

**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA**



**DISEÑO DE OBSERVADORES CONTINUOS PARA DIAGNÓSTICO DE
FALLAS A PARTIR DE DATOS MUESTREADOS**

POR

ING. LUIS ANGEL SÁNCHEZ RIVERA

**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE
MAestrÍA EN CIENCIAS DE LA INGENIERÍA ELÉCTRICA**

OCTUBRE, 2018

**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO**



TESIS

**DISEÑO DE OBSERVADORES CONTINUOS PARA DIAGNÓSTICO DE
FALLAS A PARTIR DE DATOS MUESTREADOS**

POR

ING. LUIS ANGEL SÁNCHEZ RIVERA

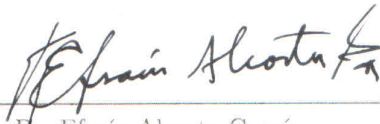
**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE
MAESTRÍA EN CIENCIAS DE LA INGENIERÍA ELÉCTRICA**

OCTUBRE, 2018

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica
Subdirección de Estudios de Posgrado

Los miembros del comité de tesis recomendamos que la tesis "Diseño de observadores continuos para diagnóstico de fallas a partir de datos muestreados", realizada por el alumno Ing. Luis Angel Sánchez Rivera, con número de matrícula 1510646, sea aceptada para su defensa como requisito parcial para obtener el grado de Maestría en Ciencias de la Ingeniería Eléctrica.

El Comité de Tesis



Dr. Efraín Alcorta García

Asesor


Dr. David Alejandro Díaz Romero

Revisor


Dra. Diana Hernández Alcántara

Revisor

Vo. Bo.


Dr. Simón Martínez Martínez

Subdirección de Estudios de Posgrado



San Nicolás de los Garza, Nuevo León, a 30 de octubre de 2018

ÍNDICE GENERAL

Agradecimientos	IX
Resumen	XI
1. Introducción	1
1.1. Hipótesis	3
1.2. Objetivos	3
1.2.1. Generales	3
1.2.2. Particulares	3
1.3. Metodología	4
2. Preliminares	5
2.1. Estado del arte	5
2.2. Terminología utilizada en el diagnóstico de fallas	7
2.3. Generación residual	10
2.3.1. Generador de Residuo basado en Relaciones de Paridad (PRRG)	10
2.3.2. Observador de Diagnóstico (DO)	14

2.4. Algoritmo discreto 15

 2.4.1. Generador de residuos en tiempo discreto 15

 2.4.2. Aislamiento de fallas en sistemas discretos 18

 2.4.2.1. Algoritmo discreto para aislamiento de fallas 19

2.5. Identificación de sistemas 19

 2.5.1. Procedimiento general de identificación 21

 2.5.2. Identificación por subespacios 22

 2.5.2.1. Problema de identificación 22

3. Generación de residuos en tiempo continuo 24

 3.1. Descripción del capítulo 24

 3.2. Generación de residuos 25

 3.2.1. Relación sistema filtrado-sistema original 25

 3.2.2. Identificación del espacio nulo izquierdo 28

 3.2.3. Representación computacional del espacio nulo izquierdo 31

 3.2.4. Generación de residuo a partir de datos basada en observador 34

 3.2.4.1. Algoritmo 1 (Diseño de un observador continuo para
detección de fallas) 36

 3.2.5. Ejemplo con algoritmo 1 36

 3.2.5.1. Sistema de 3 tanques 36

 3.2.5.2. Resultados mediante simulación 37

ÍNDICE DE FIGURAS

2.1. Sistema de diagnóstico de fallas	6
2.2. Generador de residuo basado en observador	15
2.3. Obtención de un modelo matemático a partir de identificación	20
3.1. Obtención de matrices de datos	28
3.2. Sistema de 3 tanques	37
3.3. Caso sin falla	38
3.4. Caso con falla	39
4.1. Esquema para de detección de fallas	41
4.2. Esquema para aislamiento de fallas	42
4.3. Esquema de observadores generalizados	42
4.4. Modelo esquemático de un PMSM de 2 polos	47
4.5. Residuo 1 en ausencia de falla	51
4.6. Residuo 2 en ausencia de falla	51
4.7. Residuo 1 con falla en sensor	52

4.8. Residuo 2 con falla en sensor	52
4.9. Residuo 1 con falla en actuador	52
4.10. Residuo 2 con falla en actuador	53

AGRADECIMIENTOS

Agradecimientos

Agradezco primeramente a Dios, por permitirme llegar hasta este punto con salud, por obsequiarme a la familia más amorosa y a los mejores amigos; por estar siempre a mi lado guiándome y por poner a mi lado a tantas personas maravillosas que me han ayudado en mi trayectoria para ser lo que soy y que siguen ahí para apoyarme.

