC Siemens, [13]. Autores: Wang, Yao y Qiu

Donde la curva en color negro corresponde a los datos recolectados por el sensor físico, mientras que la señal roja corresponde a la estimación realizada por el Filtro de Kalman, observándose una disminución en el ruido que posee la señal y mejoramiento de la misma, lo que

resulta en una opción a considerar a la hora de necesitarse una estimación de una señal proveniente de un proceso con mucho ruido, lo que es de vital importancia para los controladores, [14].

En el presente proyecto se realizará una implementación de un Filtro de Kalman Extendido para realizar estimación del estado y parámetros del sistema según los modelos que representan a cada uno de los lazos de la planta en la que se realizará la implementación, con el fin de disminuir el ruido presente en las mediciones y lograr una comprensión del proceso, a fin de realizar una acción de control libre de ruido proveniente de las mediciones.

1.2. Trabajos Previos

1.2.1. Teoría y Técnicas Actuales

♣ G. Welch and G. Bishop “An Introduction to the Kalman Filter,” University of North Carolina at Chapel Hill, TR-95-041, Jul 24. 2006, [1].

Este documento realiza una introducción al set de ecuaciones matemáticas una versión del Filtro de Kalman para sistemas lineales y otra versión llamada Filtro de Kalman Extendido para sistemas no lineales, haciendo especial énfasis en la recursividad de las ecuaciones de forma generalizada, dejando claro el proceso a seguir para programar el algoritmo en el software que sea de interés del usuario.

♣ D. Morrel “Extended Kalman Filter Lecture Notes,” *EEE 581-Spring*, 1997, [2].

Este documento plantea la extensión del filtro de Kalman para sistemas no lineales, detallando las acciones respectivas con el modelo a estimar a partir de una linealización en el punto estimado del estado y aclarando las diferencias con la aplicación del filtro a sistemas lineales. Además, se hace una introducción al filtro de Kalman Continuo-Discreto, es decir, aplicación de la teoría hacia un proceso continuo con una medición discreta.

♣ R. Kalman “A New Approach to Linear Filtering and Prediction Problems,” in *ASME Journal of Basic Engineering*, 82, pp 35-45, Marzo 1, 1960, [3].

Este documento fue el que R. Kalman presentó en 1960 y fue el inicio de todas las aplicaciones actuales del Filtro de Kalman. Se plantea el problema que en ese entonces existía al momento de analizar el problema de Wiener para señales conocidas en presencia de ruido y predicción de éstas, presentando un nuevo método derivado de la matriz de covarianza del error estimado, limpiando las señales del ruido presente al mismo tiempo que se asegura que, para un sistema lineal, el Filtro de Kalman es la mejor estimación posible, generalizando la solución y planteando las ecuaciones que dieron origen al Filtro.

♣ W. Liu “Appendix A. System Identification: State and Parameter Estimation Techniques,” *Introduction to Hybrid Vehicle System Modeling and Control, First Edition*. John Wiley & Sons. 2013, [4].

El apéndice A de este libro realiza una introducción a técnicas para la identificación de sistemas tanto para el estado como para los parámetros, desde las conocidas técnicas de mínimos cuadrados para una estimación de los parámetros como para el estado, presentando el filtro de Kalman. Se hace hincapié en una modificación al filtro de Kalman para realizar ambas tareas de identificación de parámetros y estados de forma conjunta al realizar una extensión del estado actual conocido, es decir, tratar los parámetros desconocidos o de interés en filtrar como nuevos estados y aplicar la teoría conocida al nuevo set de estados.

♣ P. Madhukar and L. Prasad “State Estimation using Extended Kalman Filter and Unscented Kalman Filter,” in International Conference on Emerging Trends in Communications, Control and Computing (ICONC3), Mody University of Science and Technology, Lakshmanagarh, Febrero 21-22, 2020, [5].

Este documento hace una revisión de las técnicas actuales conocidas acerca del filtro de Kalman, presentando el Filtro de Kalman para sistemas lineales y el Filtro de Kalman en su versión extendida para sistemas no lineales. Sin embargo, se aborda la problemática que el Filtro de Kalman Extendido tiene con términos de naturaleza numérica elevada, es decir, las aproximaciones a través de jacobianos y hessianas, presentando una alternativa al Filtro de Kalman Extendido que aborda el mismo problema con una solución diferente, llamada Filtro de Kalman Suavizado, a través de puntos sigma para capturar el promedio y covarianza del estado estimado.

♣ B. Tamani “Obtención de Modelos de Procesos Mediante Métodos de Identificación Recursiva”. Artículo de Revista. Facultad de Ingeniería Electrónica y Eléctrica, Universidad Nacional Mayor de San Marcos, Lima, Perú. No. 19, agosto 18, 2007, [6].

Este artículo hace un análisis detallado hacia la identificación recursiva de un sistema, derivando las ecuaciones que componen la técnica conocida como mínimos cuadrados recursivos con factor de olvido, esto con el fin de obtener una estimación de los parámetros que componen el sistema e identificar posibles variaciones temporales que puedan existir en el proceso y realizar un seguimiento. Es importante recalcar que los parámetros a obtener dependen de la estructura con la que se defina el proceso, por lo que previo a realizar la iteración de esta técnica, es importante asegurarse de que la estructura a usar representa en buena manera al proceso de interés.

♣ E. Wan and R. van der Merwe “The Unscented Kalman Filter for Nonlinear Estimation,” in *Adaptative Systems for Signal Processing, Communications, and Control Symposium*, Lake Louise, AB, Canada. Octubre 2020, pp 153-158, [7].

Este documento plantea los actuales usos que tiene el Filtro de Kalman Extendido, el cual es el más usado actualmente en estimaciones de sistemas no lineales, al mismo tiempo que plantea las debilidades de la técnica, al realizar una estimación de primer orden al sistema, lo cual puede inducir a errores en estados posteriores tanto en la media como en la covarianza. Es por esto que se hace énfasis en las ventajas que el Filtro de Kalman Suavizado trae al realizar una aproximación del sistema usando un set mínimo de muestras cuidadosamente elegidas, llamados puntos sigma a través de la transformada suavizada, en ingles *unscented transformation*.

♣ G. Terejanu “Extended Kalman Filter Tutorial,” Department of Computer Science and Engineering, University of Buffalo, Buffalo, NY, Estados Unidos, 2009, [8].

Este documento explica los orígenes estadísticos de las ecuaciones que componen el Filtro de Kalman Extendido. Se realiza una revisión a las fórmulas que componen las secuencias de predicción y corrección para comprender su origen, considerando los supuestos realizados a lo largo de los desarrollos para el ruido de proceso y ruido de medida como ruido gaussiano blanco de media cero. Se presentan las ecuaciones iterativas para el proceso de estimación de las variables en espacios de estado, para finalmente realizar un análisis de la estabilidad del método.

♣ F. Sánchez “Un Nuevo Método de Identificación de Procesos Continuos No Oscilatorios de Alto Orden,” Tesis de Master, Instituto Tecnológico y de Estudios Superiores de Monterrey, Monterrey, México, 2001, [9].

Este documento hace una introducción a un método para la identificación de sistemas de forma gráfica, de forma de analizar si un sistema a partir de datos experimentales puede ser modelado como un sistema de primer o segundo orden. Se hace énfasis tanto en la ganancia y constantes de tiempo del proceso como también del retardo, haciendo especial hincapié en este último en las formas de aproximación que existen, planteando una forma distinta de aproximar el retardo desde el punto de vista del error cuadrático, realizando una comparación con la aproximación de Padé.

♣ H. Wang “Improved Extended Kalman Particle Filter Based on Markov chain Monte Carlo for Nonlinear State Estimation,” in International Conference on Uncertainty Reasoning and Knowledge Engineering, Jalarta, Indonesia. Agosto 2012, pp 281-285, [10].

Este documento plantea los desafíos existentes en la precisión en el seguimiento y degradación de partículas en el filtro de partícula tradicional. Para solucionar esto, se plantea un algoritmo ligeramente diferente usando el filtro de Kalman extendido y el filtro de partícula extendido, siendo optimizado por el método de Monte Carlo basado en cadenas de Markov para el muestreo, dando origen Filtro de Partícula de Kalman Extendido, resolviendo la degradación de las partículas y mejorando la precisión en el seguimiento de las partículas durante la estimación.

1.2.2. Simulación y Experimentación

♣ T. Phuong, M. Belov, N. Lanh and N. Thanh “State Estimation of Nonlinear Electromechanical System using Extended Kalman Filter,” 3rd International Conference on Control in Technical Systems, San Petersburgo, Rusia, octubre 2019, [11].

Este documento hace una aplicación del filtro de Kalman para estimar las coordenadas mecánicas no medibles del vector de estados en el controlador de un servomotor para el control de un radiotelescopio. Se hace especial énfasis en las capacidades recursivas del Filtro de Kalman Extendido y su aplicación al sistema no lineal planteado en una fase inicial, así como las consideraciones a tener a la hora de estimar un modelo de esta forma.

♣ J. Seung, A. Atiya, A. Parlos and K. Chong “Parameter Estimation for Coupled Tank using Estimate Filtering”, in International Journal of Control and Automation. Vol.6 No.5, 2013, pp 91 – 102, [12].

Este documento presenta un uso del Filtro de Kalman aplicado a un sistema de tanques acoplados, realizando una estimación del estado al mismo tiempo que se realiza una estimación de parámetros que son de interés para determinar en el tipo de tanques usado. El documento realiza una breve introducción teórica a los métodos del Filtro de Kalman Extendido y Filtro de Kalman Suavizado, detallando las ecuaciones y metodología de ambos métodos, así como las ventajas que cada uno tiene. Luego se realiza una fase de experimentación donde se aplica la metodología a un sistema real a través de una tarjeta de adquisición de datos y Matlab para capturar los datos de la planta y realizar una comparación entre ambas técnicas.

♣ W. Wang, J. Yao and S. Qiu “Implementation of Kalman Filter in PLC” in International Symposium on Big Data and Artificial Intelligence. Hong Kong, Hong Kong. 2018, pp 242-245, [13].

Este documento detalla una implementación del Filtro de Kalman en un PLC de la marca SIEMENS S7-1200 para la detección de mediciones de temperatura. Se realiza un resumen del algoritmo del Filtro de Kalman y sus ecuaciones en su forma recursiva, explicando el proceso de predicción y corrección del método, para luego abordar el problema de memoria que tiene el método, así como resolver las operaciones matriciales que los PLC no soportan y dar una solución a la hora de aplicar las ecuaciones de predicción y corrección a texto estructurado. Finalmente se muestran los códigos usados y los modelos simulink usados para verificar los resultados obtenidos, así como los gráficos obtenidos tanto en la simulación como en el PLC.

♣ G.M. Tamiselvan and P. Aathy “Online tuning of fuzzy logic controller using Kalman algorithm for conical tank system,” Department of EIE, Bannari Amman Institute of Technology, Sathyamangalam, India. Mayo 24, 2017, [14].

Este documento detalla las dificultades que tienen los controladores hacia los sistemas que deben controlar, sobre todo cuando la dinámica del sistema involucra tiempos muertos en la entrada y mediciones, parámetros variantes en el tiempo y no linealidades inherentes al sistema que dificulta el trabajo para controladores tipo PID, por lo que se propone un controlador difuso, el cual recibe la ayuda de un Filtro de Kalman para adaptarse al rango de trabajo en el que se mueve el sistema.

♣ D. Beckmann, M. Dagen and T. Ortmaier “Symplectic Discretization Methods for Parameter Estimation of a Nonlinear Mechanical System using an Extended Kalman Filter,” in 13th International Conference on informatics in Control, Automation and Robotics ICINCO 2016. Vol.1, 2016, pp 327-334, [15].

Este documento presenta dos métodos de discretización en el contexto de estimación en línea de parámetros para un sistema mecánico no lineal. Estas aproximaciones son comparadas para establecer el método de discretización considerando precisión y esfuerzo computacional. En adición, se analiza la influencia del método de discretización con el Filtro de Kalman Extendido. La combinación del método de discretización con el Filtro de Kalman Extendido añadiendo una estimación de parámetros otorga valores estimados más precisos.

♣ J. Castañeda, M. Nieto and V. Ortiz “Análisis y Aplicación del Filtro de Kalman a una señal con ruido aleatorio,” *Scientia Et Technical*, Vol. 18, No. 1, abril 2013, [16].

Este artículo contiene una revisión acerca del algoritmo del Filtro de Kalman, dando a conocer el origen de las ecuaciones del Filtro y un análisis matemático para comprender como realizan la estimación desde la matriz de covarianza del error. Se dan a conocer muchas de las aplicaciones del filtro, entre las cuales se encuentra la estimación de parámetros y filtrado de ruido de la señal deseada. En la aplicación realizada por este filtro se simula un sistema de segundo orden en el software Matlab, comentando resultados y dando las conclusiones pertinentes.

♣ F. Auger, M. Hilairret, J. Guerrero, E. Monmasson, T. Orłowska-Kowalska and S. Katsura “Industrial Applications of the Kalman Filter: A Review,” *IEEE Transactions on Industrial Electronics*, Vol. 60, No. 12, Enero 2013, pp 5458-5471, [17].

Este documento trata acerca de las diversas aplicaciones que ha tenido el Filtro de Kalman en la industria. Este filtro ha captado el interés de la comunidad industrial electrónica, y ha jugado un papel clave en muchos campos de la ingeniería desde 1970, en aplicaciones de estimación de trayectoria, estimación de estado y parámetros para control o diagnóstico, procesamiento de señales y datos, entre otros. El documento desarrolla una visión general de las aplicaciones industriales e implementaciones que ha tenido el Filtro de Kalman en los seis tópicos generales de la comunidad industrial electrónica.

♣ L. Sánchez, J. Ordoñez and S. Infante “Filtro de Kalman Extendido y Filtro de Partículas Kalman Extendido para problemas de estimación No lineal,” *Revista Ingeniería UC*, Vol. 20, No. 1, abril 2013, pp 7-16, [18].

Este artículo presenta una metodología basada en los algoritmos del Filtro de Kalman Extendido y Filtro de Partículas Kalman Extendido para estudiar el problema de estimación de parámetros en modelos dinámicos con estructuras no lineales, pero con errores de observación Gaussianos, se plantea un modelo en espacios de estado, donde los estados son tratados como parámetros utilizando técnicas recursivas. Se ilustran las propuestas estimando y reconstruyendo los estados de los mapas de Henón y Lorenz, además de reconstruir señales morfológicas de un electrocardiograma. Los resultados demuestran el buen comportamiento de los filtros en la estimación de estados a través de desviación estándar empírica y costo computacional.

♣ A. Gautam and S. Majumdar “State Estimation of Single-Phase Rectifier Based Load Circuit using Unscented Kalman Filter,” in 2nd IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES-2018), Delhi, India, octubre 2018, [19].

Este documento realiza una aplicación del Filtro de Kalman Suavizado a un rectificador de puente completo modelado en el espacio de estados. Se realiza una introducción a los algoritmos del Filtro de Kalman Extendido y Filtro de Kalman Suavizado, explicando detalladamente el procedimiento requerido para la predicción y corrección de ambas técnicas para luego realizar una comparación entre ambas técnicas y los resultados que éstas entregan, ya que el Filtro de Kalman Suavizado tiene su origen desde la transformada suavizada y el Filtro de Kalman Extendido desde una expansión en series de Taylor para linealización de un sistema no lineal, detallando la ventaja computacional del Filtro de Kalman Suavizado al no requerir jacobianos.

♣ R. Choomuang and N. Afzulpurkar “Hybrid Kalman Filter/Fuzzy Logic based Position Control of Autonomous Mobile Robot” in Internetional Journal of Advanced Robotic Systems, Vol. 2, No. 3, 2005, pp 197-208, [20].

Este documento describe el control de posición de un robot móvil autónomo usando Filtro de Kalman y técnicas de lógica difusa. Ambas técnicas se usan para fusionar información de sensores internos y externos para la navegación del robot en un ambiente desconocido. Se implementa un algoritmo de esquite de obstáculos utilizando visión estéreo. La principal preocupación de los sistemas de navegación es lograr un rendimiento robusto en tiempo real para el control del movimiento del robot, dando una posición y orientación precisos para el movimiento del robot.

1.2.3. Teoría y Aplicación de Controladores

♣ C. Chuen Lee “Fuzzy Logic in Control Systems: Fuzzy Logic Controller – Part I” in Transaction on Systems, Man, and Cybernetics, Vol. 20, No. 2, 1990, pp 404-418, [21].

Este documento describe los inicios de la lógica difusa y la forma en que realiza el control sobre la planta, enfocándose en métodos que tratan con datos cualitativos relacionando la entrada y salida, acercando la lógica de toma de decisiones lo más cerca posible de lo que haría un operador humano. Se ilustran la teoría de la lógica difusa, centrándose en la estructura del controlador y las definiciones que son necesarias para la construcción de un controlador, como lo son los sets difusos, funciones de membresía, fuzzificación y reglas de inferencia y defuzzificación, así como sus respectivas ecuaciones.

♣ C. Chuen Lee “Fuzzy Logic in Control Systems: Fuzzy Logic Controller – Part II” in *Transaction on Systems, Man, and Cybernetics*, Vol. 20, No. 2, 1990, pp 419-435, [22].

Este documento es la continuación de la descripción de la caracterización de las diversas secciones que componen un controlador difuso. Se da un resumen de los puntos anteriormente revisados acerca de las reglas difusas y sus diversas aplicaciones. Se da a conocer los métodos de defuzzificación existentes y sus ecuaciones para lograr la conversión de términos difusos a términos que son legibles para el sistema que está destinado a ser controlado.

♣ Z. Gao, T. Trautzsch and J. Dawson “A Stable Self-Tuning Fuzzy Logic Control System for Industrial Temperature Regulation” in *Transactions on Industry Applications*, Vol. 38, No. 2, 2002, pp 414-424, [23].

Este documento describe la aplicación de un controlador difuso a un sistema de regulación de temperatura con una pequeña base de reglas, demostrando el potencial de este tipo de controladores implementados tanto en simulación como en un proceso real. Se da una leve introducción a la teoría necesaria para definir cada una de las secciones que forman parte del controlador, profundizando en cada una de ellas y dando a modo de ejemplo los utilizados para la aplicación realizada. Se entrega finalmente los resultados aplicados en diferentes procesos de temperatura, demostrando la robustez del controlador en los casos aplicados.

♣ A. Aissaoui and A. Tahour “*Application of Fuzzy Logic in Control of Electrical Machines, Fuzzy Logic – Control, Concepts, Theories and Applications*”, Prof. Elmer Dadios. Croatia: InTech, 2012, [24].

Este documento describe la definición de la base de reglas de un controlador difuso, centrándose en la forma en que opera el control según el número de reglas y términos difusos que compondrán al controlador, resumiendo estas reglas en matrices que representan las posibles decisiones que el controlador puede tomar. Se dan a conocer las distintas funciones de membresía generalmente usadas para la etapa de fuzzificación y defuzzificación. Finalmente se aplica esta teoría a un sistema eléctrico, realizando las comparaciones entre los controladores realizados con diferentes términos lingüísticos.

♣ “Apuntes de Sistemas Lineales Dinámicos”, 543 214, Departamento de Ingeniería Eléctrica, Universidad de Concepción, Marzo 2018, [25].

Estos apuntes contienen amplia información acerca de sistemas lineales y teoría aplicada a ellos. La información encontrada en este documento será utilizada con el fin de obtener un modelo discreto que el PLC pueda trabajar de forma iterativa, para finalmente representar dicho modelo en espacios de estado, que es la estructura que solicita el Filtro de Kalman Extendido.

1.3. Discusión

Las aplicaciones realizadas en procesos industriales nunca han sido una tarea evidente o sencilla, ya que, al ser procesos de gran tamaño y equipos de funcionamiento totalmente diferente, sumados a no linealidades inherentes al sistema, parámetros que varían en el tiempo debido a no linealidades y tiempos muertos o retrasos de la planta, hacen que sistemas de control de estos procesos enfrenten desafíos al realizar las rutinas de la planta diariamente.

Desde su aparición, el Filtro de Kalman ha sido objetivo de diversas aplicaciones en muchos campos diferentes, desde estimación de trayectorias, filtrado de ruido, estimación de parámetros y estados, entre otras. Sin embargo, todos estos desarrollos tienen en común la necesidad de realizar una estimación del estado para realizar un control mucho más eficiente y un mejor comportamiento del sistema a estimar en tiempo real, obteniendo una mejor comprensión del proceso que se está trabajando, sea cual sea el campo de aplicación.

En base a la revisión bibliográfica, se espera que la implementación de un Filtro de Kalman Extendido a la planta piloto ubicada en el laboratorio de control de procesos ayudaría a obtener estimaciones de las señales tomadas del proceso, siendo éstas más limpias de los distintos sensores, facilitando el acceso a las mediciones del proceso cuando sea requerido y ayudar a los sistemas de control a realizar sus tareas con mayor eficiencia al obtener una señal estimada del proceso que solucione el ruido y el tiempo muerto de la planta al acceder a estimaciones provenientes del Filtro de Kalman Extendido, dando como resultado que las señales entregadas a las válvulas posean menos ruido.

1.4. Hipótesis de Trabajo

- Es viable la implementación de un Filtro de Kalman Extendido en un PLC en tiempo real
- El modelo realizado podrá ajustarse al tiempo muerto del sistema.
- El ruido en las mediciones se verá disminuido.

1.5. Objetivos

1.5.1. Objetivo General

Diseñar e implementar un Filtro de Kalman Extendido para la planta piloto, y diseñar un controlador difuso basado en el modelo estimado.

1.5.2. Objetivo Específicos

- Desarrollar y estudiar los métodos y algoritmos actuales del Filtro de Kalman.
- Obtener modelos discretos de los lazos de flujo, nivel y temperatura de la planta piloto.
- Desarrollar un algoritmo del Filtro de Kalman Extendido y aplicarlo a la planta piloto.
- Desarrollar controladores difusos para los lazos de la planta piloto respecto a la salida estimada del Filtro de Kalman Extendido

1.6. Alcances y Limitaciones

- El modelo base a entregar al Filtro de Kalman Extendido será SISO o MISO según corresponda.
- Los algoritmos serán diseñados en Matlab e implementados en el PLC a través de Rslogix 5000.
- El ruido de proceso y medida será considerado ruido blanco gaussiano no correlacionado.
- Solo se implementará el Filtro de Kalman Extendido y Controladores Difusos.

1.7. Temario y Metodología

En el capítulo 1, se pone en contexto sobre los problemas que pueden estar involucrados en la toma de mediciones de las entradas y salidas de un proceso industrial y como éstos pueden afectar a los procesos de control e identificación a desarrollar. Luego se desarrollará la hipótesis junto con los objetivos y alcances del desarrollo del tema de memoria para dar solución a este problema.

En el capítulo 2 se cubre la etapa de modelación de la planta piloto, obteniendo un modelo para los lazos que componen la planta para posteriormente ser usado en el algoritmo que se desarrollará, implicando la estructura base a usar y los elementos que se usarán para llegar a la función final que se implementará en el PLC.

En el capítulo 3 se cubre la selección estructural del filtro, en el cual se harán las pruebas con la planta para evaluar la respuesta que éste nos entregará, enfocándose principalmente en el comportamiento de los parámetros del sistema, que implicará para la etapa posterior a desarrollar y se mostrarán pruebas de la respuesta que entregará el filtro contra la respuesta del sistema. Se hará una introducción al Filtro de Kalman Extendido y las expresiones algebraicas de predicción y

corrección que se programarán, además de realizar una revisión de las otras técnicas del Filtro de Kalman existentes en la actualidad.

En el capítulo 4 se continúan desarrollando la estructura del filtro desde el punto de vista de los parámetros, es decir, como estos variarán sus valores según la respuesta del sistema, lo que llevará a desarrollar una estrategia para obtener los parámetros del sistema e incluirlos al proceso de filtrado del algoritmo del Filtro de Kalman Extendido a desarrollar, con el fin de lograr la identificación.

En el capítulo 5 se desarrollan los controladores necesarios para controlar la planta piloto, esta vez desde la salida del Filtro de Kalman Extendido. Se explicará cómo el algoritmo entregará su salida al controlador, así como todo el procesamiento necesario que realizará el controlador para calcular la variable manipulable y lograr que el sistema siga las instrucciones de quién esté operando la planta.

En el capítulo 6 se muestran los resultados obtenidos del algoritmo del Filtro de Kalman Extendido y el controlador desarrollado implementados en la planta piloto, comprobando el correcto funcionamiento de los algoritmos propuestos.

En el capítulo 7 se dan las conclusiones obtenidas a partir del trabajo realizado durante el desarrollo de la implementación de las funciones y lazos programados en el PLC, así como posibles continuaciones que podría tener el tema abordado en este informe.

Capítulo 2. Modelación de la planta piloto

La primera tarea a realizar al momento de comenzar a trabajar en un sistema, es identificar una estructura que permita modelar el comportamiento del proceso, a fin de simular las estrategias a implementar, en nuestro caso, verificar que la estrategia del Filtro de Kalman Extendido se ajusta correctamente a los datos que nos entrega el sensor, [1, 7, 19].

Muchos sistemas en la industria son modelados como sistemas de primer o segundo orden con retardo, de acuerdo con la dinámica del sistema y la planta piloto no es la excepción. Modelos de primer o segundo orden a menudo se escogen por su sencillez de implementación y fácil manipulación, además de los sistemas no responden de forma instantánea al momento de realizar cambios en la entrada, es decir, que poseen retardo, [9]. En este caso, los lazos de la planta piloto serán modelados de acuerdo con la siguiente ecuación:

$$h(s) = \frac{k_m}{\tau \cdot s + 1} e^{-\theta s} \quad (2.1)$$

Donde k_m es la ganancia del sistema, τ es la constante de tiempo y θ es el retardo del sistema. Según sean las características del proceso, estos valores tendrán cambios durante la dinámica de la planta, en lo que habrá que tomar especial cuidado.

En cuanto al retardo del sistema θ , según sea la aplicación, puede presentar leves variaciones que deberán considerarse a la hora de identificar el modelo, por lo que en este caso se aproximará esta expresión de acuerdo a la expresión (2.1). De Sánchez [9] encontramos dos formas de aproximar el retardo, siendo éstas las aproximaciones de padé de primer y segundo orden, y el método NS4-2. Las expresiones asociadas a estos métodos son las siguientes:

$$e^{-\theta s} = \frac{1 - \frac{\theta s}{2}}{1 + \frac{\theta s}{2}} \quad (2.2)$$

$$e^{-\theta s} = \frac{1 - \frac{\theta s}{2} + \frac{\theta^2 s^2}{12}}{1 + \frac{\theta s}{2} + \frac{\theta^2 s^2}{12}} \quad (2.3)$$

$$e^{-\theta s} = \frac{1}{\left(1 + \frac{\theta s}{n}\right)^n} \quad (2.4)$$

Sánchez menciona que la aproximación NS4-2 realizada a partir de retardos consecutivos de primer orden introduce menos error cuadrático (SSE) en comparación a las aproximaciones de padé, lo cual queda ejemplificado según la siguiente tabla:

Aproximación	SSE
Padé 1° Orden	5.643
Padé 2° Orden	3.5
NS4-2 $n = 1$	4.842
NS4-2 $n = 2$	3.341

Tabla 2.1. Errores cuadráticos extraídos de [9], Autor: Sánchez

De acuerdo con la tabla 2.1, se aprecia que mientras más alto sea el valor de n , menor será el error cuadrático medio introducido por la aproximación. Sin embargo, el autor menciona que la tasa de decremento del SSE comienza a decaer mientras más grande sea el valor de n , por lo que solo es conveniente usar $n = 2$, el cual será el valor que se usará para aproximar el retardo. De esta forma, obtendremos la siguiente F. de T.:

$$h(s) = \frac{k_m}{(\tau \cdot s + 1)(1 + \frac{\theta s}{2})^2} \Bigg|_Z \quad (2.5)$$

Así, con la ecuación (2.5) será la utilizada para modelar los lazos de flujo, nivel y temperatura. Sin embargo, a la hora de realizar una implementación en PLC, estas ecuaciones deberán introducirse en texto estructurado, los cuales se ejecutarán en intervalos de tiempo fijos, razón por la que el modelo de la ecuación (2.5) deberá entregarse en tiempo discreto. De esta forma, a partir de [25], en el plano z , esta F. de T. queda de la siguiente forma:

$$H(z) = \frac{K0 + K1 \cdot z^{-1} + K2 \cdot z^{-2} + K3 \cdot z^{-3}}{1 + K4 \cdot z^{-1} + K5 \cdot z^{-2} + K6 \cdot z^{-3}}, \quad \text{para } T_o \geq \frac{\theta}{3} \quad (2.6)$$

Luego, los parámetros del sistema serán $K0$ hasta $K6$, los que a su vez dependerán de los parámetros de la F. de T. en tiempo continuo k_m, τ y θ . Según sea la dinámica del sistema dictada por estos parámetros, deberá realizarse un proceso de identificación para obtener dichos valores y obtener la salida del sistema y los parámetros de éste, lo que añadirá ecuaciones adicionales al algoritmo encargado del proceso de filtrado de la salida del sistema, pero aumentando la cercanía del valor estimado al valor real. En el Anexo A se encuentra el proceso realizado para obtener dichos parámetros, así como sus expresiones en función de los parámetros de la función de transferencia de tiempo continuo.

Capítulo 3. Selección estructural del filtro

3.1. Introducción al Filtro de Kalman

En 1960, Rudolf Kalman publicó un método para realizar estimaciones sobre sistemas lineales en base a los datos entregados por el sistema o equipo en que fuera implementado, conociéndose posteriormente como Filtro de Kalman. Esta técnica nació con el objetivo de dar solución a los problemas sobre predicción de señales, separación del ruido de las señales y detección de señales conocidas en presencia de ruido, [3].

El método tiene una naturaleza recursiva, lo que le permite ser aplicado a sistemas digitales en diversas aplicaciones en las que se pueda desenvolver, entre las que se encuentran trabajos sobre señales eléctricas, navegación y posicionamiento, sistemas hidráulicos, entre otros, [14, 19, 20]. El Filtro de Kalman realiza estimaciones y predicciones del sistema basándose en un modelo en espacios de estado que represente la dinámica del sistema, dando como resultado una estimación de la salida del sistema a través de un filtrado óptimo de los datos de entrada, [16]. Si la situación lo requiere, los parámetros del sistema pueden ser identificados de forma paralela y añadidos al proceso de filtrado, [4], lo que será tratado en el siguiente capítulo.

La solución entregada por Rudolf Kalman ayuda a resolver estos problemas, de forma que mejora la calidad de las señales, [17]. En este proyecto se busca implementar uno de los Filtros de Kalman existentes en la literatura para entregar una solución a los problemas de tiempo muerto en los sistemas y ruido presente en las mediciones de la planta piloto para las variables de flujo, nivel y temperatura que entregan los sensores.

Desde que el Filtro de Kalman original fue publicado, muchos autores han propuesto mejoras al algoritmo y ampliado su rango de aplicaciones a sistemas no lineales, realizando ecuaciones mucho más robustas y mejorando los resultados que el filtro tiene para ofrecer. Entre las soluciones existentes, se encuentra el Filtro de Kalman Extendido, el Filtro de Kalman Suavizado y el Filtro Partícula de Kalman Extendido, [18, 19]. La solución a desarrollar en este proyecto será el Filtro de Kalman Extendido, ya que requiere muchos menos recursos computacionales que las otras dos versiones disponibles del filtro, [17, 19], y es simple de implementar por su baja complejidad.

3.2. Modelo en espacios de estado

Entre los requisitos que el Filtro de Kalman Extendido, se encuentra un modelo en ecuaciones de estado, el cual sirve para obtener las matrices de las ecuaciones de proceso y medida del sistema para el proceso recursivo de las ecuaciones. Retomando la ecuación (2.6), deberá reorganizarse la

estructura para que coincida con la forma en que se solicita el modelo en el Filtro de Kalman Extendido. De esta forma, el modelo debe ser un modelo en espacios de estado:

$$x_k = A \cdot x_{k-1} + B \cdot u_{k-1} + w_{k-1} \quad (3.2.1)$$

$$y_k = H \cdot x_k + v_k \quad (3.2.2)$$

Así, según [25], es posible obtener un modelo previo en ecuaciones de estado. Hay que considerar que esto es suponiendo que los parámetros son constantes y el modelo es lineal, lo que posteriormente en el capítulo 4 se verá que no es así y deberán ser tratados en consecuencia para un nuevo modelo que se usará, pero éste nos servirá para definir el modelo no lineal para el algoritmo posteriormente.

3.3. Filtro de Kalman Extendido

El Filtro de Kalman Extendido, abreviado EKF por sus siglas en inglés, surgió como alternativa para permitir su aplicación en sistemas que presentaran no linealidades producto de los equipos que componen el sistema o la física misma de un proceso, entre otras razones.

Derivado del Filtro de Kalman inicial, en [3] se realizan modificaciones a las condiciones iniciales del filtro y al set de ecuaciones de predicción y corrección que realiza el proceso de estimación en la señal deseada. Esto permite que sin importar la dinámica que posea el proceso, mientras se cumplan los requerimientos del Filtro de Kalman Extendido, realizando el proceso de estimación de la señal de forma iterativa, [1, 2].

De forma similar a su homólogo para sistemas lineales, es necesario que los datos sean organizados en un modelo en espacios de estado, relacionando de esta forma los datos actuales con los datos pasados en forma de estados de un sistema, lo que permite realizar el cálculo de las matrices necesarias para las ecuaciones que componen el EKF. Las ecuaciones necesarias para comenzar el proceso iterativo deben tener esta forma:

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \quad (3.3.1)$$

$$y_k = h(x_k, v_k) \quad (3.3.2)$$

Donde f representa a la función de proceso y h representa a la función de medida. La primera ecuación tiene como objetivo relacionar los estados x_k actuales con los estados x_{k-1} pasados y la entrada manipulable u_{k-1} , además de que dicha función estará en presencia de ruido w_{k-1} , el cual supondremos que será ruido gaussiano blanco no correlacionado. La relación entre estas variables, en este caso, es de tipo no lineal, lo que hace que la relación entre las variables ya mencionadas no

sea clara. Además, la función de medida h tiene como objetivo relacionar los estados x_k obtenidos a partir de la función de proceso con la salida y_k del sistema, mientras que dicha relación también se encuentra en presencia de ruido, el cual se supone ruido blanco gaussiano no correlacionado. Dicho esto, estas funciones f y h será la base para obtener las matrices necesarias para el proceso recursivo de las ecuaciones, las cuales se dividirán en un set de ecuaciones destinadas a predecir la salida y otro destinado a corregir el valor predicho para obtener finalmente una estimación del sistema, en el cual las matrices jugarán un papel importante.

Las Matrices que el EKF necesita para sus ecuaciones se obtendrán a partir de Jacobianos, [2]. Esos Jacobianos tendrán dimensiones directamente relacionadas con la cantidad de estados y salidas que el sistema posea. La función que deben cumplir estas matrices es la misma que las funciones ya descritas, solo que esta vez dichas matrices deben ser obtenidas aplicando jacobianos a las matrices f y h , [2, 11]. Debido a la naturaleza recursiva de las ecuaciones, estas matrices deberán obtenerse en cada iteración del algoritmo, de forma de ajustar la predicción, lo que también nos lleva a que solo necesitaremos las matrices A y H que relacionarán los estados actuales, estados pasados y la salida, que se obtienen de la siguiente manera:

$$A = \left. \frac{\partial f_i(x, u_k)}{\partial x} \right|_{x,u} \quad (3.3.3)$$

$$H = \left. \frac{\partial h_i(x, u_k)}{\partial x} \right|_{x,u} \quad (3.3.4)$$

De [2] y [19] podemos observar que las matrices A y H deben ser evaluadas por los estados *a posteriori* y *a priori*, respectivamente, en cada iteración, donde el estado *a priori* es el estado resultante de las ecuaciones de predicción y el estado *a posteriori* el resultado de las ecuaciones de corrección. Ya que la matriz A será utilizará para el cálculo de la matriz de covarianza, debe ser evaluada en los estados *a posteriori* anteriores \hat{x}_{k-1} . En el caso de la matriz H, esta debe ser evaluada en los estados actuales *a priori* \hat{x}_k^- . Para los cálculos y ecuaciones posteriores, ambas matrices serán utilizadas en cada iteración con el fin de obtener la estimación deseada de la salida del sistema.

Como ya se ha mencionado, el ruido que afecta al proceso y a las mediciones, designado como w_k y v_k respectivamente, se supondrá ruido gaussiano blanco no correlacionado entre sí, [8]. De esta forma, ambas señales de ruido afectarán a las funciones de proceso y medida incluso después del cálculo de los jacobianos, a través de las matrices Q_k y R_k . Así, el efecto del ruido puede

expresarse a partir de lo siguiente:

$E[w_k] = 0$	$E[w_k w_k^T] = Q_k$	$E[w_k w_j^T] = 0$ si $k \neq j$	$E[w_k x_0^T] = 0$
$E[v_k] = 0$	$E[v_k v_k^T] = R_k$	$E[v_k v_j^T] = 0$ si $k \neq j$	$E[v_k x_0^T] = 0$

Luego, el ruido gaussiano no correlacionado se define de la siguiente manera:

$$E[w_k v_j^T] = 0 \text{ para todo } k \text{ y } j$$

La dimensión de las matrices Q_k y R_k , que representarán a las matrices de covarianza del ruido de proceso y ruido de medida respectivamente, será de $n \cdot n$ y $n \cdot p$, con n la cantidad de estados y p la cantidad de salidas que tendrá el algoritmo. Ya que el ruido no está correlacionado, las matrices Q_k y R_k serán matrices diagonales, las cuales pueden ser constantes o variar su valor en cada iteración al realizar la aplicación del filtro, [1, 8].

El Filtro de Kalman Extendido realiza las estimaciones sobre los estados y la salida a partir de dos sets de ecuaciones, uno destinado a predecir el valor que tomará la señal y otro a corregir el valor que se predijo en el primer set de ecuaciones. Este set de ecuaciones es el siguiente:

$$\hat{x}_k^- = f(\hat{x}_{k-1}^-, u_{k-1}, 0) \quad (3.3.5)$$

$$P_k^- = A_k P_{k-1} A_k^T + Q_{k-1} \quad (3.3.6)$$

$$K_k = P_k^- H^T (H P_k^- H^T + R_k)^{-1} \quad (3.3.7)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - h(\hat{x}_k^-, 0)) \quad (3.3.8)$$

$$P_k = (I - K_k H) P_k^- \quad (3.3.9)$$

La filosofía de ambos sets de ecuaciones, a través de datos de entrada y salida de la planta, en nuestro caso, la apertura de la válvula y el sensor de medición, las ecuaciones (3.3.5) y (3.3.6) se encargan de predecir los estados *a priori* y la matriz de covarianza *a priori*. Esto se realiza a través de la función f de proceso del sistema, en la que los estados pasados y la entrada del sistema son introducidos y se realiza una predicción del valor actual que tomará el estado. El grado en que afecte el error a las mediciones se realiza a través de la matriz de covarianza del error *a priori*.

Luego de que se realiza el cálculo de los estados *a priori*, se aplica el segundo set de ecuaciones denominado ecuaciones de corrección, las que se encargan de corregir el valor de acuerdo al ruido que haya presente a través de la matriz de covarianza del error P_k . De esta forma, finalmente se obtiene el error *a posteriori* y la estimación final del filtro, [5]. Con el estado *a posteriori* y la matriz de covarianza del error, las ecuaciones (3.3.7), (3.3.8) y (3.3.9) realizan la corrección del estado obtenido anteriormente, de forma tal que el estado resultante del segundo set de ecuaciones tendrá

menos efecto de ruido y será el que se usará para obtener la salida del sistema.

Para construir esta estrategia, se utilizó Matlab en conjunto con datos de prueba para comprobar que las ecuaciones del Filtro de Kalman Extendido realizan una estimación de la salida, manteniendo la dinámica al tiempo que disminuye el efecto del error. Los resultados de la simulación off-line se encuentran en la siguiente figura:

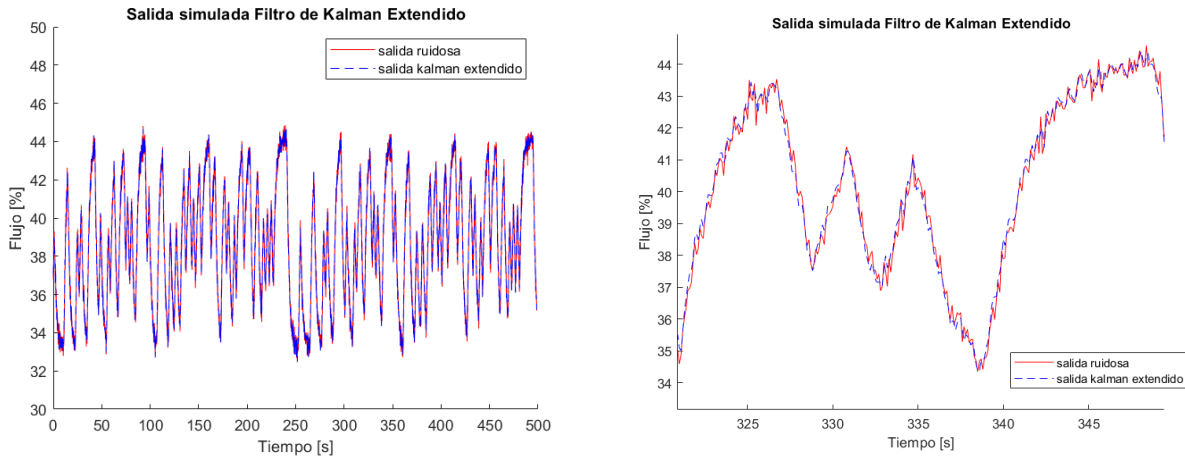


Figura 3.3.1. Salida simulada con ruido v/s Salida aproximada mediante EKF

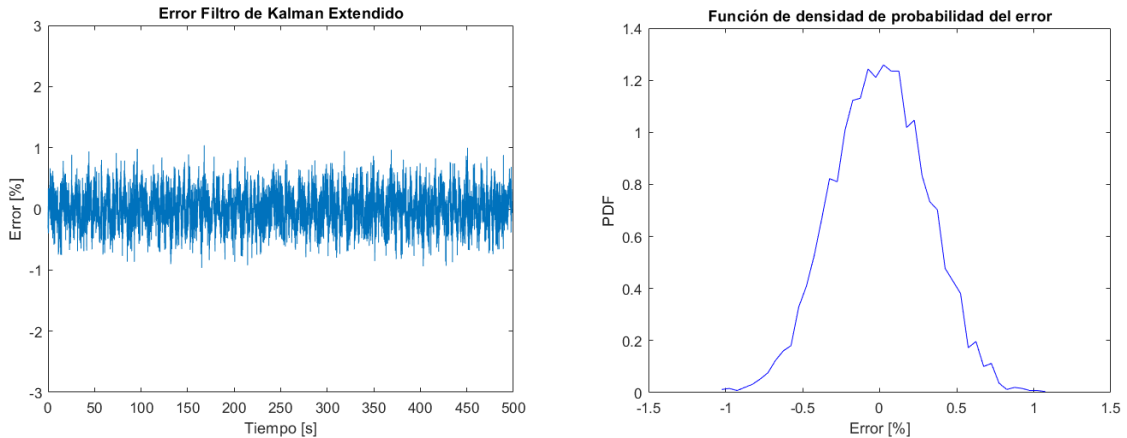


Figura 3.3.2. Error del Filtro de Kalman Extendido

Como se aprecia en las imágenes, es posible ver el efecto predictor y corrector de las ecuaciones que componen el Filtro de Kalman Extendido. La salida es capaz de seguir la trayectoria y respuesta original de los datos introducidos, en este caso, que corresponden al flujo de salida de la planta, los cuales son filtrados de las interferencias que puedan poseer o transmitirse durante la medición. La línea continua representa a la señal ruidosa del sensor, y la señal segmentada representa a la salida del filtro desarrollado, observándose una diferencia entre ambas señales, principalmente debido al trabajo del Filtro de Kalman Extendido en los datos, eliminando el ruido presente en las mediciones, ajustándose al mismo tiempo a la dinámica de la planta. El ajuste en la

dinámica realizado por las ecuaciones de predicción, y luego ajustando el filtrado con las ecuaciones de corrección, cerrándose el proceso iterativo del filtro, [2, 11]. De esta forma, con algunos cambios, el filtro es capaz de realizar este mismo proceso en tiempo real.

De forma similar a este desarrollo, existen otras alternativas al filtro de Kalman extendido, las cuales no se basan totalmente en el uso de jacobianos para realizar la estimación de los estados. Dos de estas alternativas son el Filtro de Kalman Suavizado y el Filtro Partícula de Kalman Extendido. Ambos filtros utilizan técnicas distintas para realizar el mismo procedimiento de predicción y corrección. El Filtro de Kalman Suavizado, abreviado UKF, utiliza la transformada suavizada en lugar de jacobianos, lo que reduce más el error en la estimación, [7]. A su vez, el Filtro Partícula utiliza tanto jacobianos como muestreo de datos, [10, 18]. Esta forma de trabajar permite estimar la señal de salida y reducir aún más el ruido e interferencias presentes, aunque aumenta la carga computacional que produce el filtro en el sistema en el que se programan las ecuaciones. De ambas soluciones, el Filtro Partícula pide menos recursos que el Filtro de Kalman Suavizado, lo que lo convierte en una opción a considerar si los sistemas disponibles no tienen la potencia necesaria para llevar a cabo el Filtro de Kalman Suavizado. En el Anexo B. se encuentra con más detalle la teoría y forma de trabajo de ambos filtros para la estimación de las señales.

Capítulo 4. Identificación estructural

4.1. Identificación de parámetros del sistema

La identificación de parámetros en un sistema es importante, pues permite determinar la dinámica de la planta, y confirmar si la característica de esta es lineal o no lineal. Recordando los parámetros k_m , τ y θ , respectivamente la ganancia, constante de tiempo y retardo del sistema en base a la F. de T. en tiempo continuo, los parámetros K0 a K6 tendrán un comportamiento constante o variable. En caso de ser parámetros variables, se añade una complejidad adicional, pues deberá realizarse un proceso de identificación. En este caso, se puede apreciar en la siguiente figura la respuesta de la válvula de descarga de la planta:

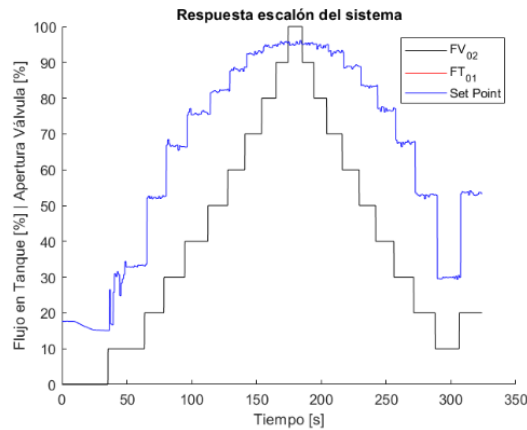


Figura 4.1.1. Respuesta del sistema ante entradas escalón

Donde es posible observar que mientras aumenta el porcentaje de apertura de la válvula en un rango de 0% a 100%, la ganancia del sistema cambia, decreciendo su valor mientras la válvula más abierta se encuentra, demostrándose que la ganancia k_m es variable, lo que hará que los parámetros del sistema discreto K0 hasta K6 sean variables.