A mi director de tesis, Dr. Efraín Alcorta García, por creer en mí desde un principio, por transmitirme su conocimiento y guiarme con paciencia en la elaboración de la tesis; a quién le guardo un gran respeto y admiración. Su manera de ser y su dedicación se han convertido en mi fuente de motivación.

A mi familia, que siempre han estado a mi lado para darme su apoyo incondicional, para guiarme y aconsejarme cuando lo necesito, por confiar en mí para la realización de todos mis sueños. Siempre serán mi ejemplo a seguir, los amo.

A mis amigos Jorge, Manuel, Ivon y Alfonso, que han estado a mi lado para escucharme y apoyarme en los momentos difíciles, y para sacarme risas cuando más lo necesitaba.

A todos mis maestros, desde que tengo memoria, han sido parte fundamental en quien soy ahora, los admiro por su labor.

A todas las personas que han influido para la realización de esta investigación,

extiendo mis más sincero agradecimiento.

Al Consejo Nacional de Ciencia y Tecnología por su apoyo económico para la realización del presente trabajo.

RESUMEN

Ing. Luis Angel Sánchez Rivera.

Candidato para obtener el grado de Maestría en Ciencias de la Ingeniería Eléctrica.

Universidad Autónoma de Nuevo León.

Facultad de Ingeniería Mecánica y Eléctrica.

Título del estudio: DISEÑO DE OBSERVADORES CONTINUOS PARA DIAGNÓSTICO
DE FALLAS A PARTIR DE DATOS MUESTREADOS.

Número de páginas: 57.

OBJETIVOS Y MÉTODO DE ESTUDIO: El objetivo general del presente trabajo es desarrollar un algoritmo para el aislamiento de fallas en sistemas lineales continuos. Se pretende extender los resultados previamente obtenidos en sistemas discretos al caso continuo, donde se utilizan únicamente mediciones de entrada-salida del sistema.

Como metodología se tiene realizar una revisión de la bibliografía referente a la generación de residuos a partir de datos de un sistema, los cuales están inspirados en métodos basados en observadores. Se realizará una lectura técnica de diagnóstico en sistemas continuos para proponer un enfoque de solución. Se prosigue con proponer un enfoque que nos permita tener residuos desacoplados de ciertas fallas.

CONTRIBUCIONES Y CONCLUSIONES: En el presente trabajo se trata con un problema de aislamiento de fallas en sistemas lineales continuos.

Combinando las ventajas de dos generadores de residuos propuestos en la literatura (Generador de residuos basado en relaciones de paridad y el Observador de diagnóstico) junto con la idea de identificación por subespacios, es posible la construcción de un grupo de generadores de residuo agrupados que desacoplan fallas cuando sólomente se tienen datos del proceso.

Se demostró que es posible la construcción de un algoritmo para aislamiento de fallas en sistemas lineales continuos utilizando únicamente mediciones del sistema. Un enfoque que no utiliza un modelo matemático del sistema en cuestión para diagnóstico, es propuesto. Además, se incluye una justificación matemática de los resultados obtenidos.

Por último, se comprobó mediante software, que es posible detectar y aislar fallas a partir de datos sin necesidad de discretizar el sistema. Gracias a los resultados del presente tema de tesis, se tiene el artículo “*Enfoque en Tiempo Continuo para Aislamiento de de Fallas a partir de Datos Muestreados*” aceptado para el Congreso Nacional de Control Automático (AMCA) 2018.

Firma del asesor: _____
Dr. Efraín Alcorta García

CAPÍTULO 1

INTRODUCCIÓN

Con el constante desarrollo de la tecnología y debido a sus exigencias, han surgido sistemas de control cada vez más y más sofisticados, un ejemplo son los sistemas de control de vuelo, los cuales por su complejidad utilizan una gran cantidad de sensores y actuadores. Al tratarse de componentes físicos, estos pueden fallar sin previo aviso y ocasionar daños o deterioro en el equipo, con lo cual el funcionamiento de éste se vuelve inestable, provocando en el peor caso un accidente catastrófico. Motivados por las exigencias de confiabilidad, estabilidad y seguridad en los sistemas, el diagnóstico de fallas se ha consolidado en los últimos años desde sus inicios por la década de los setenta.