Mínimos Cuadrados Recursivos es una estrategia que permite la identificación de parámetros en línea con cada iteración que realice el PLC. Además, incorporando un factor de olvido, permitirá olvidar información antigua y mejorar el proceso de identificación, necesitándose de esta forma datos de entrada y salida del sistema en condiciones de funcionamiento, [6]. De esta forma, se podrá capturar los valores de los parámetros K0 hasta K6 del sistema discreto. De esto, se tendrá que la estructura base para comenzar el proceso de identificación mediante mínimos cuadrados será:

$$y_k = -(K4 \cdot y_{k-1} + K5 \cdot y_{k-2} + K6 \cdot y_{k-3}) + K0 \cdot u_k + K1 \cdot u_{k-1} + K2 \cdot u_{k-2} + K3 \cdot u_{k-3} \quad (4.1.1)$$

Esta expresión será la base inicial para el proceso de identificación, obteniendo un valor para los parámetros, que posteriormente deberán ajustar sus valores según el modelo y algoritmo del

Filtro de Kalman, [6]. Para comenzar el proceso iterativo de identificación, es necesario definir primero la matriz de regresores y vector de parámetros identificados, los cuales se rigen según la siguiente ecuación:

$$y_k = \varphi_k \cdot \theta_k \quad (4.1.2)$$

Ecuación que servirá para realizar un seguimiento del proceso real y realizar una buena identificación. Definiendo el error de predicción como $e_k = y_k - \tilde{y}_k$ como la resta entre el valor real y el valor estimado de la salida, junto con la ecuación (4.1.2), se definen las ecuaciones que componen el algoritmo inicial de identificación, [6]:

$$Pi_k = \frac{1}{\lambda} \left[P_{k-1} - \frac{Pi_{k-1} \cdot \varphi_{k-1} \cdot \varphi_{k-1}^T \cdot Pi_{k-1}}{\lambda + \varphi_{k-1}^T \cdot Pi_{k-1} \cdot \varphi_{k-1}} \right] \quad (4.1.3)$$

$$Ki_k = Pi_k \cdot \varphi_k \quad (4.1.4)$$

$$\theta_k = \theta_{k-1} + Ki_k \cdot (y_k - \varphi_{k-1}^T \cdot \theta_{k-1}) \quad (4.1.5)$$

Donde Pi_k es la matriz de covarianza, Ki_k es la ganancia de corrección y θ_k son los parámetros ya identificados mediante RLS. Notar que la matriz Pi_k se inicia con un valor diagonal grande y será de dimensión $nu \cdot nu$, con nu el total de parámetros a identificar. Debido a la recursividad del método, será necesario de datos de entrada y salida en los tiempos $(k - 1)$, $(k - 2)$ y $(k - 3)$ de la matriz de regresores, obteniendo así la ganancia de corrección en cada iteración para la obtención de los parámetros de la planta, de forma que se realizará un seguimiento de los parámetros de la planta.

4.2. Identificación simultánea de parámetros y estados con el Filtro de Kalman Extendido

Si bien el Filtro de Kalman Extendido permite realizar una estimación de la salida del sistema, también es de interés añadir a la estimación los parámetros del sistema, pues estos parámetros son obtenidos a partir de los datos del sensor de la planta, cuya señal aún no ha sido tratada a través del EKF, [5]. Esto puede resultar de interés si existe variación en los valores de los parámetros del sistema, como también aplicable para detectar fallas o deterioro del sistema, [12]. En este caso, el sistema inicial sería el siguiente:

$$x_k = f(x_{k-1}, u_{k-1}, a, w_{k-1}) \quad (4.2.1)$$

$$y_k = h(x_k, b, v_k) \quad (4.2.2)$$

Con a y b siendo los parámetros presentes en las ecuaciones de proceso y medida, respectivamente. Para añadir los parámetros a la estimación, es necesario realizar una “extensión” de los estados del sistema, al considerar los parámetros como nuevos estados que formarán parte de las ecuaciones del modelo en espacios de estado, de forma que esto se refleja como:

$$\begin{pmatrix} x_k \\ \theta_k \end{pmatrix} = z_k \quad (4.2.3)$$

De esta forma, los parámetros del sistema pasarán a ser nuevos estados, formando así un sistema no lineal. Las ecuaciones de predicción y corrección del Filtro de Kalman Extendido podrán aplicarse de forma similar, solo que se deberán tomar consideraciones adicionales, ya que se deben agregar los nuevos estados a las ecuaciones del modelo en espacios de estado:

$$a_k = a_{k-1} \quad (4.2.4)$$

$$b_k = b_{k-1} \quad (4.2.5)$$

Con estas ligeras modificaciones al sistema, será posible aumentar la precisión de la identificación, disminuyendo también el ruido presente al realizar una estimación de los parámetros, al mismo tiempo que se toman acciones similares con la salida del sistema. De esta forma, las ecuaciones nuevas de proceso y medida serán:

$$z_k = f \begin{pmatrix} x_{k-1}, u_{k-1}, a_{k-1}, w_{k-1} \\ a_{k-1} \\ b_{k-1} \end{pmatrix} \quad (4.2.6)$$

$$y_k = h(x_k, b_k, v_k) \quad (4.2.7)$$

Al realizar los cambios respectivos señalados en las ecuaciones (4.2.4), (4.2.5), se tendrá una estimación tanto de los estados como de los parámetros que definen al sistema. En el caso de los parámetros, se necesita el valor real de los parámetros obtenido a partir del proceso de identificación primario para obtener la innovación y realizar el ajuste mediante el EKF. Ya que, en el caso de este proyecto, los parámetros K0 hasta K6 son resultado de una aproximación matemática, no tienen un significado físico, por lo que se obtendrán estos parámetros a través de RLS. Finalmente, las matrices y parámetros de inicio del algoritmo se definirían en función de los estados y los parámetros iniciales dados, lo que se da a continuación:

$$\hat{x}_0 = [\hat{x}_{init}^T \quad \theta_{init}^T]^T \quad (4.2.18)$$

$$P_0 = \begin{bmatrix} P_{init} & 0 \\ 0 & \theta_{init} \end{bmatrix} \quad (4.2.19)$$

$$Q = \begin{bmatrix} Q_x & 0 \\ 0 & Q_\theta \end{bmatrix} \quad (4.2.20)$$

$$R = \begin{bmatrix} R_y & 0 \\ 0 & R_\theta \end{bmatrix} \quad (4.2.21)$$

Como se puede ver, los vectores y matrices aumentan en tamaño para permitir el inicio del proceso de estimación para los estados y los parámetros de forma conjunta, [15]. Junto con esto, la

salida del Filtro de Kalman Extendido serán los parámetros, y lo que resta de momento es la reconstrucción de la salida del sistema, es decir, la señal de medición de los sensores FT-01, LT-01 y TT-01 ya tratada por el filtro, para lo cual se utilizará la siguiente ecuación, correspondiente a la ecuación de medida definida en (4.2.7):

$$z_k = h(\hat{x}_k, 0) \quad (4.2.22)$$

Recordando previamente que, al realizar la ampliación, solo los estados x_1, x_2 y x_3 corresponderán a los estados relacionados a la señal de salida del sistema, y los estados x_4 al x_{10} corresponderán a los parámetros ya filtrados por el algoritmo del EKF. De esta forma, reconstruyendo la salida, se tiene que:

$$y_k = (K3 - K6 \cdot K0) \cdot x_{1k} + (K2 - K5 \cdot K0) \cdot x_{2k} + (K1 - K4 \cdot K0) \cdot x_{3k} + K0 \cdot u_k \quad (4.2.13)$$

Así, completando el algoritmo recursivo, el cual será capaz de estimar simultáneamente parámetros y la señal de salida del sistema.

4.3. Aplicación del Filtro de Kalman al Sistema en Modo Manual

Considerando las ecuaciones del Filtro de Kalman Extendido realizando la estimación de la salida, se realiza la programación del Filtro de Kalman Extendido en formato de texto estructurado en el PLC del Laboratorio de Control de Procesos. La primera sección del algoritmo desarrollado corresponde a identificación de parámetros mediante RLS para obtener los parámetros K0 hasta K6 del sistema, para luego ser tratados junto con los estados del sistema discreto desarrollado a través de un Filtro de Kalman Extendido. Inicialmente en esta etapa solo se probará el algoritmo EKF, como forma de comprobar que las ecuaciones realizan la estimación de las salidas de flujo, nivel y temperatura de la planta piloto. En este caso, la planta se opera en modo manual, comenzando por el lazo de flujo, cuya señal de salida es medida por el sensor con Tag FT-01, obteniendo la siguiente aproximación, error de estimación y error de predicción:

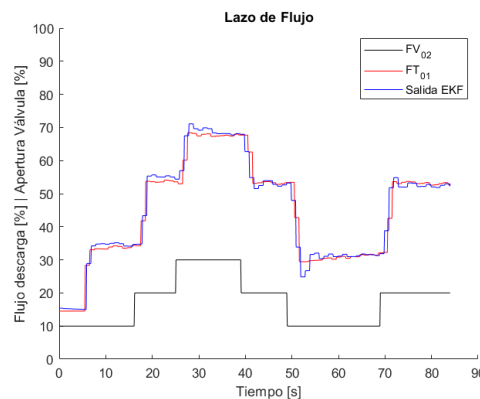


Figura 4.3.1. Comparación entre Sensor de Flujo FT-01 y Salida de Flujo Estimada por EKF

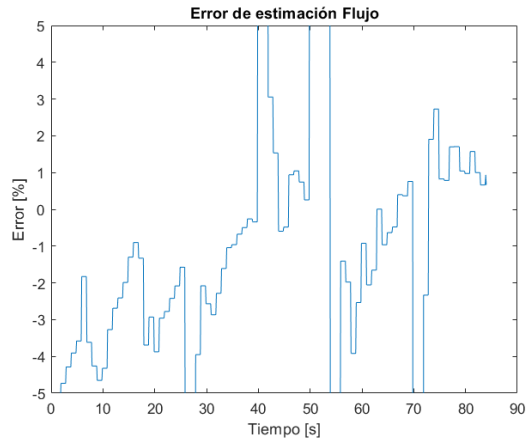


Figura 4.3.2. Error de Estimación de Flujo por EKF

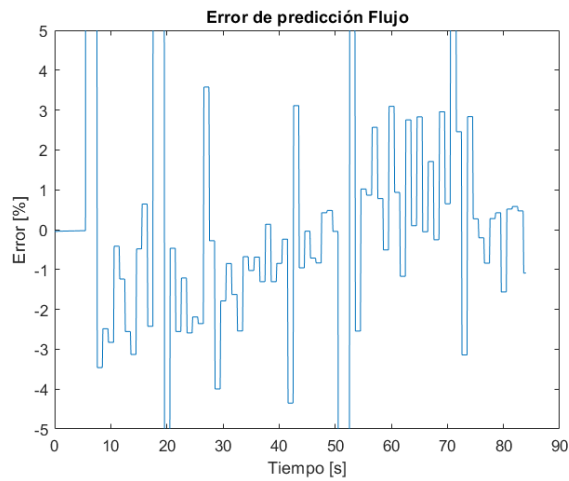


Figura 4.3.3. Error de Predicción de Flujo en Etapa de Identificación

Además, se obtienen los parámetros de flujo mediante un archivo .CSV a Matlab para graficar los parámetros del lazo de flujo y así tener una mejor visualización, los que se muestran a continuación:

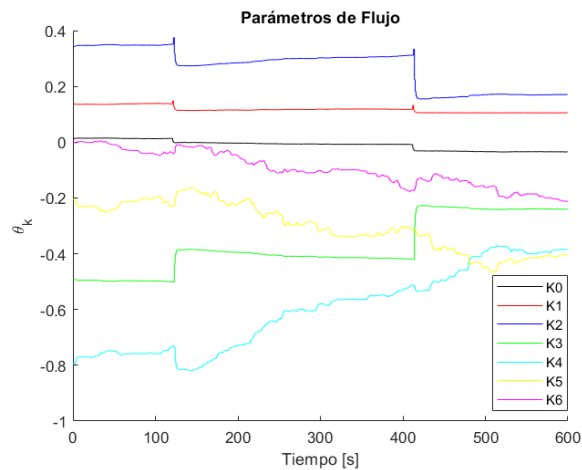


Figura 4.3.4. Parámetros del Lazo de Flujo

De esta forma, es posible observar la evolución de los parámetros durante los cambios realizados en la planta. Se tiene que el error de predicción se encuentra dentro de un rango de 0% y 3%, junto con el error de estimación de sobre el flujo de descarga se encuentra entre 0% y 2%, lo cual es bastante bajo y se encuentra dentro de un rango razonable, observándose que se realiza una disminución en el ruido de la señal durante el tratamiento en el Filtro de Kalman Extendido. Durante la evolución del sistema, se observa que la señal estimada se superpone a la señal del sensor, solo existiendo leves peaks en el momento de cambio de la apertura de la válvula.

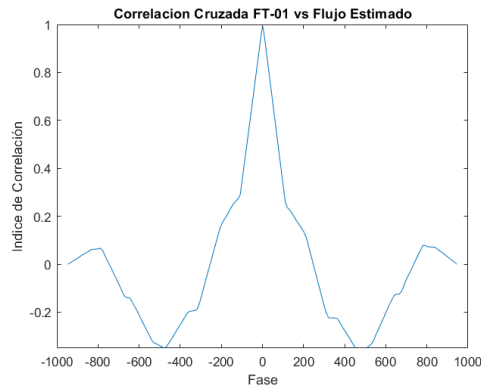


Figura 4.3.5. Correlación cruzada entre FT-01 y Flujo estimado

De la figura 4.3.5, se puede apreciar la correlación cruzada entre el flujo real y el flujo estimado notando que dicho gráfico se encuentra centrado en fase cero y la correlación máxima es de 0,9994, lo que nos da a entender que la señal estimada se ajusta a la señal medida por el sensor.

Realizando este proceso de forma similar con el nivel, solo que, para no rebalsar el estanque, se realiza el cambio de la válvula a través de un controlador PID previamente programado, solo para fines de realizar la prueba del Filtro de Kalman Extendido para el nivel. Así, se tiene que la salida estimada, el error de predicción y error de corrección son:

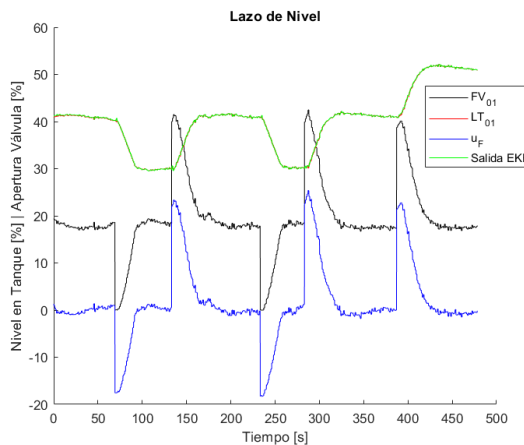


Figura 4.3.6. Comparación entre Sensor de Nivel LT-01 y Salida de Nivel Estimada por EKF

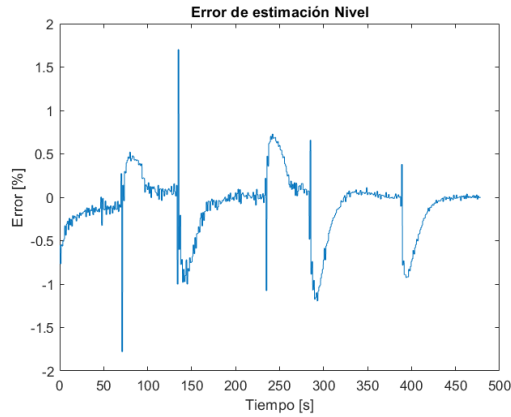


Figura 4.3.7. Error de Estimación de Nivel por EKF

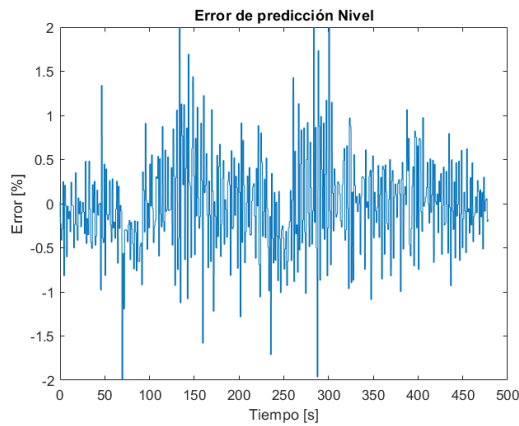


Figura 4.3.8. Error de Predicción de Nivel en Etapa de Identificación

De forma similar, se obtienen los parámetros para el Lazo de Nivel durante el cambio en el sistema en Matlab llevando los datos a través de un archivo en formato .CSV:

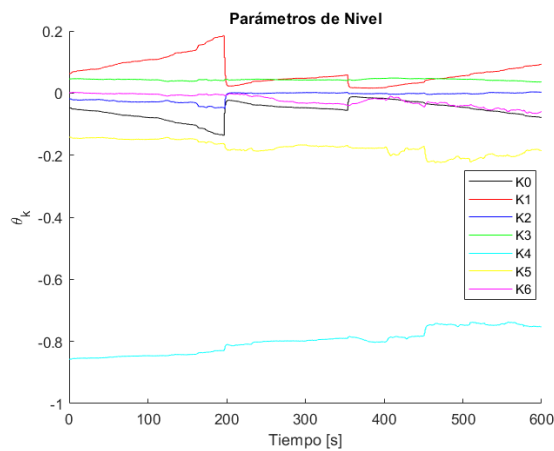


Figura 4.3.9. Parámetros del Lazo de Nivel

De la figura 4.3.6. podemos deducir que la estimación se realiza de forma correcta, pues la señal estimada se superpone a la señal real dada por el sensor. Esto es apoyado en base que el error

de predicción se encuentra en el rango de 0% a 2%, y luego reduciéndose el error de estimación a un rango de 0% a 0.5%, disminuyendo considerablemente el efecto del ruido. Se aprecia que existe un leve peak durante el instante en que cambia la apertura de la válvula, el cual no supera el 1%.

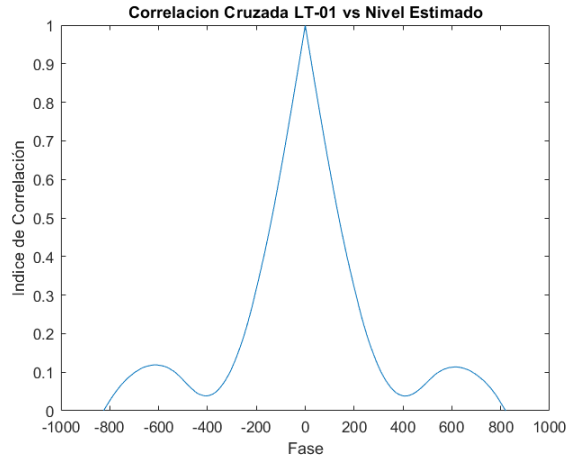


Figura 4.3.10. Correlación cruzada entre LT-01 y Nivel estimado

Con la correlación entre la señal del sensor LT-01 y el nivel estimado, se tiene que, al igual que en el flujo, el gráfico se encuentra centrado en fase cero, y su correlación máxima es de 0,99996, lo que nos demuestra que la estimación se realiza correctamente y se ajusta a la señal medida por el sensor.

Realizando el mismo procedimiento para temperatura, comprobando que la estimación de dicha señal dada por el sensor TT-01, se tiene que la señal estimada, el error de estimación y el error de corrección son:

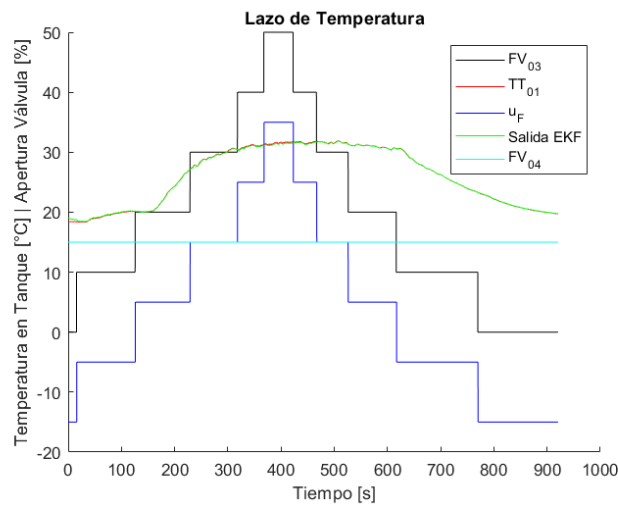


Figura 4.3.11. Comparación entre Sensor de Temperatura TT-01 y Temperatura estimada por EKF

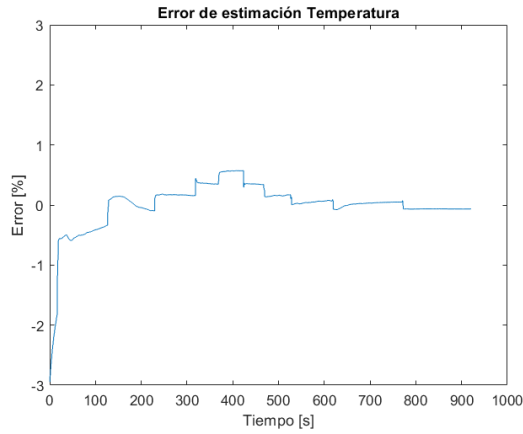


Figura 4.3.12. Error de estimación de Temperatura por EKF

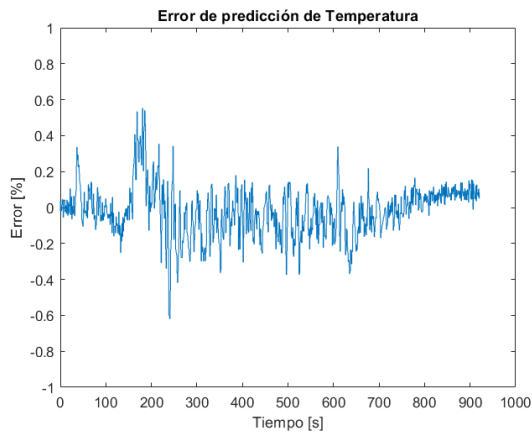


Figura 4.3.13. Error de predicción de Temperatura en Etapa de identificación

Obteniendo los parámetros a partir de los datos entregados por el algoritmo, se llevan a Matlab en forma de un archivo con formato .CSV al igual que en los dos casos anteriores:

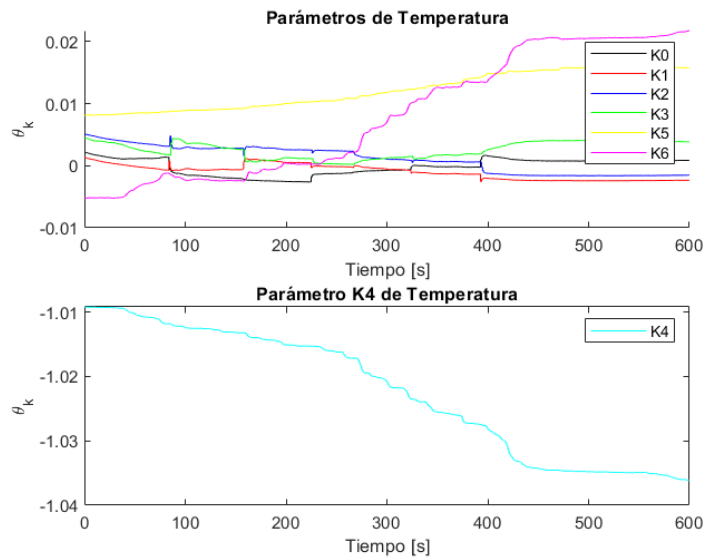


Figura 4.3.14. Parámetros del Lazo de Temperatura

De la figura 4.3.11, se observa que el proceso de estimación se realiza sin inconvenientes, pues ambas señales se superponen, tanto la señal estimada como la señal real. Además, el error de predicción se encuentra entre 0% y 0.5%, y luego el error de estimación se reduce a un valor cercano a 0% y con muy poca variación, solo cambiando levemente cuando la válvula se abre, ajustándose rápidamente a la señal real entregada por el sensor.

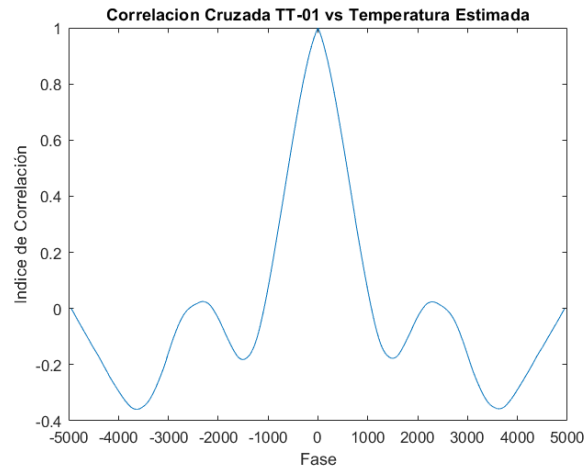


Figura 4.3.15. Correlación cruzada entre TT-01 y Temperatura estimada

Revisando la correlación entre las señales para comprobar que la estimación se realizó correctamente, ésta se encuentra centrada en fase cero, con una correlación máxima de 0,99995, señalando que los datos entregados por el algoritmo desarrollado y aplicado al lazo de temperatura se ajustan correctamente a los datos medidos por el sensor de temperatura.

Capítulo 5. Controladores

5.1. Introducción a Lógica Difusa

Durante los últimos años, el control difuso ha tenido una gran cantidad de aplicaciones en la industria. La investigación de inicial de Mamdani ha permitido desarrollar una estrategia de control basada en la experiencia de control de un operador humano con la planta de interés, lo que beneficia si el sistema a controlar es complejo o no se tiene completo conocimiento de su dinámica, [14, 21].

La idea de este tipo de control es llevar el pensamiento y forma de pensar de la persona que normalmente controlaría el proceso, y llevar este tipo de razonamiento y términos lingüísticos usados a un sistema lógico, [21]. Esto se traduce en que este tipo de control se basará en términos que normalmente pueden ser ambiguos, por ejemplo, usando términos como frío, tibio y caliente, y luego traducir esto a una forma de control para que el sistema siga una referencia en base a la decisión tomada por el controlador.

Muchas veces ocurre que, durante el control, como ya se ha mencionado en otros capítulos, las señales de medición no vienen del todo limpias y tienen interferencias, que sumado a la compleja dinámica que el sistema pueda poseer, hace difícil realizar rutinas de control. Aunque para el control difuso controlar un sistema del que se conoce su dinámica según la rutina de control del operador no es difícil, el ruido presente en la medición es traspasado a los actuadores durante el cálculo de la señal de control, lo que puede afectar enormemente a los accionamientos según la cantidad de ruido que haya en los datos obtenidos, por lo que una combinación con el Filtro de Kalman Extendido y controladores difusos será desarrollada, concentrándose este capítulo en la definición del control.

5.2. Fuzzificación y Funciones de Inferencia

La forma de trabajo de la lógica difusa permite el trabajo con información que no es del todo precisa, siendo capaz de capturar cualitativamente los atributos del sistema de control, basándose principalmente en un fenómeno observable, [21]. De esta forma, el control se realizará a través de la aplicación de términos cualitativos en tiempo real.

La siguiente imagen retrata las diferentes secciones que componen la totalidad de un controlador difuso:

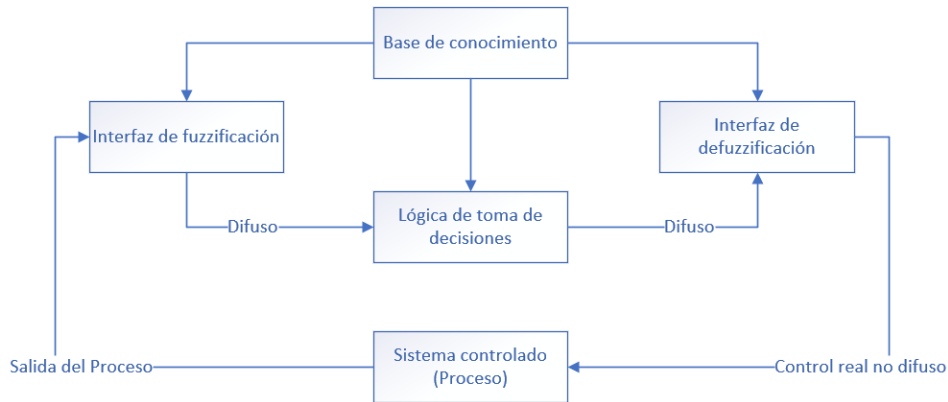


Figura 5.2.1. Composición de un controlador difuso, [23]

Siendo la primera parte la Interfaz de Fuzzificación de los datos, etapa en la cual se introducen los datos al controlador y se reescriben en función a términos que miden las variables de una forma subjetiva. Esto juega un rol importante cuando los datos de medición entregados por los sensores no son precisos.

Inicialmente, los datos fuzzificados son transformados a sets difusos, caracterizados principalmente por funciones de membresía que tomarán valores en el intervalo [0, 1], con variables dentro de un cierto rango, que define el universo del discurso. En nuestro caso, el universo del discurso tomará valores entre [-1, 1], por lo que el error y la derivada del error, las cuales son usualmente las entradas de un controlador difuso, [21, 24]. Con esto, se tendrá que el error y la derivada del error serán normalizados de la siguiente forma:

$$Es = \frac{e_k}{Se} \quad (5.2.1)$$

$$Eds = \frac{de_k}{Sde} \quad (5.2.2)$$

Donde Es y Eds serán el error y la derivada del error normalizados, tomando valores entre [-1, 1] en el universo del discurso. A través de estos términos se trabajarán con las funciones de membresía, las cuales representan el grado de pertenencia de un elemento o subconjunto definido por una etiqueta, [23].

Existe una gran variedad de formas para las funciones de membresía, siendo las más comunes las del tipo trapezoidal, triangular, singleton, S, etc. Todas estas funciones tienen como objetivo definir una característica del control, principalmente apoyada por los términos lingüísticos a usar. En este caso, ya que se requerirá indicar al controlador situaciones en que se necesitará abrir la válvula para dar paso al flujo de agua, mantener el porcentaje de apertura en caso de que se llegue a

la referencia, y cerrar la válvula cuando se supere la referencia, los términos a usar serán:

N: Cerrar la válvula

Z: Mantener apertura de la válvula

P: Abrir la válvula

Usando funciones de membresía triangulares tanto para el Error como para la Derivada del error, apoyándose gráficamente con la herramienta “Fuzzy” de Matlab, las funciones serán:

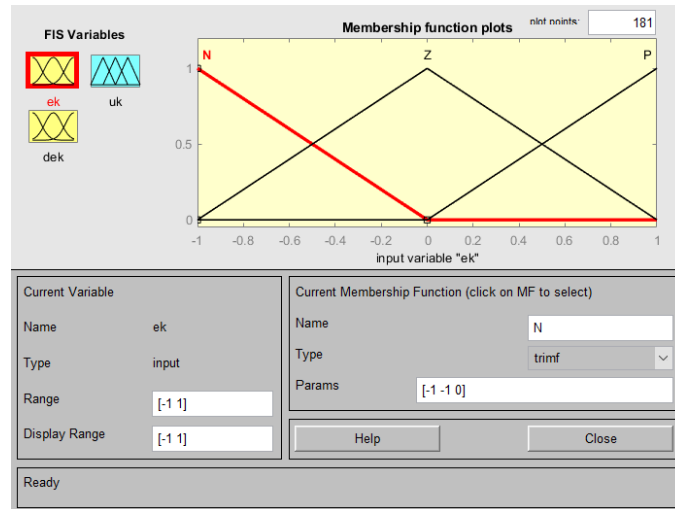


Figura 5.2.2. Funciones de membresía para error

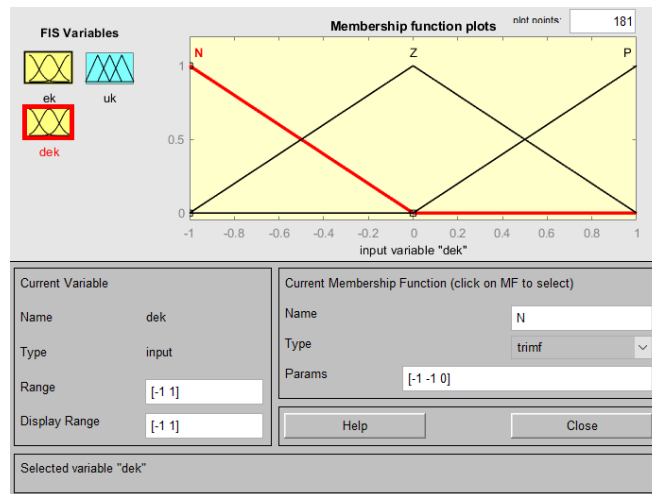


Figura 5.2.3. Funciones de membresía para derivada del error

De esta forma, según sea el valor que tome el valor escalado del error o la derivada del error, tomarán distintos valores según sea la función de inferencia respectiva. A su vez, esto dará paso a la base de reglas difusas y la toma de decisiones del controlador para llevar al sistema hacia la referencia que se introduzca al controlador.

5.3. Reglas Difusas y Matriz de Inferencia

Los controladores difusos, una vez que se definen las funciones de membresía para caracterizar la forma en que se va a tratar la información, es necesario desarrollar la base de toma de decisiones del controlador, es decir, bajo que razonamiento el controlador interpretará datos difusos de entrada asociados al error y a la derivada del error, para asociar otro conjunto difuso de salida como respuesta, llamado consecuente. Esta forma de trabajo se lleva a cabo mediante reglas difusas principalmente con operaciones lógicas AND y OR, desarrolladas de la forma IF-THEN, [21].

Las reglas difusas serán las encargadas de relacionar los antecedentes y los consecuentes, existiendo tantas reglas difusas como antecedentes y consecuentes se definan en el controlador. La forma clásica de estas reglas son reglas de Mamdani, las que tienen la siguiente estructura:

$$\text{If } (X1 \text{ is } A) \& (X2 \text{ is } B) \text{ Then } (X3 \text{ is } C) \quad (5.3.1)$$

Donde X1, X2 y X3 las variables difusas de entrada y salida, y A, B y C son las distintas funciones de inferencia de entrada y salida definidas previamente. En este caso, ya que tenemos dos entradas, siendo estos el error y la derivada del error, tendremos un total de nueve reglas difusas que se encargarán de calcular el cambio en la acción de control, [24].

Para llevar a cabo estas reglas, existen métodos para realizar el procesamiento para el controlador, siendo estas las operaciones difusas entre los sets difusos definidos por las funciones de membresía, siendo los principales la intersección y la unión, respectivamente a continuación, [21]:

$$\mu_{\{A \cap B\}} = \min \{ \mu_A, \mu_B \} \quad (5.3.2)$$

$$\mu_{\{A \cup B\}} = \max \{ \mu_A, \mu_B \} \quad (5.3.3)$$

Donde la ecuación (5.3.2) principalmente será utilizada para definir las nueve reglas iniciales y el peso que tendrán en la toma de decisiones, obteniendo de esta forma términos intermedios que representarán esta última idea, y así la ecuación (5.3.3) entregar el consecuente final. Unificando todo esto en una matriz, se tendrá que las reglas serán las siguientes:

Acción Válvula	de_N	de_Z	de_P
e_N	N	N	Z
e_Z	N	Z	P
e_P	Z	P	P

Tabla 5.3.1. Matriz de Inferencia

En la tabla 5.3.1 se encuentran las reglas difusas que se programarán en el controlador. Por ejemplo, si la señal entregada por el sensor de flujo FT-01 es mayor a la referencia y la derivada del error es negativa, entonces el controlador le indicará al sistema que debe comenzar a cerrar la válvula para que el sistema comience a seguir a la referencia, ya que, si abrimos las válvulas de la planta, el caudal de flujo aumentará. Este razonamiento se extiende para las otras ocho reglas, de forma que, según esta matriz, el controlador irá tomando diferentes acciones para lograr que el sistema alcance de forma correcta el sistema.

5.4. Defuzzificación

La etapa de defuzzificación es el proceso inverso a la fuzzificación, encargándose de convertir los términos lingüísticos resultantes de la base de toma de decisiones a valores concretos que el sistema puede leer y seguir, [22]. Al igual que para las entradas, la salida también se define en base a funciones de membresía, las que se muestran a continuación, definiendo como el cambio en la acción de control, en este caso, dirigido a las válvulas del sistema, debiese proceder:

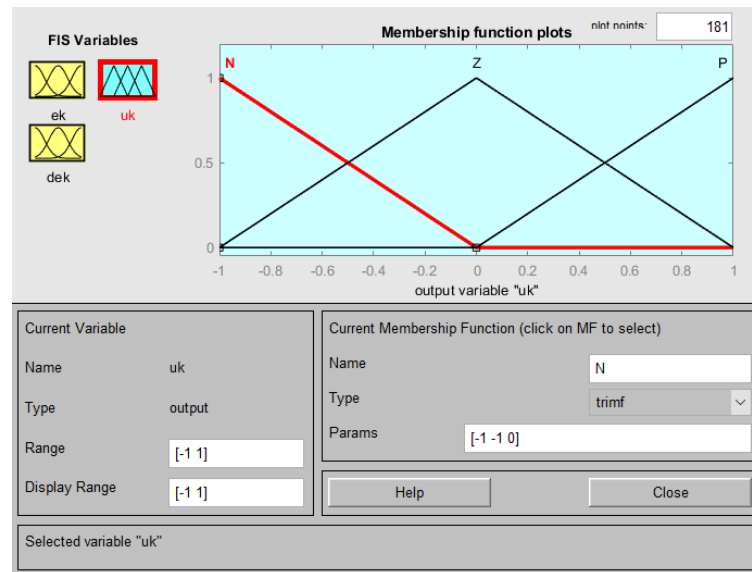


Figura 5.4.1. Funciones de membresía para cambio de la acción de control

Existen distintos métodos para seguir este procedimiento, de los cuales, el más utilizado es el método del centroide, el que se define según la siguiente ecuación, [22]:

$$du_k = \frac{Mdu_N \cdot w_N + Mdu_Z \cdot w_Z + Mdu_P \cdot w_P}{Mdu_N + Mdu_Z + Mdu_P} \quad (5.4.1)$$

$$du_k = Sdu \cdot du_k \quad (5.4.2)$$

Con estas ecuaciones se realizará el proceso de defuzzificación, de forma que podremos reconstruir la señal de control para el accionamiento del sistema. La ecuación (5.4.1) se refiere al

método del centroide, donde los puntos w se refieren al punto en donde el área de la función de membresía correspondiente se divide en dos áreas iguales, las que en nuestro caso corresponden a $w_N = -0.707$, $w_Z = 0$ y $w_P = 0.707$. La ecuación (5.4.2) es utilizada como factor para ajustar el término de la salida, en caso de que requiera que la señal du_k cambie más rápida o más lenta, lo que estará relacionado con la sintonización del sistema. Cabe destacar que los términos de escalamiento Se y Sde mencionados anteriormente también se utilizarán con este fin, de forma que podremos cambiar estos valores para lograr una salida que se ajuste a nuestros requerimientos. Una vez realizado este procedimiento, logrando que los lazos de flujo, nivel y temperatura, designados por los sensores FT-01, LT-01 y TT-01, que entregan cada señal de salida respectivamente, son los siguientes:

	Se	Sde	Sdu
Flujo	90	70	30
Nivel	300	8.3	30.3
Temperatura	290	3.2	2.8

Tabla 5.4.1. Términos utilizados para Sintonización de los controladores

Capítulo 6. Resultados Experimentales

6.1. Generalidades

El algoritmo desarrollado implementa una sección de Mínimos Cuadrados Recursivos para obtener los parámetros del modelo utilizado para estimar la respuesta de los diferentes lazos de flujo, nivel y temperatura que componen la planta. Una segunda sección corresponde a un Filtro de Kalman Extendido para obtener una estimación de la salida y los parámetros antes obtenidos mediante RLS, para disminuir el ruido de predicción de la primera sección del algoritmo y obtener mejor calidad en la señal estimada. En el Anexo D.1 se encuentra el código implementado en el PLC y desarrollado en Rslogix 5000. El código es similar para los lazos de flujo, nivel y temperatura, solo cambiando las entradas y salidas del algoritmo. El procedimiento a seguir es similar al encontrado en [13].

La planta piloto se compone principalmente de un estanque de 50 [cm] de alto y 35 [cm] de diámetro, en el cual se compone de una entrada controlada por una válvula de control con tag FV-01 para el lazo de nivel y una salida en la zona inferior del estanque controlada por una válvula de control con tag FV-02 para el lazo de flujo, junto con una bomba B-01 para este lazo e impulsar la descarga del estanque. Se cuenta con una chaqueta de agua para el lazo de temperatura, con una válvula de control FV-03 para el agua caliente y una válvula de control FV-04 para el agua fría. Para impulsar este lazo, se cuenta además con una bomba B-02 para el agua fría de recirculación.

En las siguientes figuras se observan los equipos que componen los lazos de flujo, nivel y temperatura, que en su conjunto forman la planta en que se realizará la implementación:

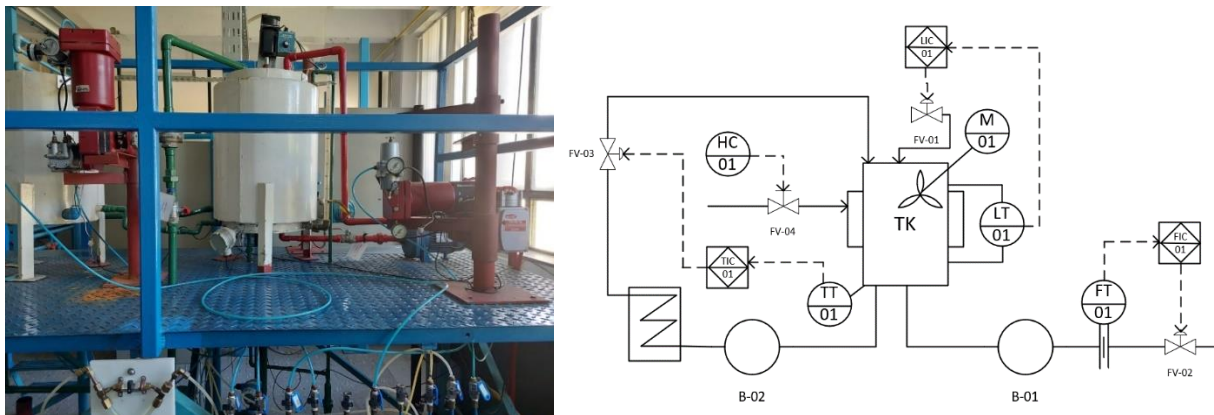


Figura 6.1.1. Vista general de planta piloto y P&ID



Figura 6.1.2. Bomba B-01 y Válvula FV-02



Figura 6.1.3. Calefones para el lazo de temperatura

Los sensores involucrados en cada uno de los lazos de flujo, nivel y temperatura que fueron utilizados para tomar las mediciones e implementarlas en el EKF desarrollado están etiquetados como FT-01 ubicado en el lazo de descarga a través de una placa orificio, LT-01 como un sensor de nivel en el estanque y TT-01 a través de una Pt100 en el fondo del estanque para tomar la temperatura del agua, la cual posee un motor agitador M-01 para homogeneizar la temperatura del contenido del estanque.

El PLC en el que se cargó el algoritmo es de la marca Allen Bradley, que cuenta con un módulo Ethernet para conexiones, y módulos de entrada y salida para los distintos sensores. El software utilizado para conectarse al PLC y cargar el algoritmo es Rslonix 5000. Los tiempos de muestreo utilizados para cada lazo fueron de 1.0 [s], que es el tiempo mínimo con el cual no hay inconvenientes para la correcta ejecución del programa desarrollado. En la siguiente imagen se observa la configuración dentro del gabinete de control, donde se aprecia la diferente disposición de

los módulos mencionados:



Figura 6.1.4. PLC Allen Bradley 1756 de la planta piloto

El algoritmo desarrollado, como se explicó anteriormente, consta de dos secciones de identificación y Filtro de Kalman Extendido, respectivamente. La primera sección se encarga de obtener los parámetros del lazo en el que se aplica mientras que la segunda sección aplica el Filtro de Kalman Extendido, filtrando y mejorando la calidad de la señal de salida. Para realizar el control, se aplica un controlador difuso para realizar el control sobre la variable de interés, ya sea flujo, nivel o temperatura, el cual realiza el cálculo del error a partir de la estimación realizada a partir del Filtro de Kalman Extendido.

De esta forma, un diagrama en bloques que representa el controlador, el algoritmo desarrollado y la planta sería el siguiente:

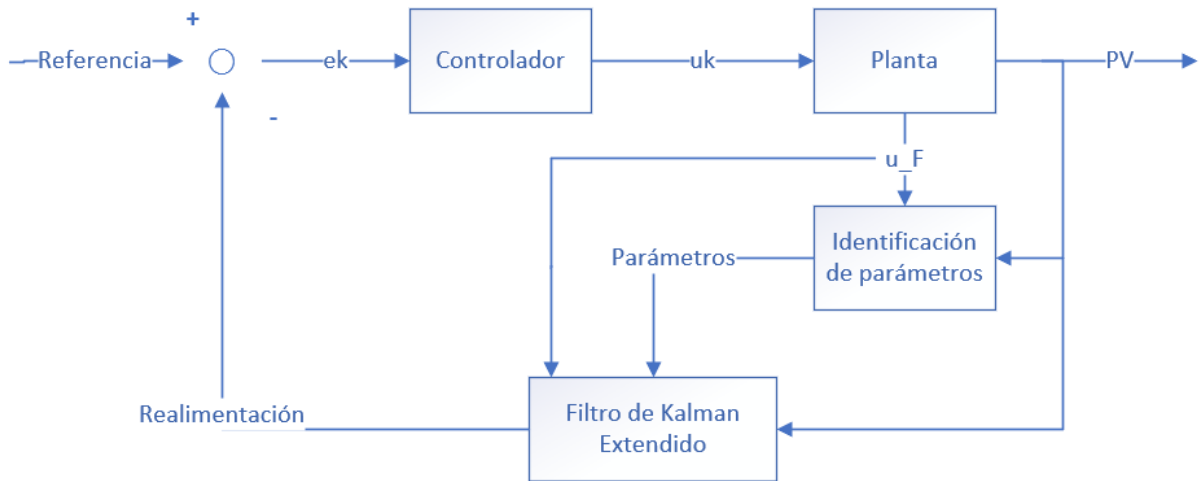


Figura 6.1.5. Diagrama en bloques del conjunto control, planta y algoritmo

En donde, la señal e_k es el error entre la referencia y la medición estimada, u_k es la señal de control que entrega el controlador, u_F es la entrada del algoritmo correspondiente a variables manipulables, y el sensor corresponde al instrumento que toma la medición. Cabe destacar que las entradas u_F cambian para los distintos lazos. Para el lazo de temperatura, la entrada al Filtro será la válvula de descarga FV-02, para el nivel será la diferencia entre la válvula de carga FV-01 y la válvula de descarga FV-02, y finalmente para la temperatura será la diferencia entre la válvula de agua caliente FV-03 y la válvula de agua fría FV-04.

De esta forma, el controlador realizará el cálculo del error entre la referencia deseada para la planta y la señal estimada por el Filtro de Kalman Extendido, [20]. El controlador realizará los ajustes en base a la válvula de control correspondiente al lazo para lograr la llegada a la referencia, indicando el porcentaje de apertura de la válvula para que la planta realice los cambios en la variable que se desea controlar. A su vez, el algoritmo desarrollado tomará como entradas el sensor mediante el cual se tomen las mediciones de la variable de flujo, nivel o temperatura, y las válvulas que correspondan para realizar las estimaciones correspondientes. Los resultados se obtienen en el PLC y se llevan a un archivo .CSV para evaluar el comportamiento del controlador junto con el Filtro de Kalman Extendido, así como comprobar el comportamiento del control en Matlab, donde se podrá observar mejor los resultados obtenidos.

6.2. Lazo de Flujo

Para obtener los resultados del lazo de flujo, se tiene la estructura mostrada en la figura 6.1.5, en donde el Filtro de Kalman Extendido recibirá como entradas la apertura de la válvula de descarga FV-02 y como salida a estimar la señal del sensor FT-01, llevando a la realimentación la señal resultante de la estimación realizada por el filtro hacia el controlador, resultando en una señal con menos ruido y que estima el retardo en la señal en base al modelo en el cual se basa el Filtro de Kalman Extendido. De esta forma, llevando las señales a Matlab, se obtiene lo siguiente:

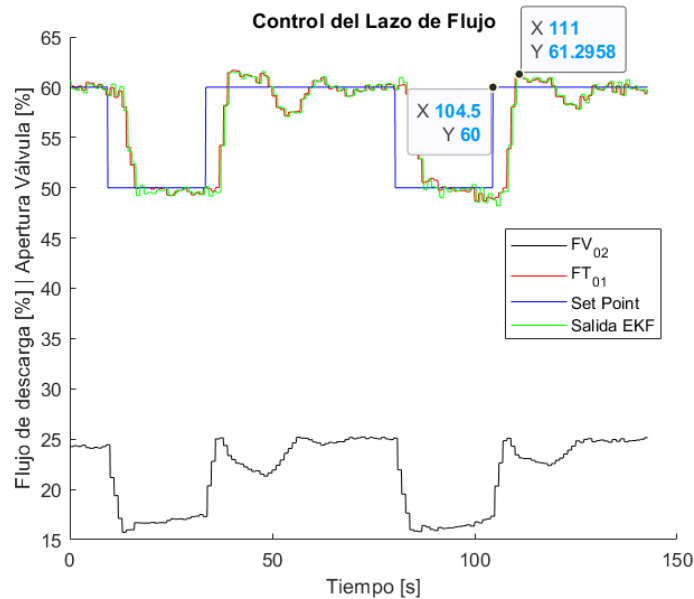


Figura 6.2.1. Control de Flujo de descarga con controlador y Filtro de Kalman Extendido

Al observar la figura, podemos ver en verde la señal estimada del Filtro de Kalman Extendido, el rojo la señal obtenida por el sensor, en negro la apertura de la válvula dada por el controlador, y en azul la referencia que se le da al sistema. Podemos notar que existe un sobrepaso de 2.15% junto con un tiempo de asentamiento de 6.5 [s], notando que en todo momento existe un seguimiento de la dinámica de la señal estimada por el EKF a la respuesta de la planta, por lo que el control mediante la señal estimada se cumple sin inconvenientes al disminuir el error, como se muestra en los siguientes gráficos.

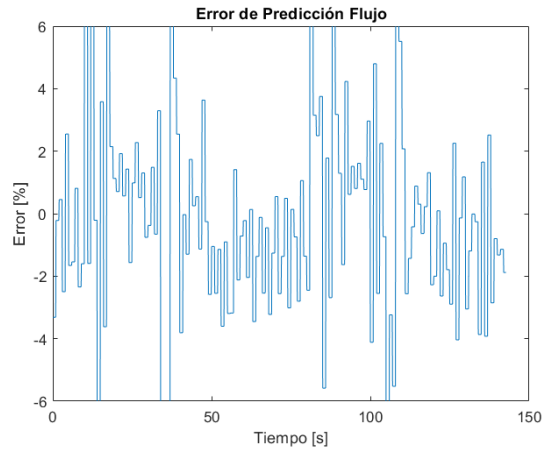


Figura 6.2.2. Error de predicción durante el control de flujo

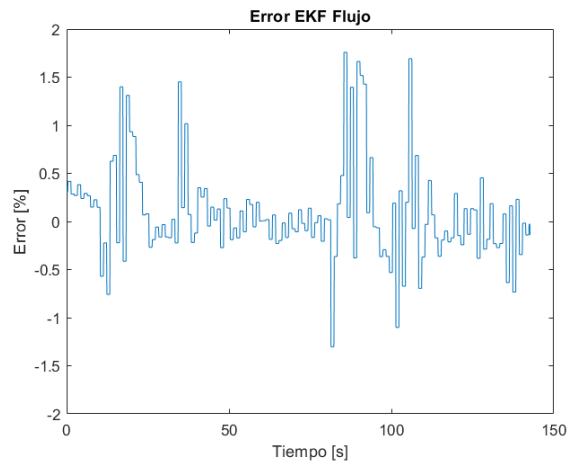


Figura 6.2.3. Error de estimación durante el control de flujo

Donde a partir de las figuras 6.2.2 y 6.2.3 es posible notar una clara disminución del ruido de la señal de estimación, solo presentando variaciones entre 0% y 1.5% durante el cambio en forma escalón de la referencia, ajustándose luego con error cercano a cero en estado estacionario.

De esta forma, es posible entregarle al actuador del lazo de control, en este caso, la válvula de descarga FV-02 una señal que no tiene inconvenientes como ruido y retardo, ya que estos problemas son solucionados durante la estimación en tiempo real del Filtro de Kalman, por lo que el controlador no traspasará el ruido a la válvula.

6.3. Lazo de Nivel

Para el lazo de nivel, se sigue la misma estructura mostrada en el diagrama en bloques de la figura 6.1.5, siguiendo un proceso similar a como se hizo con el lazo de flujo. En este caso, el Filtro de Kalman Extendido que estimará la señal de nivel dada por el sensor LT-01 recibirá como entrada la diferencia entre la carga y descarga del estanque, dada por la expresión $u_F = FV_{01} - FV_{02}$,

además de la señal misma del sensor LT-01 de nivel. De esta forma el Filtro de Kalman Extendido procesará estos datos y le entregará al controlador de nivel una señal estimada del nivel en el estanque con ruido disminuido y basada en el modelo con el cual se realizan las estimaciones en el Filtro de Kalman Extendido. Así, se tiene lo siguiente:

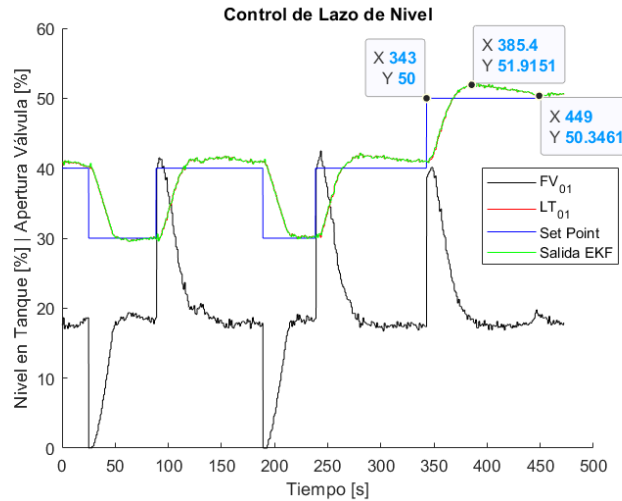


Figura 6.3.1. Control del Nivel del Estanque con controlador y Filtro de Kalman Extendido

Observando la figura 6.3.1, podemos notar en rojo la señal del sensor LT-01, en azul la referencia, en negro la apertura de la válvula FV-01 y verde la señal estimada del Filtro de Kalman Extendido. Al evaluar el gráfico, se obtiene un sobrepaso de 3.84% junto con un tiempo de asentamiento de 106 [s], logrando un seguimiento de la referencia. Durante el transcurso de la acción de control, se realiza un seguimiento de la señal estimada del EKF respecto a la señal del sensor, representando la dinámica del sistema, a la vez que se disminuye el ruido, tal como se muestra a continuación:

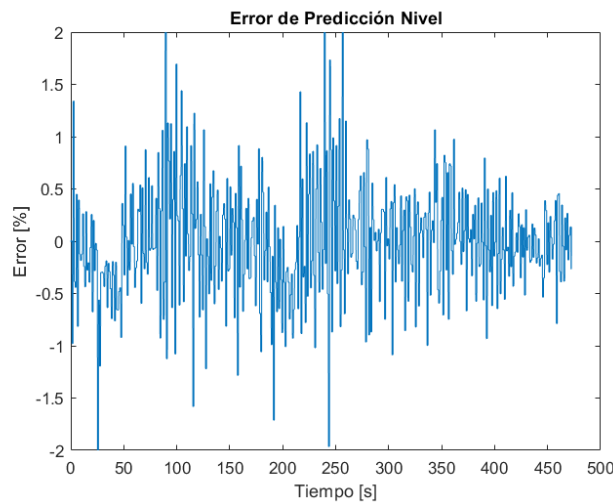


Figura 6.3.2. Error de predicción durante el control de nivel

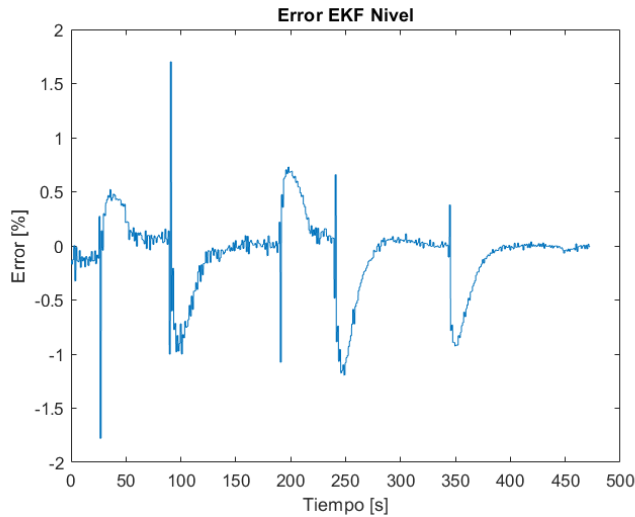


Figura 6.3.3. Error de estimación durante el control de nivel

Con las figuras que representan el error de predicción y de estimación, se observa una clara disminución del error en estado estacionario, el cual es cercano a cero. Solo se aprecia error entre 0% y 1% durante la variación de la señal en la fase transitoria, el cual disminuye con el tiempo.

6.4. Lazo de Temperatura

De igual forma a los casos anteriores, para la temperatura se introducen los datos de la planta hacia el Filtro de Kalman Extendido. Las señales entregadas son la diferencia entre el agua caliente y el agua fría $u_F = FV_{03} - FV_{04}$, y la señal de medición entregada por el sensor de temperatura TT-01. Con esto, se tiene la siguiente figura con el controlador realizando el ajuste con el EKF.

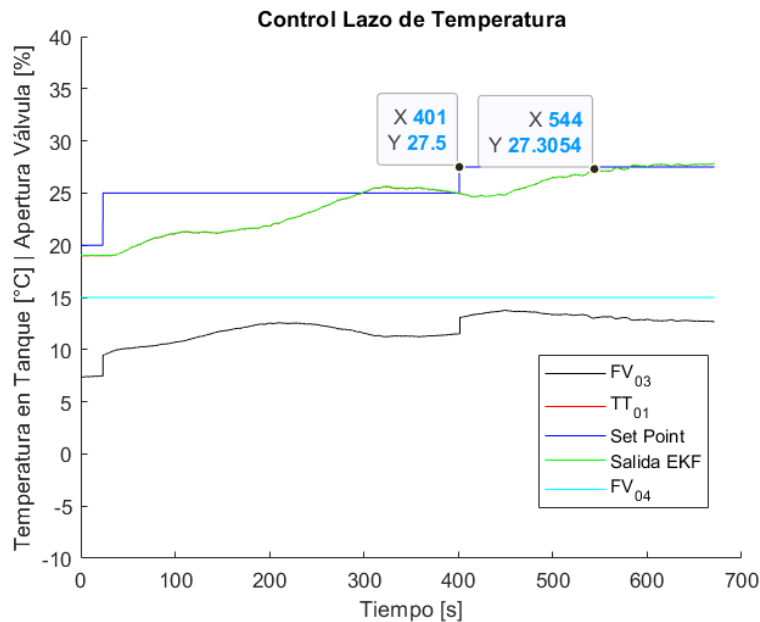


Figura 6.4.1. Control de temperatura con controlador y Filtro de Kalman Extendido

Donde la señal morada es la salida del Filtro de Kalman Extendido de temperatura, la señal verde es la referencia, la señal roja es la válvula de agua caliente, la señal roja es la válvula de agua fría y la señal negra es la señal tomada por el sensor de temperatura TT-01. Al evaluar los resultados del control junto con el EKF, se puede observar ausencia de sobrepaso, es decir, 0% de sobrepaso, y un tiempo de asentamiento de 143 [s]. Podemos notar también que se realiza un correcto seguimiento de la respuesta de la planta por parte de la salida estimada por el filtro, ya que ambas señales se superponen. El error de predicción y error de estimación se pueden observar a continuación:

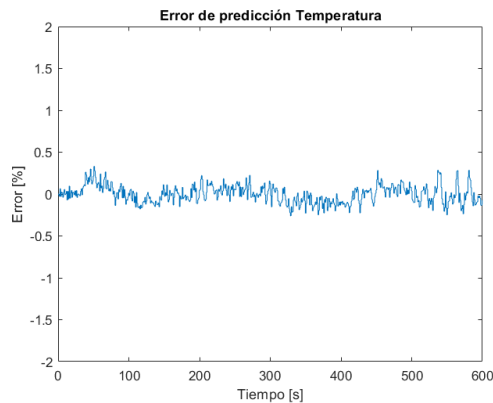


Figura 6.4.2. Error de predicción durante el control de temperatura

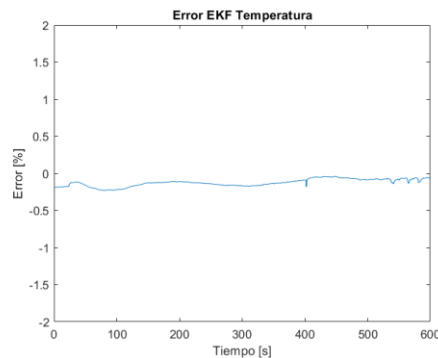


Figura 6.4.3. Error de estimación durante el control de temperatura

De las figuras 6.4.2 y 6.4.3 se aprecia la reducción del ruido. El error de predicción, es decir, antes de la aplicación del Filtro de Kalman Extendido, tiene un error que se encuentra entre 0% y 0.5%, para luego de la aplicación del Filtro de Kalman Extendido verse reducido a un valor muy cercano a 0%. De esto, es posible deducir que durante las acciones de control que efectúa el controlador, el Filtro de Kalman Extendido es capaz de seguir realizando estimaciones de la señal de temperatura, entregando de esta forma una señal mucho más limpia al controlador, de forma que no hay ruido que se traspasa al actuador, en este caso, la válvula de agua caliente.

Capítulo 7. Conclusiones

7.1. Sumario

En el desarrollo de este proyecto se realizó la búsqueda de una F. de T. que modelara el comportamiento de los sistemas que conforman la planta piloto, siendo estos lazos de flujo, nivel y temperatura, considerados como modelos SISO en el caso del flujo, y MISO en el caso del nivel y la temperatura. Se escogió una estructura de primer orden con tiempo muerto, aproximando este último a través del método que mostró mejores resultados según la bibliografía, correspondiendo al método NS4-2 a través de retrasos de primer orden, para posteriormente discretizar la estructura resultante para uso posterior.

Para el desarrollo de la estrategia a implementar en la planta piloto, debido a las características no lineales observadas en la dinámica de la planta, fue necesario realizar una identificación de parámetros del modelo antes mencionado a través de Mínimos Cuadrados Recursivos con Factor de Olvido. De esta forma, obteniendo los parámetros del sistema en todo momento a través de datos de la planta tomados en tiempo real desde los sensores. Sin embargo, estos parámetros necesitarán un filtrado debido al ruido presente en los sensores, el cual se consideró como ruido gaussiano blanco no correlacionado, correspondiendo a la segunda etapa del algoritmo.

Para la segunda etapa del algoritmo a desarrollar, se requirió una estimación de la señal de salida del sistema debido al tiempo muerto junto con una disminución del ruido, por lo que se realizó un estudio y búsqueda bibliográfica para las distintas alternativas del Filtro de Kalman existentes actualmente, escogiéndose finalmente el Filtro de Kalman Extendido para realizar la estimación de los estados del sistema y los parámetros de forma simultánea. La estimación requirió entregar al sistema los parámetros obtenidos en la etapa de identificación, la señal del sensor de medición y entrada manipulable del sistema, donde esta última estará compuesta de las válvulas de la planta. Cabe destacar que estas últimas variables variarán dependiendo del lazo que se desee estimar.

En la fase de implementación, se desarrolló el Filtro de Kalman Extendido en una fase inicial de diseño en Matlab para verificar la efectividad del algoritmo previamente a su implementación en el PLC. Luego de desarrollado el algoritmo en Matlab, se tradujo a formato de texto estructurado en el software Rslogix 5000 para implementar el código estimador en el PLC de la planta piloto, ubicada en el laboratorio de control de procesos en el segundo piso del DIE. Se evaluó el Filtro de Kalman Extendido desarrollado con los lazos de la planta en modo manual, a excepción del nivel para no

rebalsar el estanque, verificando que la estimación del algoritmo se realizará sin problemas, evaluando la disminución del ruido en las diferentes etapas a evaluar, la evolución en el tiempo de los parámetros y mediante correlación de datos, que la salida estimada por el algoritmo y la salida real del sistema tuvieran la misma dinámica.

Luego se desarrolló un controlador difuso para los tres lazos ya mencionados que componen la planta, a modo de utilizar en el PLC en tiempo real la señal estimada durante el control y seguimiento de una referencia deseada. El cálculo del error se realiza a partir de la referencia que se desea siga la planta y la salida estimada por el Filtro de Kalman, a modo de obtener un error que se vea menos afectado por el ruido de las mediciones y otros problemas. Finalmente se evaluaron los resultados del controlador junto con el Filtro de Kalman Extendido para los lazos de flujo, nivel y temperatura.

7.2. Conclusiones

Se desarrolló una estrategia de estimación de sistemas basada en el Filtro de Kalman Extendido, el cual se aplicó en tiempo real junto con un controlador difuso para entregar a los accionamientos señales de control más limpias y libres de ruido. El código desarrollado fue aplicado en formato de texto estructurado en el PLC con una tasa de muestreo de 1.0 [s], siendo capaz de realizar estimaciones de las señales tomadas de los sensores FT-01, LT-01 y TT-01 de forma simultánea al cálculo de las acciones de control para indicarle a las variables de interés que sigan la referencia indicada.

Para el lazo de flujo se consideró como entrada al algoritmo del Filtro de Kalman Extendido la válvula de descarga FV-02 y el sensor de flujo FT-01, mientras que al controlador de flujo se le entrega la señal estimada de flujo por el filtro, pues esta señal tiene ruido disminuido y el modelo desarrollado se ha ajustado al tiempo muerto presente en el flujo. La señal resultante presenta un error de predicción menor a 3%, para luego de la estimación encontrarse en el rango menor a 1.5% durante la fase transitoria, y tomando valores cercanos a cero durante la fase estacionaria. La acción de control obtuvo un sobrepaso de 2.15% y un tiempo de asentamiento de 6.5 [s], donde la señal estimada sigue a la señal real tomada por el sensor de flujo, ajustándose de esta forma a la dinámica de la planta. Además, se realizó una correlación entre la señal estimada de flujo y la señal real de flujo tomada por el sensor, dando un valor de 0,9994 en su punto máximo, el cual se encuentra centrado en cero, lo que resulta en que la estimación se ajusta a la respuesta real del flujo.

En el caso del nivel, se siguió una metodología similar al flujo. Como entrada al algoritmo del EKF, se consideró como entrada la diferencia entre las válvulas de carga y descarga del estanque,

además del sensor de nivel LT-01. Además, al controlador de nivel se le entrega la señal de nivel estimada por el filtro, la cual ya tiene ruido disminuido y a través del modelo, se ha ajustado al tiempo muerto del nivel. Esto resulta en un error de predicción menor a 1% antes de la aplicación del filtro, y posterior a su aplicación, la señal estimada presenta un error de estimación cercano a cero en la fase estacionaria, junto con un error de estimación menor a 1% en fase transitoria, el cual se corrige rápidamente. La acción de control en el nivel presenta un sobrepaso de 3.84% y un tiempo de asentamiento de 106 [s]. Durante la acción tomada por el controlador, se observa que la estimación responde de forma similar a la respuesta real de la planta tanto en la fase transitoria como en la fase estacionaria. Este ajuste se comprueba a través de la correlación entre la señal estimada y la señal real, dando un valor de 0,99996 en su punto máximo, el cual está centrado en cero.

Para el lazo de temperatura, análogamente al nivel, las entradas al algoritmo fueron la diferencia entre la válvula de agua caliente y la válvula de agua fría, además de la señal de temperatura tomada por el sensor de temperatura TT-01. Al controlador de temperatura, al igual que en los casos anteriores, se le entrega la señal de temperatura estimada, la cual ya presenta disminución en el ruido, y a través del modelo desarrollado, se ajusta al tiempo muerto de la planta. Con esto, se obtiene un error de predicción menor a 0.5%, y posterior a la aplicación del algoritmo, se observa un error cercano a cero. En cuanto al control, no se aprecia sobrepaso y el tiempo de asentamiento son 143 [s]. Se obtuvo además la correlación entre la señal estimada y la señal real de temperatura, lo que resultó en un valor de 0,99995 en su punto máximo, el cual está centrado en cero, lo que nos corrobora que la señal estimada de temperatura por el Filtro de Kalman Extendido se ajusta a la respuesta de la planta.

Con esto, podemos concluir que el ruido se logró disminuir haciendo uso del Filtro de Kalman Extendido al aplicar el algoritmo desarrollado a los lazos de flujo, nivel y temperatura, siendo reducido a valores cercanos a cero en la fase estacionaria y valores menores a 1% durante la fase transitoria. Además, el modelo realizado logro ajustarse a los diferentes tiempos muertos que presentaron los lazos respectivos, de esta forma ajustándose a la respuesta real de la planta, sin importar el tiempo muerto que presentara el lazo. De todos los gráficos mostrados en el capítulo 6, es posible concluir que el Filtro de Kalman Extendido es capaz de funcionar en tiempo real en un PLC, añadiéndose a las operaciones diarias de la planta, a fin de disminuir el ruido presente en las mediciones, además de ajustarse a los tiempos muertos que el sistema presente, y de esta forma, proteger los accionamientos de señales que presenten variaciones muy rápidas y bastante seguidas

producto del ruido y durante el control de la planta, extendiendo la vida útil del equipo en el que sea implementado y facilitando las labores de los controladores de la planta.

7.3. Trabajos futuros

Si bien el desarrollo actual permitió generar un algoritmo capaz de ajustarse a la respuesta que da la planta para los lazos de flujo, nivel y temperatura, hay aspectos interesantes que pueden seguir desarrollándose desde lo detallado en este informe.

En el trabajo presentado, el tiempo muerto es estimado de forma implícita a través de los parámetros del modelo, permitiendo de esta forma ajustarse a la salida deseada y realizar un control más limpio. Otro aspecto que queda por desarrollar es obtener el tiempo muerto de forma explícita, haciendo uso de éste para aplicar técnicas de auto sintonía de la planta, por ejemplo, la técnica de auto sintonía de Aström y Häggland, u otras técnicas de auto sintonía en lazo abierto o cerrado. Esto permitiría explorar controladores aplicables a sistemas de primer o segundo orden de naturaleza inestable, siendo algunos ejemplos las operaciones de biorreactores o reactores exotérmicos.

Además, en base a la salida del Filtro de Kalman Extendido, existen dos formas de realizar un control sobre la planta. La forma implementada en este informe fue el “modo sensor”, realizado a través de la reconstrucción de la salida a través de la ecuación de medida de la planta. El segundo modo de control aplicable sería a través de la utilización directa de los parámetros, los cuales se obtienen en cada iteración debido al proceso de identificación en tiempo real. De esta forma, el control de la planta sería a través de la utilización de los parámetros ya estimados por el Filtro de Kalman Extendido, por ejemplo, haciendo uso de técnicas de control basadas en modelos como controladores MPC, GPC, u otros controladores predictivos que sean capaces de hacer uso de esta forma de control, haciendo finalmente una comparación entre ambos posibles modos de control “sensorial” o “paramétrico”, identificando cual sería el recomendado según los resultados del desempeño de ambas técnicas.

Referencias

- [1] G. Welch and G. Bishop “An Introduction to the Kalman Filter,” University of North Carolina at Chapel Hill, TR-95-041, Jul 24. 2006.
- [2] D. Morrel “Extended Kalman Filter Lecture Notes,” *EEE 581-Spring*, 1997,
- [3] R. Kalman “A New Approach to Linear Filtering and Prediction Problems,” in *ASME Journal of Basic Engineering*, 82, pp 35-45, Marzo 1, 1960.
- [4] W. Liu “Appendix A. System Identification: State and Parameter Estimation Techniques,” *Introduction to Hybrid Vehicle System Modeling and Control, First Edition*. John Wiley & Sons. 2013.
- [5] P. Madhukar and L. Prasad “State Estimation using Extended Kalman Filter and Unscented Kalman Filter,” in *International Conference on Emerging Trends in Communications, Control and Computing (ICONC3)*, Mody University of Science and Technology, Lakshmangarh, Febrero 21-22, 2020.
- [6] B. Tamani “Obtención de Modelos de Procesos Mediante Métodos de Identificación Recursiva”. Artículo de Revista. Facultad de Ingeniería Electrónica y Eléctrica, Universidad Nacional Mayor de San Marcos, Lima, Perú. No. 19, agosto 18, 2007.
- [7] E. Wan and R. van der Merwe “The Unscented Kalman Filter for Nonlinear Estimation,” in *Adaptative Systems for Signal Processing, Communications, and Control Symposium*, Lake Louise, AB, Canada. Octubre 2020, pp 153-158.
- [8] G. Terejanu “Extended Kalman Filter Tutorial,” Department of Computer Science and Engineering, University of Buffalo, Buffalo, NY, Estados Unidos, 2009.
- [9] F. Sánchez “Un Nuevo Método de Identificación de Procesos Continuos No Oscilatorios de Alto Orden,” Tesis de Master, Instituto Tecnológico y de Estudios Superiores de Monterrey, Monterrey, México, 2001.
- [10] H. Wang “Improved Extended Kalman Particle Filter Based on Markov chain Monte Carlo for Nonlinear State Estimation,” in *International Conference on Uncertainty Reasoning and Knowledge Engineering*, Jalarta, Indonesia. Agosto 2012, pp 281-285.
- [11] T. Phuong, M. Belov, N. Lanh and N. Thanh “State Estimation of Nonlinear

- Electromechanical System using Extended Kalman Filter,” 3rd International Conference on Control in Technical Systems, San Petersburgo, Rusia, octubre 2019.
- [12] J. Seung, A. Atiya, A. Parlos and K. Chong “Parameter Estimation for Coupled Tank using Estimate Filtering”, in International Journal of Control and Automation. Vol.6 No.5, 2013, pp 91 – 102.
- [13] W. Wang, J. Yao and S. Qiu “Implementation of Kalman Filter in PLC” in International Symposium on Big Data and Artificial Intelligence. Hong Kong, Hong Kong. 2018, pp 242-245.
- [14] G.M. Tamiselvan and P. Aathy “Online tuning of fuzzy logic controller using Kalman algorithm for conical tank system,” Department of EIE, Bannari Amman Institute of Technology, Sathyamangalam, India. Mayo 24, 2017.
- [15] D. Beckmann, M. Dagen and T. Ortmaier “Symplectic Discretization Methods for Parameter Estimation of a Nonlinear Mechanical System using an Extended Kalman Filter,” in 13th International Conference on informatics in Control, Automation and Robotics ICINCO 2016. Vol.1, 2016, pp 327-334.
- [16] J. Castañeda, M. Nieto and V. Ortiz “Análisis y Aplicación del Filtro de Kalman a una señal con ruido aleatorio,” Scientia Et Technical, Vol. 18, No. 1, abril 2013.
- [17] F. Auger, M. Hilairet, J. Guerrero, E. Monmasson, T. Orłowska-Kowalska and S. Katsura “Industrial Applications of the Kalman Filter: A Review,” IEEE Transactions on Industrial Electronics, Vol. 60, No. 12, Enero 2013, pp 5458-5471.
- [18] L. Sánchez, J. Ordoñez and S. Infante “Filtro de Kalman Extendido y Filtro de Partículas Kalman Extendido para problemas de estimación No lineal,” Revista Ingeniería UC, Vol. 20, No. 1. abril 2013, pp 7-16.
- [19] A. Gautam and S. Majumdar “State Estimation of Single-Phase Rectifier Based Load Circuit using Unscented Kalman Filter,” in 2nd IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES-2018), Delhi, India, octubre 2018.
- [20] R. Choomuang and N. Afzulpurkar “Hybrid Kalman Filter/Fuzzy Logic based Position Control of Autonomous Mobile Robot” in International Journal of Advanced Robotic

- Systems, Vol. 2, No. 3, 2005, pp 197-208.
- [21] C. Chuen Lee “Fuzzy Logic in Control Systems: Fuzzy Logic Controller – Part I” in Transaction on Systems, Man, and Cybernetics, Vol. 20, No. 2, 1990, pp 404-418.
- [22] C. Chuen Lee “Fuzzy Logic in Control Systems: Fuzzy Logic Controller – Part II” in Transaction on Systems, Man, and Cybernetics, Vol. 20, No. 2, 1990, pp 419-435.
- [23] Z. Gao, T. Trautzsch and J. Dawson “A Stable Self-Tuning Fuzzy Logic Control System for Industrial Temperature Regulation” in Transactions on Industry Applications, Vol. 38, No. 2, 2002, pp 414-424.
- [24] A. Aissaoui and A. Tahour “*Application of Fuzzy Logic in Control of Electrical Machines, Fuzzy Logic – Control, Concepts, Theories and Applications*”, Prof. Elmer Dadios. Croatia: InTech, 2012.
- [25] “Apuntes de Sistemas Lineales Dinámicos”, 543 214, Departamento de Ingeniería Eléctrica, Universidad de Concepción, Marzo 2018.

Anexo A. Obtención del Modelo Discreto

Del capítulo 2, se tiene que se está modelando el Sistema con una función de primer orden con retardo aproximado como se muestran en la ecuación (2.5). Todos los cálculos mostrados en este Anexo fueron realizados en el software “Mathcad” para facilitar la obtención de las expresiones matemáticas para realizar el modelo a usar y aplicar en el código del PLC. El primer paso a seguir es aplicar transformada de Laplace inversa a la F. de T. del modelo, lo cual nos arroja el siguiente resultado, considerando que la F. de T. tiene como parámetros las constantes k, τ y θ .

$$h(t) = \frac{4 \cdot \tau^2 \cdot k + \theta^2 \cdot k - 4 \cdot \tau \cdot \theta \cdot k - 4 \cdot \tau^2 \cdot k \cdot e^{-\frac{t}{\tau}} - \theta^2 \cdot k \cdot e^{-\frac{2 \cdot t}{\theta}} + 4 \cdot \tau \cdot \theta \cdot k \cdot e^{-\frac{2 \cdot t}{\theta}} + 4 \cdot \tau \cdot k \cdot t \cdot e^{-\frac{2 \cdot t}{\theta}} - 2 \cdot \theta \cdot k \cdot t \cdot e^{-\frac{2 \cdot t}{\theta}}}{(\theta - 2 \cdot \tau)^2}$$

Luego de realizar el reemplazo $t = n * T$, se aplica la transformada z, como se muestra en la siguiente fórmula hacia la F.T. $h(s)$:

$$h(z) = (1 - z^{-1}) \cdot Z \left\{ L^{-1} \left\{ \frac{h(s)}{s} \right\} \right\}$$

Primero aplicaremos la transformada Z antes de realizar la multiplicación por la expresión que representa al retentor de orden cero. Lo que da el siguiente resultado:

$$\frac{4 \cdot \tau^2 \cdot k \cdot z}{(\theta - 2 \cdot \tau)^2 \cdot (z - 1)} + \frac{\theta^2 \cdot k \cdot z}{(\theta - 2 \cdot \tau)^2 \cdot (z - 1)} - \frac{4 \cdot \tau^2 \cdot k \cdot z}{(\theta - 2 \cdot \tau)^2 \cdot \left(z - e^{-\frac{T}{\tau}} \right)} - \frac{\theta^2 \cdot k \cdot z}{(\theta - 2 \cdot \tau)^2 \cdot \left(z - e^{-\frac{2 \cdot T}{\theta}} \right)} - \frac{4 \cdot \tau \cdot \theta \cdot k \cdot z}{(\theta - 2 \cdot \tau)^2 \cdot (z - 1)} + \frac{4 \cdot \tau \cdot \theta \cdot k \cdot z}{(\theta - 2 \cdot \tau)^2 \cdot \left(z - e^{-\frac{2 \cdot T}{\theta}} \right)} + \frac{4 \cdot T \cdot \tau \cdot k \cdot z \cdot e^{-\frac{2 \cdot T}{\theta}}}{(\theta - 2 \cdot \tau)^2 \cdot \left(z \cdot e^{-\frac{2 \cdot T}{\theta}} - 1 \right)} - \frac{2 \cdot T \cdot \theta \cdot k \cdot z \cdot e^{-\frac{2 \cdot T}{\theta}}}{(\theta - 2 \cdot \tau)^2 \cdot \left(z \cdot e^{-\frac{2 \cdot T}{\theta}} - 1 \right)}$$

Para ordenar la expresión y facilitar el álgebra posterior, la parte constante de las funciones que esté en función de los parámetros de la F.T. inicialmente usada, serán numerados de la A hasta la H como continua:

$$A = \frac{(4 \cdot \tau^2 \cdot k)}{(\theta - 2 \cdot \tau)^2} \quad B = \frac{(\theta^2 \cdot k)}{(\theta - 2 \cdot \tau)^2} \quad C = \frac{(4 \cdot \tau^2 \cdot k)}{(\theta - 2 \cdot \tau)^2} \quad D = \frac{(\theta^2 \cdot k)}{(\theta - 2 \cdot \tau)^2}$$

$$E = \frac{(4 \cdot \tau \cdot \theta \cdot k)}{(\theta - 2 \cdot \tau)^2} \quad F = \frac{(4 \cdot \tau \cdot \theta \cdot k)}{(\theta - 2 \cdot \tau)^2} \quad G = \frac{\left(4 \cdot T \cdot \tau \cdot k \cdot e^{-\frac{2 \cdot T}{\theta}} \right)}{(\theta - 2 \cdot \tau)^2} \quad H = \frac{\left(2 \cdot T \cdot \theta \cdot k \cdot e^{-\frac{2 \cdot T}{\theta}} \right)}{(\theta - 2 \cdot \tau)^2}$$

De esta forma, se tendrá la siguiente expresión de $h(z)$:

$$H(z) = \left(\frac{z-1}{z} \right) \left[\frac{A \cdot z}{z-1} + \frac{B \cdot z}{z-1} - \frac{C \cdot z}{\left(z - e^{-\frac{T}{\tau}} \right)} - \frac{D \cdot z}{\left(z - e^{-\frac{2 \cdot T}{\theta}} \right)} - \frac{E \cdot z}{(z-1)} + \frac{F \cdot z}{\left(z - e^{-\frac{2 \cdot T}{\theta}} \right)} + \frac{G \cdot z}{\left(z - e^{-\frac{2 \cdot T}{\theta}} \right)^2} - \frac{H \cdot z}{\left(z - e^{-\frac{2 \cdot T}{\theta}} \right)^2} \right]$$

Para obtener las expresiones de los parámetros que necesitamos, es necesario realizar un ajuste a la expresión obtenida. Inicialmente vamos a mantener un denominador común, el cual está dado por la siguiente expresión en función de los polos del sistema:

$$\left(z - e^{-\frac{T}{\tau}} \right) \left(z - e^{-\frac{2 \cdot T}{\theta}} \right)^2$$

Con esto, es posible dividir $h(z)$ en tres expresiones más que tienen un denominador común:

$$\begin{aligned} & \frac{(A + B - E) \cdot \left(z \cdot e^{-\frac{4 \cdot T}{\theta}} - e^{-\frac{T}{\tau}} \cdot e^{-\frac{4 \cdot T}{\theta}} + z^3 - z^2 \cdot e^{-\frac{T}{\tau}} - 2 \cdot z^2 \cdot e^{-\frac{2 \cdot T}{\theta}} + 2 \cdot z \cdot e^{-\frac{T}{\tau}} \cdot e^{-\frac{2 \cdot T}{\theta}} \right)}{\left(z - e^{-\frac{T}{\tau}} \right) \left(z - e^{-\frac{2 \cdot T}{\theta}} \right)^2} \\ & - C \cdot \frac{\left(2 \cdot z \cdot e^{-\frac{2 \cdot T}{\theta}} - e^{-\frac{4 \cdot T}{\theta}} + z \cdot e^{-\frac{4 \cdot T}{\theta}} - z^2 + z^3 - 2 \cdot z^2 \cdot e^{-\frac{2 \cdot T}{\theta}} \right)}{\left[\left(z - e^{-\frac{T}{\tau}} \right) \left(z - e^{-\frac{2 \cdot T}{\theta}} \right)^2 \right]} \\ & \frac{(-D + F) \cdot \left(z \cdot e^{-\frac{T}{\tau}} + z \cdot e^{-\frac{2 \cdot T}{\theta}} - e^{-\frac{T}{\tau}} \cdot e^{-\frac{2 \cdot T}{\theta}} - z^2 + z^3 - z^2 \cdot e^{-\frac{T}{\tau}} - z^2 \cdot e^{-\frac{2 \cdot T}{\theta}} + z \cdot e^{-\frac{T}{\tau}} \cdot e^{-\frac{2 \cdot T}{\theta}} \right)}{\left[\left(z - e^{-\frac{T}{\tau}} \right) \left(z - e^{-\frac{2 \cdot T}{\theta}} \right)^2 \right]} \\ & \frac{(G - H) \cdot \left(e^{-\frac{T}{\tau}} - z - z \cdot e^{-\frac{T}{\tau}} + z^2 \right)}{\left[\left(z - e^{-\frac{T}{\tau}} \right) \left(z - e^{-\frac{2 \cdot T}{\theta}} \right)^2 \right]} \end{aligned}$$

Ahora que ya conocemos estas expresiones, es posible comenzar a darle forma al numerador de $h(z)$. Para el numerador se tendrán términos independientes, de primer, segundo y tercer orden. Realizando el despeje algebraico y reemplazando las expresiones de A hasta H, se tienen los siguientes parámetros de KO hasta K6, en función de la siguiente F.T. llamada $h(z)$:

$$h(z) = \frac{K1 \cdot z^{-1} + K2 \cdot z^{-2} + K3 \cdot z^{-3}}{1 + K4 \cdot z^{-1} + K5 \cdot z^{-2} + K6 \cdot z^{-3}}$$

Las expresiones de cada uno de los parámetros K0 hasta K6 en función de los parámetros iniciales k, τ y θ se muestran a continuación. Tomar consideración de que el parámetro K0 se anula algebraicamente, por lo que es cero.

De esta forma, es posible obtener los valores de K1 hasta K6 y saber cómo obtendrán su valor. Con estas expresiones, es posible obtener la expresión de la salida de forma discreta y obtener el modelo en ecuaciones de estado que necesitaremos posteriormente a este desarrollo. Las expresiones algebraicas se encuentran en la siguiente tabla:

$K1 = \frac{\left(4\tau^2k + \theta^2k - 4\tau\theta k + 4T\tau k \cdot e^{\frac{-2T}{\theta}} - 2T\theta k \cdot e^{\frac{-2T}{\theta}}\right) + e^{\frac{-T}{\tau}} \cdot (-4\tau^2k) + e^{\frac{-2T}{\theta}} \cdot (4\tau\theta k - \theta^2k)}{(\theta - 2\tau)^2}$
$K2 = \frac{\left(2T\theta k \cdot e^{\frac{-2T}{\theta}} - 4T\tau k \cdot e^{\frac{-2T}{\theta}}\right) + e^{\frac{-4T}{\theta}} \cdot (\theta^2k - 4\tau\theta k) + e^{\frac{-2T}{\theta}} \cdot (4\tau\theta k - 8k\tau^2 - \theta^2k) + e^{\frac{-T}{\tau}} \cdot (4\tau\theta k - \theta^2k)}{2T\theta k \cdot e^{\frac{-2T}{\theta}} + \left(e^{\frac{-T}{\tau}} \cdot e^{\frac{-2T}{\theta}}\right) \cdot (8k\tau^2 - 4\tau\theta k + \theta^2k)}$
$K3 = \frac{4\tau^2k \cdot e^{\frac{-4T}{\theta}} + \left(4T\tau k \cdot e^{\frac{-2T}{\theta}} - 2T\theta k \cdot e^{\frac{-2T}{\theta}}\right) \cdot e^{\frac{-T}{\tau}} + \left(e^{\frac{-T}{\tau}} \cdot e^{\frac{-4T}{\theta}}\right) \cdot (4\tau\theta k - \theta^2k - 4\tau^2k) + \left(e^{\frac{-T}{\tau}} \cdot e^{\frac{-2T}{\theta}}\right) \cdot (8k\tau^2 - 4\tau\theta k + \theta^2k)}{(\theta - 2\tau)^2}$
$K4 = \left(-e^{\frac{-T}{\tau}} - 2 \cdot e^{\frac{-2T}{\theta}}\right)$
$K5 = \left(e^{\frac{-4T}{\theta}} + 2 \cdot \left(e^{\frac{-T}{\tau}} \cdot e^{\frac{-2T}{\theta}}\right)\right)$
$K6 = -\left(e^{\frac{-T}{\tau}} \cdot e^{\frac{-4T}{\theta}}\right)$

Anexo B. Filtro de Kalman Suavizado y Filtro Partícula de Kalman Extendido

B.1. Filtro de Kalman Suavizado

El Filtro de Kalman Suavizado, o Unscented Kalman Filter en inglés fue propuesto por Julier y Uhlman en el año 1997 como una alternativa al Filtro de Kalman Extendido. En el EKF, la distribución de los estados es aproximada mediante ruido gaussiano variable, extendiéndose a través de una linealización de primer orden al modelo no lineal, lo que a la larga puede introducir errores en el valor real y la covarianza producto del ruido, [7]. El Filtro de Kalman Suavizado, o UKF para abreviar, resuelve este problema con un enfoque diferente, para lo cual usa una aproximación muestral. Aunque la distribución de los estados es aproximada por el ruido blanco variable (GRV en inglés), ahora es representado a partir de un set mínimo de puntos de muestreo escogidos, capturando el verdadero valor y la covarianza del GRV, extendiéndose al sistema no lineal sin necesidad de aplicar matrices jacobianas o hessianas, [5]. En contraste, el EKF realiza este trabajo a partir de una aproximación de primer orden, la cual tiene un coste computacional del mismo orden que el UKF.

El UKF surge de la necesidad de resolver problemas de estimación no lineal, incluyendo identificación de sistemas no lineales, entrenamiento de redes neuronales y problemas de estimación dual, [7].

Considerando una estimación en espacios de estado básica similar a las anteriores, dada una medición ruidosa y_k , una estimación recursiva del estado x_k puede ser expresada como:

$$\hat{x}_k = (\text{Predicción de } x_k) + K_k[y_k - (\text{Predicción de } y_k)] \quad (\text{B.1.1})$$

Esta ecuación recursiva provee el mínimo error cuadrado óptimo (MMSE) estimado para x_k asumiendo que el estado *a priori* \hat{x}_{k-1} y la observación y_k son ruido gaussiano variable (GRV). Notar que todas las ecuaciones involucradas en el proceso recursivo, ya sea la matriz de covarianza y la ganancia de Kalman dependerán de valores del estado *a priori* obtenidos a partir de la función no lineal que define al sistema, realizando los cálculos de forma similar al Filtro de Kalman aplicado a sistemas lineales.

De esta forma, el UKF es una generalización del EKF realizada a través de la captura de puntos obtenidos a partir de un muestreo, capturando el valor medio y la covarianza, aproximando al sistema sin importar la no linealidad del modelo, [7].

Todos los cálculos involucrados en esta alternativa del Filtro de Kalman se basan en la

transformada suavizada, método que tiene por objetivo calcular valores estadísticos de señales que sufren de no linealidad. Considerando una señal x de dimensión L a través de una función no lineal, se tendrá que x tiene como promedio \bar{x} y covarianza P_x . Para comenzar el proceso, se toman $2L + 1$ vectores sigma X_i de peso W_i de acuerdo con [7]:

$$X_0 = \bar{x} \quad (\text{B.1.2})$$

$$X_i = \bar{x} + \left(\sqrt{(L + \lambda)P_x} \right)_i, i = 1, \dots, L \quad (\text{B.1.3})$$

$$X_i = \bar{x} - \left(\sqrt{(L + \lambda)P_x} \right)_{i-L}, i = L + 1, \dots, 2L \quad (\text{B.1.4})$$

$$W_0^{(m)} = \frac{\lambda}{L + \lambda} \quad (\text{B.1.5})$$

$$W_0^{(c)} = \frac{\lambda}{L + \lambda} + (1 - \alpha^2 + \beta) \quad (\text{B.1.6})$$

$$W_i^{(m)} = W_i^{(c)} = \frac{\lambda}{2(L + \lambda)}, i = 1, \dots, 2L \quad (\text{B.1.7})$$

Donde $\lambda = \alpha^2(L + \kappa) - L$ es un parámetro escalar, α determina la propagación de los puntos sigma X_i en torno a la media, tomando generalmente valores positivos pequeños. κ es un parámetro escalar secundario que usualmente tiene valor cero, y β se usa para incorporar valores previos de la distribución x . $\left(\sqrt{(L + \lambda)P_x} \right)_i$ es la fila i -ésima de la raíz cuadrada de la matriz. Los vectores sigma se propagan a través del sistema no lineal:

$$Y_i = g(X_i), i = 0, \dots, 2L \quad (\text{B.1.8})$$

Notar que la media y la covarianza para y son aproximadas usando la media muestral ponderada y la covarianza de los puntos sigma:

$$\bar{y} \approx \sum_{i=0}^{2L} W_i^{(m)} Y_i \quad (\text{B.1.9})$$

$$P_y = \sum_{i=0}^{2L} W_i^{(c)} (Y_i - \bar{y})(Y_i - \bar{y})^T \quad (\text{B.1.10})$$

Esta aproximación realizada por la transformada suavizada resulta en aproximaciones que tienen una precisión de tercer orden para entradas gaussianas para todas las no linealidades. En caso de tratarse de entradas no gaussianas, la precisión es de segundo orden, con posibilidad de alcanzar

el tercer orden al escoger parámetros α y β adecuadamente.

A través de esta transformada, se tiene que el UKF realiza la estimación de la salida a través de estos puntos sigma, sin necesidad de obtener matrices jacobianas a partir de un modelo previamente conocido. De esta forma, el filtro aplicará los puntos sigma a las ecuaciones de predicción y corrección del algoritmo, de forma que hay diferencias con respecto a las alternativas anteriores, manteniendo la filosofía detrás de la técnica de estimación.

El primer set de ecuaciones, llamado ecuaciones de predicción, no solo se encargarán de obtener los estados *a priori*, sino también de los puntos sigma encargados de construir estos estados, lo que nos lleva a lo siguiente, [5, 12]:

$$x_{k-1}^j = \hat{x}_{k-1}^+ + \hat{x}^j \quad (\text{B.1.11})$$

$$x_k^j = f(x_{k-1}^j, u_{k-1}, w_{k-1}) \quad (\text{B.1.12})$$

$$\hat{x}_k^- = \frac{1}{2m} \sum_{i=1}^{2m} x_k^i \quad (\text{B.1.13})$$

$$P_k^- = \frac{1}{2m} \sum_{i=1}^{2m} (x_k^i - \hat{x}_k^-)(x_k^i - \hat{x}_k^-)^T + Q_k \quad (\text{B.1.14})$$

Donde x_k^j son los puntos sigma en el tiempo k , \hat{x}_k^- es el estado estimado *a priori* y \hat{x}_k^+ es el estado estimado *a posteriori*. Podemos ver una diferencia en las ecuaciones con respecto a las versiones anteriores del Filtro de Kalman, ya que en este caso el modelo solicitado por UKF usa la ecuación de proceso f para calcular el valor de los puntos sigma para el estado, para posteriormente reconstruir los estados estimados, siguiendo una filosofía similar para calcular la matriz de covarianza *a priori* P_k^- mediante estas cantidades, [5]. Adicionalmente, se tiene la matriz Q_k que es la matriz de covarianza del ruido de proceso.

En el caso de las ecuaciones de corrección, ahora tendremos añadido un estimador para la salida, ya que al usar los puntos sigma, la ecuación de medida se usará para el cálculo de los puntos sigma para la salida. Al igual que con las ecuaciones de predicción, estos puntos sigma para el estado y la salida afectarán a las ecuaciones del set de corrección, por lo que algunas ecuaciones cambiarán para adaptarse al uso de la transformada suavizada usada por el Filtro de Kalman Suavizado, y al igual que en el caso anterior, no se realiza uso de matrices jacobianas o hessianas para relacionar la salida y el estado, pues se hace uso de los puntos sigma para realizar la relación a través de la ecuación no lineal g que define esta relación no lineal para el sistema.

Las ecuaciones en este caso que se encargarán de disminuir el ruido presente en las mediciones a través de la salida ruidosa tomada del sistema, por [5, 12], son:

$$y_k^j = g(x_k^j, u_k, v_k) \quad (\text{B.1.15})$$

$$\hat{y}_k = \frac{1}{2m} \sum_{i=1}^{2m} y_k^j \quad (\text{B.1.16})$$

$$K_k = P_{xy} P_y^{-1} \quad (\text{B.1.17})$$

$$P_y = \frac{1}{2m} \sum_{i=1}^{2m} (y_k^j - \hat{y}_k)(y_k^j - \hat{y}_k)^T + R_k \quad (\text{B.1.18})$$

$$P_{xy} = \frac{1}{2m} \sum_{i=1}^{2m} (x_k^j - \hat{x}_k)(y_k^j - \hat{y}_k)^T \quad (\text{B.1.19})$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k(y_k - \hat{y}_k) \quad (\text{B.1.20})$$

$$P_k^+ = P_k^- - K_k P_y K_k^T \quad (\text{B.1.21})$$

Como se puede apreciar, las matrices Q_k y R_k son las matrices de covarianza del ruido de proceso y medida respectivamente que afectan a la matriz de covarianza *a posteriori* P_k^+ , la cual en este caso se obtiene de forma recursiva a partir de la matriz de covarianza *a priori*. La ecuación que difiere de las técnicas anteriores es la ecuación (B.1.3), pues la ganancia de Kalman dependerá de los puntos sigma de la salida y los estados del sistema. A través de este segundo set de ecuaciones, la salida ruidosa se verá filtrada, resultando en una nueva señal más limpia resultante de la estimación del UKF, y al no usar aproximaciones a partir de jacobianos y matrices hessianas, el error de estimación debería ser menor al error de estimación de EKF, [7].

Así, ambos sets de ecuaciones calculan en cada iteración los puntos sigma correspondientes a cada estado y salida del sistema, de forma de realizar la estimación mediante las funciones f y g de proceso y medida respectivamente de los puntos sigma para estimar la salida del sistema mediante la señal ruidosa en la que se basa el filtro para calcular la innovación y estimar el estado *a posteriori* para obtener una salida limpia.

B.2. Filtro Partícula de Kalman Extendido

Esta alternativa del Filtro de Kalman gira en torno a la búsqueda de mejoras y aumento en la precisión del filtro junto con los últimos avances en las computadoras. La idea básica detrás del EKPF es lograr el filtrado a través del método de simulación de Monte Carlo. La información *a*

priori y la información *a posteriori* son descritas, tomando una muestra de dicha información y combinándose en un proceso iterativo con el fin de que las propiedades de la simulación de Monte Carlo sean iguales a la función de densidad de probabilidad posterior, lo que le permite aproximar el sistema requerido, ya sea lineal o no lineal, como también a sistemas que no necesariamente tienen un comportamiento gaussiano, [10].

Un problema común que tiene el filtro de partícula es el fenómeno de degeneración, el cual se genera cuando luego de unas pocas iteraciones las partículas pierden diversidad, resultando en que las partículas de importancia toman un peso muy cercano a 1. Con esto, es importante escoger una distribución razonable para evitar este problema.

Al basarse en el Filtro de Kalman Extendido, la técnica se basa en la expansión de series de Taylor de primer orden de la distribución de transición y de la distribución de importancia, [18]. Esto se logra a través de un set de muestras de partículas y sus pesos asociados para representar la densidad de probabilidad, calculando los estados estimados, [10]. En el marco recursivo, la aproximación de la distribución *a posteriori* en el tiempo se combina en cada paso con las nuevas observaciones del sistema.

De esta forma, el EKPF puede ser resumido a través de un set de ecuaciones que calculará la estimación de los estados y muestreará las partículas junto con obtener sus pesos correspondientes. Para iniciar el proceso recursivo, es necesario definir las condiciones iniciales, [10, 18]:

$$x_0^i \sim p(x_0), \text{ para } i = 1, \dots, N \quad (\text{B.2.1})$$

$$w_0^i \sim p(y_0 | x_0^i) \rightarrow \tilde{w}_0^i = \frac{w_0^i}{\sum_{j=0}^N w_0^j} \quad (\text{B.2.2})$$

Donde x_0^i son las muestras y \tilde{w}_0^i son los pesos normalizados. Una vez definidas las condiciones iniciales, se tendrán las ecuaciones de predicción y corrección, que actualizan los valores a través de las ecuaciones del EKF:

$$\hat{x}_{k|k-1}^i = f(x_{k-1}^i, 0) \quad (\text{B.2.3})$$

$$P_{k|k-1}^i = F_k P_{k-1}^i F_k^T + Q_{k-1} \quad (\text{B.2.4})$$

$$K_k^i = P_{k|k-1}^i H_k^T (H_k P_{k|k-1}^i H_k^T + R_k)^{-1} \quad (\text{B.2.5})$$

$$\bar{x}_k^i = \hat{x}_{k|k-1}^i + K_k^i (y_k - h(\hat{x}_{k|k-1}^i, 0)) \quad (\text{B.2.6})$$

$$\hat{P}_k^i = (I - K_k^i H_k) P_{k|k-1}^i \quad (\text{B.2.7})$$

Donde las matrices Q y R son las matrices de covarianza del ruido respectivamente, las cuales pueden ser variables o constantes. Además, las matrices F y H se obtienen a partir de los jacobianos de las ecuaciones de proceso y medida respectivamente:

$$F_k = \left. \frac{\partial f(x)}{\partial x} \right|_{x=x_{k-1}} \quad (\text{B.2.8})$$

$$H_k = \left. \frac{\partial h(x)}{\partial x} \right|_{x=\hat{x}_{k|k-1}} \quad (\text{B.2.9})$$

Notar que adicionalmente hay que tomar muestras de las partículas para obtener las estimaciones, las cuales se obtienen de la siguiente forma:

$$x_k^i \sim N(\hat{x}_k^i, \hat{P}_k^i) \quad (\text{B.2.10})$$

$$w_k^i \sim p(y_k | x_k^i) \tilde{w}_{k-1}^i \quad (\text{B.2.11})$$

Para volver a normalizar los pesos de las muestras antes tomadas, se debe aplicar la siguiente ecuación:

$$\tilde{w}_k^i = \frac{w_k^i}{\sum_{j=0}^N w_k^j} \quad (\text{B.2.12})$$

Si el tamaño de la muestra efectivo \hat{N}_{TME} es menor que un cierto umbral definido como $N_U = N/2$, esto es:

$$\hat{N}_{TME} = \frac{1}{\sum_{i=1}^N (\tilde{w}_k^i)^2} \quad (\text{B.2.13})$$

De esta forma, es posible realizar una estimación de los estados con una mayor precisión que el Filtro de Kalman Extendido. Notar que en cada iteración se toman muestras de las partículas para los estados junto con los pesos correspondientes durante el desarrollo del algoritmo. De cumplirse la condición antes mencionada de que el tamaño de la muestra es menor al umbral mencionado, se realizan nuevas muestras con pesos uniformes $1/N$, [18]. Ahora para reconstruir la salida, se aplica lo siguiente:

$$\hat{p}(x_k | y_{1:k}) = \sum_{i=1}^N \tilde{w}_k^i \delta(x_k - x_k^i) \quad (\text{B.2.14})$$

$$\hat{x}_k = E(x_k | y_{1:k}) \approx \sum_{i=1}^N \tilde{w}_k^i x_k^i \quad (\text{B.2.15})$$

$$\hat{P}_k = C(x_{0:k} | y_{1:k}) \approx \sum_{i=1}^N \tilde{w}_k^i (\hat{x}_k^i - x_k^i) (\hat{x}_k^i - x_k^i)^T \quad (\text{B.2.16})$$

$$\sqrt{\text{Var}(x_{k|l})} = \frac{1}{N} \sum_{k=1}^N \left(\frac{1}{M} \sum_{j=1}^M (x_{k|l}^i - x_k^i)^2 \right)^{1/2} \quad (\text{B.2.17})$$

Donde x_k^i es el estado verdadero para la simulación j -ésima, $x_{k|l}^i = \sum_{i=1}^N \tilde{w}_k^i x_k^{j,(i)}$ es el estimador de Monte Carlo de $x_{k|l} = E(x_k | y_{1:k})$ para la j -ésima señal de prueba, $x_k^{j,(i)}$ es la i -ésima trayectoria simulada asociada con la señal j , y $\tilde{w}_{k|l}^i = \tilde{w}_k^i$ es el peso de importancia.

La ecuación (B.2.17) se usa para validar los resultados obtenidos y aplicarse como una medida de adecuación. Notar que en este método recursivo se siguen utilizando aproximaciones mediante jacobianos para las matrices del sistema. También la ecuación (B.2.13) nos permite ajustar las muestras en caso de que el tamaño efectivo sea menor que el umbral, ajustando los pesos de las mediciones. Además, la aplicación del filtrado mediante Monte Carlo permite que la aumente la precisión, y aunque no posee la misma precisión que UKF, este set de ecuaciones disminuye el tiempo de ejecución, permitiendo ajustarse con mayor precisión a los valores reales de las mediciones del sistema con menos recursos, [10].

Anexo C. Diagramas de flujo

En la siguiente figura se observa un diagrama de flujo de las distintas secciones de código que componen el algoritmo diseñado.

La primera sección corresponde a una identificación de parámetros en tiempo real del sistema según el modelo que se desarrolló, para luego ordenar los estados y parámetros en un modelo en espacios de estado, el cual es utilizado por el Filtro de Kalman Extendido para realizar el filtrado de la señal y obtener de esta forma con los estados *a posteriori* una salida reconstruida que se ajusta a la salida real del sistema.

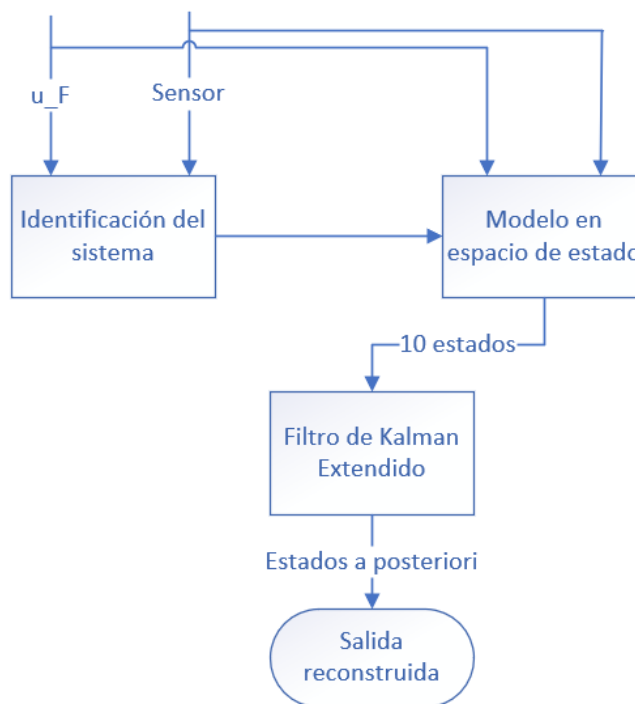


Figura C.1. Diagrama de Flujo del algoritmo

Anexo D. Códigos implementados

D.1. Código Filtro de Kalman Extendido

```
u_F:=FV_02;
y_F:=FT_01;

// INICIO DE MINIMOS CUADRADOS RECURSIVOS

// matriz de regresores
fhi_F[0]:=-y1_F;
fhi_F[1]:=-y2_F;
fhi_F[2]:=-y3_F;
fhi_F[3]:=u_F;
fhi_F[4]:=u1_F;
fhi_F[5]:=u2_F;
fhi_F[6]:=u3_F;

// matriz de covarinza
for i:=0 to nu-1 by 1 do
  for j:=0 to nu-1 by 1 do
    sumaux_F:=fhi_F[j]*P_F[j,i];
    sum_F:= sum_F + sumaux_F;
    sumaux2_F:=P_F[i,j]*fhi_F[j];
    sum2_F:= sum2_F + sumaux2_F;
  end_for;
  aux1_F[i]:=sum_F; //esto es phi*P1 = 1x7
  sum_F:=0;
  sumaux3_F:=aux1_F[i]*fhi_F[i];
  sum1_F:=sum1_F+sumaux3_F;
  aux2_F[i]:=sum2_F; //esto es P1*phi = 7x1
  sum2_F:=0;
end_for;
final_F:=lambda_F+sum1_F; //calculo constante inv(lambda+phi*P1*phi)
sum1_F:=0;
for i:=0 to nu-1 by 1 do
  for j:=0 to nu-1 by 1 do
    sumaux4_F:=aux1_F[i]*aux2_F[j];
    P_F[i,j]:=(1/lambda_F)*(P_F[i,j] - sumaux4_F/final_F); //matriz covarianza
  end_for;
end_for;

// vector de ganancia

for i:=0 to nu-1 by 1 do
  for j:=0 to nu-1 by 1 do
    sumaux5_F:=P_F[i,j]*fhi_F[j];
    sum3_F:=sum3_F+sumaux5_F;
  end_for;
  Ka_F[i]:=sum3_F;
  sum3_F:=0;
```



```

end_for;

// vector de parametros
for j:=0 to nu-1 by 1 do
    sumaux6_F:=fhi_F[j]*theta_F[j];
    sum4_F:=sum4_F+sumaux6_F;
end_for;
for i:=0 to nu-1 by 1 do
    Pred_Error:=y_F - sum4_F;
    Pred_Gain[i]:=Ka_F[i]*Pred_Error;
    theta_F[i]:=theta_F[i] + Pred_Gain[i];
end_for;
sum4_F:=0;

////////////////////////////////////
// CONTINUAR CON FILTRO DE KALMAN

//hacer matrices auxiliares cero
for i:=0 to 9 by 1 do //matrices de 10x10
    for j:=0 to 9 by 1 do
        aux_cov_F[i,j]:=0;
        aux_cov1_F[i,j]:=0;
        aux_pk_F[i,j]:=0;
        Pu_F[i,j]:=0;
    end_for;
end_for;

for i:=0 to 7 by 1 do //matrices de 8x10 o 10x8
    for j:=0 to 9 by 1 do
        aux_s_F[i,j]:=0; //8x10
        aux_k_F[j,i]:=0; //10x8
        K_kf_F[j,i]:=0; //10x8
    end_for;
end_for;

for i:=0 to 7 by 1 do //matrices de 8x8
    for j:=0 to 7 by 1 do
        aux_s1_F[i,j]:=0;
    end_for;
end_for;

//1.- derivada de funcion de estado
//primero hay que declarar la matriz Ak como conjunto de puros 1 y 0
for i:=0 to 9 by 1 do
    for j:=0 to 9 by 1 do
        if i<=2 then
            if i=0 AND j=1 then
                Ak_F[i,j]:=1;
            elseif i=1 AND j=2 then
                Ak_F[i,j]:=1;
            else

```

```

        Ak_F[i,j]:=0;
    end_if;
end_if;
if i>=3 then
    if i=j then
        Ak_F[i,j]:=1;
    else
        Ak_F[i,j]:=0;
    end_if;
end_if;
end_for;
end_for;
//ahora se declaran los valores que cambian de la matriz Ak
Ak_F[2,0]:=-x_kf_F[9];
Ak_F[2,1]:=-x_kf_F[8];
Ak_F[2,2]:=-x_kf_F[7];
Ak_F[2,7]:=-x_kf_F[2];
Ak_F[2,8]:=-x_kf_F[1];
Ak_F[2,9]:=-x_kf_F[0];

//Ecuaciones de prediccion (modelo en espacio de estados)

x1_est_F:=x_kf_F[1];
x2_est_F:=x_kf_F[2];
x3_est_F:=-x_kf_F[9]*x_kf_F[0]-x_kf_F[8]*x_kf_F[1]-x_kf_F[7]*x_kf_F[2]+u_F;
K0kf_est_F:=x_kf_F[3];
K1kf_est_F:=x_kf_F[4];
K2kf_est_F:=x_kf_F[5];
K3kf_est_F:=x_kf_F[6];
K4kf_est_F:=x_kf_F[7];
K5kf_est_F:=x_kf_F[8];
K6kf_est_F:=x_kf_F[9];

//Matriz de covarianza del error
// P_prev=Ak*Pk*Ak'+Q;

for i:=0 to 9 by 1 do
    for j:=0 to 9 by 1 do
        for k:=0 to 9 by 1 do
            aux_cov_F[i,k]:=aux_cov_F[i,k]+Ak_F[i,j]*Pk_F[j,k];
        end_for;
    end_for;
end_for;

for i:=0 to 9 by 1 do
    for j:=0 to 9 by 1 do
        for k:=0 to 9 by 1 do
            aux_cov1_F[i,k]:=aux_cov1_F[i,k]+aux_cov_F[i,j]*Ak_F[k,j];
        end_for;
    end_for;
end_for;
for j:=0 to 9 by 1 do

```

```

for k:=0 to 9 by 1 do
    P_prev_F[j,k]:=aux_cov1_F[j,k]+Q_F[j,k];
end_for;
end_for;

//primero se declara Hk como solo ceros
for i:=0 to 7 by 1 do
    for j:=0 to 9 by 1 do
        Hk_F[i,j]:=0;
    end_for;
end_for;
//declarar coordenadas de Hk que tienen valor
Hk_F[1,3]:=1;
Hk_F[2,4]:=1;
Hk_F[3,5]:=1;
Hk_F[4,6]:=1;
Hk_F[5,7]:=1;
Hk_F[6,8]:=1;
Hk_F[7,9]:=1;
Hk_F[0,0]:=K3kf_est_F-K6kf_est_F*K0kf_est_F;
Hk_F[0,1]:=K2kf_est_F-K5kf_est_F*K0kf_est_F;
Hk_F[0,2]:=K1kf_est_F-K4kf_est_F*K0kf_est_F;
Hk_F[0,3]:=-K6kf_est_F*x1_est_F-K5kf_est_F*x2_est_F-K4kf_est_F*x1_est_F+u_F;
Hk_F[0,4]:=x3_est_F;
Hk_F[0,5]:=x2_est_F;
Hk_F[0,6]:=x1_est_F;
Hk_F[0,7]:=-K0kf_est_F*x3_est_F;
Hk_F[0,8]:=-K0kf_est_F*x2_est_F;
Hk_F[0,9]:=-K0kf_est_F*x1_est_F;

//vector de mediciones y vector de estados
//

z_med_F[0]:=y_F;
z_med_F[1]:=theta_F[3];
z_med_F[2]:=theta_F[4];
z_med_F[3]:=theta_F[5];
z_med_F[4]:=theta_F[6];
z_med_F[5]:=theta_F[0];
z_med_F[6]:=theta_F[1];
z_med_F[7]:=theta_F[2];

x_vect_F[0]:=x1_est_F;
x_vect_F[1]:=x2_est_F;
x_vect_F[2]:=x3_est_F;
x_vect_F[3]:=K0kf_est_F;
x_vect_F[4]:=K1kf_est_F;
x_vect_F[5]:=K2kf_est_F;
x_vect_F[6]:=K3kf_est_F;
x_vect_F[7]:=K4kf_est_F;
x_vect_F[8]:=K5kf_est_F;
x_vect_F[9]:=K6kf_est_F;

```

```

//salida prueba para ganancia de correccion de kalman h(x_vect,0)
x_pb_F:=(x_kf_F[6]-x_kf_F[3]*x_kf_F[9])*x_kf_F[0] + (x_kf_F[5]-x_kf_F[3]*x_kf_F[8])*x_kf_F[1] +
(x_kf_F[4]-x_kf_F[3]*x_kf_F[7])*x_kf_F[2]+x_kf_F[3]*u_F;
h_prueba_F[0]:=x_pb_F;
h_prueba_F[1]:=K0kf_est_F;
h_prueba_F[2]:=K1kf_est_F;
h_prueba_F[3]:=K2kf_est_F;
h_prueba_F[4]:=K3kf_est_F;
h_prueba_F[5]:=K4kf_est_F;
h_prueba_F[6]:=K5kf_est_F;
h_prueba_F[7]:=K6kf_est_F;

//6.- ecuaciones de correccion
//S_kf=H*P_prev*H' + R;

for i:=0 to 7 by 1 do
  for j:=0 to 9 by 1 do
    for k:=0 to 9 by 1 do
      aux_s_F[i,k]:=aux_s_F[i,k]+Hk_F[i,j]*P_prev_F[j,k];
    end_for;
  end_for;
end_for;
for i:=0 to 7 by 1 do
  for j:=0 to 9 by 1 do
    for k:=0 to 7 by 1 do
      aux_s1_F[i,k]:=aux_s1_F[i,k]+aux_s_F[i,j]*Hk_F[k,j];
    end_for;
  end_for;
end_for;
for j:=0 to 7 by 1 do
  for k:=0 to 7 by 1 do
    S_kf_F[j,k]:=aux_s1_F[j,k]+R_F[j,k];
  end_for;
end_for;

//K_kf=P_prev*H*inv(S_kf);
for i:=0 to 9 by 1 do
  for j:=0 to 9 by 1 do
    for k:=0 to 7 by 1 do
      aux_k_F[i,k]:=aux_k_F[i,k]+P_prev_F[i,j]*Hk_F[k,j];
    end_for;
  end_for;
end_for;

//inversa de S_kf
for i:=0 to 7 by 1 do
  for j:=0 to 15 by 1 do
    if (j<8) then
      augmat_F[i,j]:=S_kf_F[i,j];
    else
      augmat_F[i,j]:=Id_F[i,j-8];
    end_if;
  end_for;
end_for;

```

```

    end_if;
  end_for;
end_for;

//aplicar eliminacion gauss jordan
for i:=0 to 7 by 1 do
  for j:=0 to 7 by 1 do
    if (i<>j) then
      ratio:=augmat_F[j,i]/augmat_F[i,i];
      for k:=0 to 15 by 1 do
        augmat_F[j,k]:=augmat_F[j,k]-ratio*augmat_F[i,k];
      end_for;
    end_if;
  end_for;
end_for;

//operacion de filas para hacer diagonal principal a 1
for i:=0 to 7 by 1 do
  for j:=8 to 15 by 1 do
    augmat_F[i,j]:=augmat_F[i,j]/augmat_F[i,i];
  end_for;
end_for;

//entregar matriz inversa
for i:=0 to 7 by 1 do
  for j:=8 to 15 by 1 do
    S_inv_F[i,j-8]:=augmat_F[i,j];
  end_for;
end_for;

//multiplicacion de P_prev*H' con la inversa de S_kf
for i:=0 to 9 by 1 do
  for j:=0 to 7 by 1 do
    for k:=0 to 7 by 1 do
      K_kf_F[i,k]:=K_kf_F[i,k]+aux_k_F[i,j]*S_inv_F[j,k];
    end_for;
  end_for;
end_for;

//x_kf=x_vect+K_kf*(z_med - h_prueba);
for i:=0 to 7 by 1 do
  e_kf_F[i]:=z_med_F[i]-h_prueba_F[i];
end_for;
//multiplicacion de k_kf*error
for i:=0 to 9 by 1 do
  for j:=0 to 7 by 1 do
    sumg_F := sumg_F + K_kf_F[i,j]*e_kf_F[j];
  end_for;
  gain_F[i]:=sumg_F;
  sumg_F:=0;
end_for;

```

```

//suma de x_vect + gain
for i:=0 to 9 by 1 do
    x_kf_F[i]:=x_vect_F[i]+gain_F[i];
end_for;

//Pk=(eye(10,10) - K_kf*H)*P_prev; (identidad 10x10 es Ip)
for i:=0 to 9 by 1 do
    for j:=0 to 7 by 1 do
        for k:=0 to 9 by 1 do
            aux_pk_F[i,k]:=aux_pk_F[i,k]+K_kf_F[i,j]*Hk_F[j,k]; //K_kf*H
        end_for;
    end_for;
end_for;

for i:=0 to 9 by 1 do
    for j:=0 to 9 by 1 do
        aux_pk1_F[i,j]:=Ip_F[i,j]-aux_pk_F[i,j]; //(eye(10,10) - K_kf*H)
    end_for;
end_for;

for i:=0 to 9 by 1 do
    for j:=0 to 9 by 1 do
        Pu_F[i,j]:=0;
    end_for;
end_for;

for i:=0 to 9 by 1 do
    for j:=0 to 9 by 1 do
        for k:=0 to 9 by 1 do
            Pu_F[i,k]:=Pu_F[i,k]+aux_pk1_F[i,j]*P_prev_F[j,k]; //(eye(10,10) - K_kf*H)*P_prev;
        end_for;
    end_for;
end_for;

for i:=0 to 9 by 1 do
    for j:=0 to 9 by 1 do
        Pk_F[i,j]:=Pu_F[i,j];
    end_for;
end_for;

//7.- reconstrucción salida;
//y_estk=H*x_kf;

//for i:=0 to 9 by 1 do
//    sumy_F:=sumy_F+Hk_F[0,i]*x_kf_F[i];
//end_for;
y_estk_F:=(x_kf_F[6]-x_kf_F[3]*x_kf_F[9])*x_kf_F[0] + (x_kf_F[5]-x_kf_F[3]*x_kf_F[8])*x_kf_F[1] +
(x_kf_F[4]-x_kf_F[3]*x_kf_F[7])*x_kf_F[2]+x_kf_F[3]*u_F;
//y_estk_F:=sumy_F/2;
sumy_F:=0;

```

```
//Calculo error estimación
e_efk:=((y_F - y_estk_F)/y_F)*100;
```

```
//memorias de proceso y matriz de regresores
u3_F:=u2_F; // u(k-3)
u2_F:=u1_F; // u(k-2)
u1_F:=u_F; // u(k-1)
y3_F:=y2_F; // y(k-3)
y2_F:=y1_F; // y(k-2)
y1_F:=y_F; // y(k-1)
```

D.2. Código Controlador Flujo

```
//Colocar aqui arriba linea de funcion maestro-esclavo al flujo segun nivel
```

```
if sw_m_a1 then spk_f:=uk1_n+sduk_n*abs(duk_n); end_if;
```

```
//error
```

```
ek_f1:=spk_f - y_estk_F; //calculo el error
```

```
ek_f1:=swa_f*ek_f1; //switch de accion al error
```

```
//derivada del error
```

```
dek_f1:=ek_f1 - ek1_f1; //calculo la derivada del error
```

```
//escalamiento de ek y dek
```

```
Es_f1:=ek_f1/Se; //escalo el error y la derivada del error
```

```
Eds_f1:=dek_f1/Sde;
```

```
if Es_f1 > 1 then
```

```
    Es_f1:=1; //se limitan entre 1 y -1 el error y derivada escalados
```

```
end_if;
```

```
if Es_f1 < -1 then
```

```
    Es_f1:=-1;
```

```
end_if;
```

```
if Eds_f1 > 1 then
```

```
    Eds_f1:=1;
```

```
end_if;
```

```

if Eds_f1 < -1 then
    Eds_f1 := -1;
end_if;

//Fuzzificación
//Se escogieron funciones triangulares para Es y Eds

//Error
//Función negativa "N"
if Es_f1 < 0 then MeN_f1 := -Es_f1; end_if;
if Es_f1 >= 0 then MeN_f1 := 0; end_if;

if MeN_f1 > 1 then MeN_f1 := 1; end_if;
if MeN_f1 < 0 then MeN_f1 := 0; end_if;

//Función intermedia cero "Z"
if Es_f1 < 0 then MeZ_f1 := 1 + Es_f1; end_if;
if Es_f1 >= 0 then MeZ_f1 := 1 - Es_f1; end_if;

if MeZ_f1 > 1 then MeZ_f1 := 1; end_if;
if MeZ_f1 < 0 then MeZ_f1 := 0; end_if;

//Función positiva "P"
if Es_f1 <= 0 then MeP_f1 := 0; end_if;
if Es_f1 > 0 then MeP_f1 := Es_f1; end_if;

if MeP_f1 > 1 then MeP_f1 := 1; end_if;
if MeP_f1 < 0 then MeP_f1 := 0; end_if;

//Derivada del Error
//Función negativa "N"
if Eds_f1 < 0 then MdeN_f1 := -Eds_f1; end_if;

```



```
if Eds_f1 >= 0 then MdeN_f1 := 0; end_if;
```

```
if MdeN_f1 > 1 then MdeN_f1 := 1; end_if;
```

```
if MdeN_f1 < 0 then MdeN_f1 := 0; end_if;
```

```
//Función intermedia cero "Z"
```

```
if Eds_f1 < 0 then MdeZ_f1 := 1 + Eds_f1; end_if;
```

```
if Eds_f1 >= 0 then MdeZ_f1 := 1 - Eds_f1; end_if;
```

```
if MdeZ_f1 > 1 then MdeZ_f1 := 1; end_if;
```

```
if MdeZ_f1 < 0 then MdeZ_f1 := 0; end_if;
```

```
//Función positiva "P"
```

```
if Eds_f1 <= 0 then MdeP_f1 := 0; end_if;
```

```
if Eds_f1 > 0 then MdeP_f1 := Eds_f1; end_if;
```

```
if MdeP_f1 > 1 then MdeP_f1 := 1; end_if;
```

```
if MdeP_f1 < 0 then MdeP_f1 := 0; end_if;
```

```
//Motor de inferencia (Toma de decisiones)
```

```
min(Regla_f1, MeN_f1, MdeN_f1, alfa_R1_f1); //N
```

```
min(Regla_f1, MeN_f1, MdeZ_f1, alfa_R2_f1); //N
```

```
min(Regla_f1, MeN_f1, MdeP_f1, alfa_R3_f1); //Z
```

```
min(Regla_f1, MeZ_f1, MdeN_f1, alfa_R4_f1); //N
```

```
min(Regla_f1, MeZ_f1, MdeZ_f1, alfa_R5_f1); //Z
```

```
min(Regla_f1, MeZ_f1, MdeP_f1, alfa_R6_f1); //P
```

```
min(Regla_f1, MeP_f1, MdeN_f1, alfa_R7_f1); //Z
```

```
min(Regla_f1, MeP_f1, MdeZ_f1, alfa_R8_f1); //P
```

```
min(Regla_f1, MeP_f1, MdeP_f1, alfa_R9_f1); //P
```

```
max(Rmax_f1, alfa_R6_f1, alfa_R8_f1, alfa_R9_f1, Mdu_P_f1); // R6 R8 R9
```

```
max(Rmax_f1, alfa_R3_f1, alfa_R5_f1, alfa_R7_f1, Mdu_Z_f1); // R3 R5 R7
```

```

max(Rmax_f1, alfa_R1_f1, alfa_R2_f1, alfa_R4_f1, Mdu_N_f1); // R1 R2 R4

//Defuzzificación
//Si es singleton, valor peso sera en donde se situe
//Si es triangular 3 terminos, sera -0.7071 y 0.07071
//duk:=(Mdu_P-Mdu_N)/(Mdu_P+Mdu_Z+Mdu_N); //Singleton

duk_f1:=(Mdu_P_f1*0.7071 + Mdu_Z_f1*0 + Mdu_N_f1*-0.7071)/(Mdu_P_f1 + Mdu_N_f1 +
Mdu_Z_f1);
duk_f1:=sdu_f*duk_f1;
/*
//guardar signo de la accion de control
if duk_f1>0 then sduk_f1:=1;
elsif duk_f1<0 then sduk_f1:=-1;
else sduk_f1:=0; end_if;

//limitador de cambio en la accion de control
if abs(duk_f1)>dumax_f then duk_f1:=dumax_f; end_if;

//si esta en automatico, se calcula accion del controlador
if swam_f then
    uk_f1:=uk1_f1+sduk_f1*abs(duk_f1);
else
    uk_f1:=uk1_f1; //si esta en manual, se mantiene accion de control
end_if;

// limitador de salida
if uk_f1 > umax_f then uk_f1:=umax_f; end_if;
if uk_f1 < umin_f then uk_f1:=umin_f; end_if;

//entregar salida del controlador a válvula
if swam_f then

```

```

        FV_02:=uk_f1;
else
        uk_f1:=FV_02; spk_f:=FT_01; //setpoint tracking en manual
end_if;

uk1_f1:=uk_f1;
ek2_f1:=ek1_f1;
ek1_f1:=ek_f1;
*/
if duk_f1>0 then sduk_f1:=1;
ELSIF duk_f1<0 then sduk_f1:=-1;
ELSE sduk_f1:=0; end_if;

//limitador de variacion (proteccion del actuador)
if ABS(duk_f1)>dumax_f then duk_f1:=dumax_f; end_if;

if swam_f then uk_f1:=uk1_f1+sduk_f1*ABS(duk_f1); //automatico
else uk_f1:=uk1_f1; end_if; //manual

//limitador de maximo y minimo del actuador (proteccion del actuador)
if uk_f1>umax_f then uk_f1:=umax_f; end_if;
if uk_f1<umin_f then uk_f1:=umin_f; end_if;

//enviar resultado de accionamiento a valvula correspondiente
if swam_f then FV_02:=uk_f1;
else uk_f1:=FV_02; spk_f:=FT_01; end_if;

//memoria actuador
uk1_f1:=uk_f1;
//memoria error
ek2_f1:=ek1_f1;
ek1_f1:=ek_f1;

```

D.3. Código Controlador Nivel

```
if sw_m_a1 then swa_n:=-1.0; //(Cambia la acción del controlador a inversa)
else swa_n:=1.0; end_if;
```

```
//error
```

```
ek_n:= spk_n - y_estk_N;
```

```
//Switch del accionamiento (+1/-1) (cambia el error de signo)
```

```
ek_n:=ek_n*swa_n;
```

```
//derivada del error
```

```
dek_n:= ek_n - ek1_n;
```

```
//Escalamiento del error y su derivada
```

```
Es_n:= ek_n/Se_n;
```

```
Eds_n:= dek_n/Sde_n;
```

```
if Es_n > 1 then Es_n:=1; end_if;
```

```
if Es_n < -1 then Es_n:=-1; end_if;
```

```
if Eds_n > 1 then Eds_n:=1; end_if;
```

```
if Eds_n < -1 then Eds_n:=-1; end_if;
```

```
//Fuzzificación
```

```
//Función negativa "N"
```

```
if Es_n < 0 then MeN_N := -Es_n; end_if;
```

```
if Es_n >= 0 then MeN_N := 0; end_if;
```

```
if MeN_N>1 then MeN_N := 1; end_if;
```

```
if MeN_N<0 then MeN_N := 0; end_if;
```

```
//Función intermedia cero "Z"
```

```
if Es_n<0 then MeZ_n := 1 + Es_n; end_if;  
if Es_n>=0 then MeZ_n := 1 - Es_n; end_if;
```

```
if MeZ_n>1 then MeZ_n := 1; end_if;  
if MeZ_n<0 then MeZ_n := 0; end_if;
```

```
//Función positiva "P"
```

```
if Es_n <= 0 then MeP_N := 0; end_if;  
if Es_n > 0 then MeP_N := Es_n; end_if;
```

```
if MeP_N>1 then MeP_N := 1; end_if;  
if MeP_N<0 then MeP_N := 0; end_if;
```

```
//Derivada del Error
```

```
//Función negativa "N"
```

```
if Eds_n < 0 then MdeN_N := -Eds_n; end_if;  
if Eds_n >= 0 then MdeN_N := 0; end_if;
```

```
if MdeN_N>1 then MdeN_N := 1; end_if;  
if MdeN_N<0 then MdeN_N := 0; end_if;
```

```
//Función intermedia cero "Z"
```

```
if Eds_n<0 then MdeZ_N := 1 + Eds_n; end_if;  
if Eds_n>=0 then MdeZ_N := 1 - Eds_n; end_if;
```

```
if MdeZ_N>1 then MdeZ_N := 1; end_if;  
if MdeZ_N<0 then MdeZ_N := 0; end_if;
```

```
//Función positiva "P"
```

```
if Eds_n <= 0 then MdeP_N := 0; end_if;  
if Eds_n > 0 then MdeP_N := Eds_n; end_if;
```

```
if MdeP_N>1 then MdeP_N := 1; end_if;
if MdeP_N<0 then MdeP_N := 0; end_if;
```

```
//Motor de inferencia (Toma de decisiones)
```

```
min(Regla_n, MeN_N, MdeN_N, alfa_R1_N); //N
min(Regla_n, MeN_N, MdeZ_N, alfa_R2_N); //N
min(Regla_n, MeN_N, MdeP_N, alfa_R3_N); //Z
min(Regla_n, MeZ_n, MdeN_N, alfa_R4_N); //N
min(Regla_n, MeZ_n, MdeZ_N, alfa_R5_N); //Z
min(Regla_n, MeZ_n, MdeP_N, alfa_R6_N); //P
min(Regla_n, MeP_N, MdeN_N, alfa_R7_N); //Z
min(Regla_n, MeP_N, MdeZ_N, alfa_R8_N); //P
min(Regla_n, MeP_N, MdeP_N, alfa_R9_N); //P
```

```
max(Rmax_n, alfa_R6_N, alfa_R8_N, alfa_R9_N, MduP_N);
max(Rmax_n, alfa_R3_N, alfa_R5_N, alfa_R7_N, MduZ_N);
max(Rmax_n, alfa_R1_N, alfa_R2_N, alfa_R4_N, MduN_N);
```

```
//Defuzzificación
```

```
// duk_n:=(MduP_n-MduN_n)/(MduP_n+MduZ_n+MduN_n); //Singleton
duk_n:=(MduP_N*0.7071 + MduZ_N*0 + MduN_N*-0.7071)/(MduP_N+MduZ_N+MduN_N);
duk_n:=Sdu_N*duk_n;
```

```
//guardar signo de la accion de control
```

```
if duk_n>0 then sduk_n:=1;
elsif duk_n<0 then sduk_n:=-1;
else sduk_n:=0; end_if;
```

```
//limitador cambio de la entrada
```

```
if abs(duk_n)>dumax_n then duk_n:=dumax_n; end_if;
```

```
//si esta en automatico, se calcula accion del controlador
```

```

if swam_n then
    uk_n:=uk1_n+sduk_n*abs(duk_n);
else
    uk_n:=uk1_n; //si esta en manual, se mantiene accion de control
end_if;

//limitador de minimo y maximo
if uk_n > umax_n then uk_n:=umax_n; end_if;
if uk_n < umin_n then uk_n:=umin_n; end_if;

//entregar salida del controlador a válvula
if swam_n AND NOT sw_m_a1 then FV_01:=uk_n; //(Valvula de carga ahora es perturbacion)
elsif swam_n AND sw_m_a1 then FV_01:=FV_01;
else FV_01:=FV_01; spk_n:=LT_01; end_if;

//memorias
uk1_n:=uk_n;
ek2_n:=ek1_n;
ek1_n:=ek_n;

```

D.4. Código Controlador Temperatura

```

//error
ek_t:=spk_t - y_estk_T;

//derivada del error
dek_t:=ek_t - ek1_t;

//escalamiento de ek y dek
Es_t:=ek_t/Se_t;
Eds_t:=dek_t/Sde_t;

if Es_t > 1 then Es_t:=1; end_if;
if Es_t < -1 then Es_t:=-1; end_if;

```

```

if Eds_t > 1 then Eds_t:=1; end_if;
if Eds_t < -1 then Eds_t:=-1; end_if;

//Fuzzificación
//Se escogieron funciones triangulares para Es y Eds

//Error
//Función negativa "N"
if Es_t < 0 then MeN_t := -Es_t; end_if;
if Es_t >= 0 then MeN_t := 0; end_if;

if MeN_t>1 then MeN_t := 1; end_if;
if MeN_t<0 then MeN_t := 0; end_if;

//Función intermedia cero "Z"
if Es_t<0 then MeZ_t := 1 + Es_t; end_if;
if Es_t>=0 then MeZ_t := 1 - Es_t; end_if;

if MeZ_t>1 then MeZ_t := 1; end_if;
if MeZ_t<0 then MeZ_t := 0; end_if;

//Función positiva "P"
if Es_t <= 0 then MeP_t := 0; end_if;
if Es_t > 0 then MeP_t := Es_t; end_if;

if MeP_t>1 then MeP_t := 1; end_if;
if MeP_t<0 then MeP_t := 0; end_if;

//Derivada del Error
//Función negativa "N"
if Eds_t < 0 then MdeN_t := -Eds_t; end_if;

```



```
if Eds_t >= 0 then MdeN_t := 0; end_if;
```

```
if MdeN_t > 1 then MdeN_t := 1; end_if;
```

```
if MdeN_t < 0 then MdeN_t := 0; end_if;
```

```
//Función intermedia cero "Z"
```

```
if Eds_t < 0 then MdeZ_t := 1 + Eds_t; end_if;
```

```
if Eds_t >= 0 then MdeZ_t := 1 - Eds_t; end_if;
```

```
if MdeZ_t > 1 then MdeZ_t := 1; end_if;
```

```
if MdeZ_t < 0 then MdeZ_t := 0; end_if;
```

```
//Función positiva "P"
```

```
if Eds_t <= 0 then MdeP_t := 0; end_if;
```

```
if Eds_t > 0 then MdeP_t := Eds_t; end_if;
```

```
if MdeP_t > 1 then MdeP_t := 1; end_if;
```

```
if MdeP_t < 0 then MdeP_t := 0; end_if;
```

```
//Motor de inferencia (Toma de decisiones)
```

```
min(Regla_T, MeN_t, MdeN_t, alfa_R1_t); //N
```

```
min(Regla_T, MeN_t, MdeZ_t, alfa_R2_t); //N
```

```
min(Regla_T, MeN_t, MdeP_t, alfa_R3_t); //Z
```

```
min(Regla_T, MeZ_t, MdeN_t, alfa_R4_t); //N
```

```
min(Regla_T, MeZ_t, MdeZ_t, alfa_R5_t); //Z
```

```
min(Regla_T, MeZ_t, MdeP_t, alfa_R6_t); //P
```

```
min(Regla_T, MeP_t, MdeN_t, alfa_R7_t); //Z
```

```
min(Regla_T, MeP_t, MdeZ_t, alfa_R8_t); //P
```

```
min(Regla_T, MeP_t, MdeP_t, alfa_R9_t); //P
```

```
max(Rmax_T, alfa_R6_t, alfa_R8_t, alfa_R9_t, MduP_t); // R6 R8 R9
```

```
max(Rmax_T, alfa_R3_t, alfa_R5_t, alfa_R7_t, MduZ_t); // R3 R5 R7
```

```

max(Rmax_T, alfa_R1_t, alfa_R2_t, alfa_R4_t, MduN_t); // R1 R2 R4

//Defuzzificación
//Si es singleton, valor peso sera en donde se situe
//Si es triangular 3 terminos, sera -0.7071 y 0.07071
//duk:=(Mdu_P-Mdu_N)/(Mdu_P+Mdu_Z+Mdu_N); //Singleton

duk_t:=(MduP_t*0.7071 + MduZ_t*0 + MduN_t*-0.7071)/(MduP_t + MduN_t + MduZ_t);
duk_t:=Sdu_t*duk_t;

//guardar signo de la accion de control
if duk_t>0 then sduk_t:=1;
elseif duk_t<0 then sduk_t:=-1;
else sduk_t:=0; end_if;

//limitador de cambio en la accion de control
if abs(duk_t)>dumax_t then duk_t:=dumax_t; end_if;

//si esta en automatico, se calcula accion del controlador
if swam_t then
    uk_t:=uk1_t+sduk_t*abs(duk_t);
else
    uk_t:=uk1_t; //si esta en manual, se mantiene accion de control
end_if;

// limitador de salida
if uk_t > umax_t then uk_t:=umax_t; end_if;
if uk_t < umin_t then uk_t:=umin_t; end_if;

//entregar salida del controlador a válvula
if swam_t then
    FV_03:=uk_t;

```

```
else
    FV_03:=FV_03; spk_t:=TT_01; //setpoint tracking en manual
end_if;
```

```
//memorias
uk1_t:=uk_t;
ek2_t:=ek1_t;
ek1_t:=ek_t;
```

**UNIVERSIDAD DE CONCEPCION – FACULTAD DE INGENIERIA
RESUMEN DE MEMORIA DE TITULO**

Departamento : Departamento de Ingeniería Eléctrica
Carrera : Ingeniería civil electrónica
Nombre del memorista : José Matías Hernández Ramírez
Título de la memoria : Implementación de un Filtro de Kalman Extendido en tiempo real
Fecha de la presentación oral : 10/01/2023

Profesor(es) guía : Juan Pablo Segovia Vera
Profesor(es) revisor(es) : Luis Arias/Sergio Torres
Concepto :
Calificación :

Resumen (máximo 200 palabras)

En las mediciones, el ruido añadido a las lecturas y el retardo que el sistema posee al momento de hacer cambios en éste, dificulta las labores de los controladores.

El presente proyecto consiste en el diseño e implementación de un Filtro de Kalman Extendido en tiempo real a través de un PLC. Se estudia el comportamiento de los distintos lazos que conforman el sistema con la finalidad de diseñar un modelo que se ajuste a la respuesta de la planta para flujo, nivel y temperatura.

Se diseñan e implementan algoritmos capaces de realizar una identificación de parámetros del modelo antes realizado y posteriormente, a través del Filtro de Kalman Extendido, se realiza una estimación de la salida y los parámetros para obtener una mejor señal con la que trabajar.

Finalmente se diseña un controlador capaz de realizar cambios deseados en los lazos del sistema en base a una referencia, haciendo uso de la salida del Filtro de Kalman Extendido para obtener mejores resultados.