Muchos métodos basados en modelos han sido propuestos desde entonces [1], [2]; sin embargo, la principal característica de ellos es que requieren el modelo matemático del sistema, el cual no siempre es posible de obtener, y en muchos casos se requiere de un análisis matemático complejo. Una de las técnicas ampliamente reportadas en la literatura hace uso de observadores de salida [1]. Los observadores, como forma de acceder a las variables no medibles de un sistema, fueron introducidos en 1964 por D. Luenberger [3], no obstante su aplicación para el diagnóstico de fallas comenzó en 1971 por la propuesta realizada por el M.I.T. [4], a partir de ahí se desarrolla intensamente por más de 30 años.

La herramienta utilizada para realizar diagnóstico de fallas basado en modelos es la redundancia analítica, la cual se refiere al uso de modelos matemáticos del sistema en consideración para llevar a cabo la detección de las fallas a partir de la generación de un residuo [1]. El residuo es una señal indicadora de falla. En un escenario libre de perturbaciones y suponiendo que se tiene un conocimiento perfecto del sistema, normalmente el residuo será cero en el caso de no existir ninguna falla y diferente de cero cuando se presente una falla. Cuando cualquier tipo de ruido o perturbación está presente en el sistema, es necesario considerar una cota o umbral en el residuo, a partir del cual se considera que se ha producido una falla, el objetivo de dicho umbral es disminuir (y eliminar) falsas alarmas en la detección [5].

Existen diferentes métodos en la literatura para la generación de residuos [2], no obstante, este trabajo se enfoca en la utilización de observadores como generadores de residuo, el residuo se obtiene como una diferencia entre una salida de medición del sistema menos un estimado (con fallas de interés). Una vez generada la señal residual, se analizará con el fin de determinar si existe o no falla en el sistema a supervisar.

La principal característica de los métodos que existen, es que están fueron desarrollados en tiempo discreto. Es importante tomar en cuenta esta situación a la hora de llevar a cabo el diagnóstico; ya que, si éste se realiza considerando que el sistema es discreto, pudiera haber acoplamiento entre fallas (debido al proceso implícito de discretización), lo cual dificultaría el poder aislar las fallas.

Este trabajo trata el tema de aislamiento de fallas en sistemas continuos utilizando generadores de residuo basados en observadores que se diseñan a partir de datos entrada-salida. Se propone utilizar el principio del algoritmo para aislamiento en sistemas discretos para la solución en sistemas en tiempo continuo. Las bases del trabajo se encuentra reportada en [6] y [7].

1.1 HIPÓTESIS

Es posible detectar y aislar fallas por medio del residuo en sistemas lineales continuos, utilizando como única información disponible mediciones de entrada y salida del sistema.

1.2 OBJETIVOS

1.2.1 GENERALES

Extender los resultados de detección y localización de fallas en sistemas lineales de tiempo discreto al caso continuo.

1.2.2 PARTICULARES

El desglose del objetivo general es como sigue:

- Rehacer los resultados existentes en la literatura para la generación residual directamente de datos en tiempo discreto.
- Construir un algoritmo que permita realizar detección y localización de fallas en sistemas lineales continuos a partir de muestras.
- Desarrollo de un programa en lenguaje MATLAB, u otro similar, para implementar el algoritmo en cuestión.
- Verificación del del algoritmo.

1.3 METODOLOGÍA

La estrategia propuesta para alcanzar el objetivo general (y los objetivos particulares) es la siguiente:

- Se realizará una revisión bibliográfica referente a generación de residuos a partir de datos y se hará una revisión de algoritmo para el caso discreto.
- Realizar una lectura técnica para caso continuo.
- En base a la revisión bibliográfica se propondrá e implementará un algoritmo para el caso continuo.
- Se programará y probará el algoritmo con diferentes datos para asegurarse de obtener el resultado deseado.
- Reporte de resultados.

CAPÍTULO 2

PRELIMINARES

2.1 ESTADO DEL ARTE

Una falla puede ser entendida como un cambio inesperado en el funcionamiento del sistema, aunque no siempre representa una avería física o descompostura total. Un *sistema de diagnóstico de fallas* (Fig. (2.1)) consiste en un sistema de monitoreo para detectar fallas, saber de donde se originan y determinar su relevancia. Tales actividades son:

- **Detección de fallas:** Sólo se indica si ocurrió la falla y, de ser así, el tiempo en el que ocurrió.
- **Localización de fallas:** Se determina en dónde ocurrió la falla, ya sea en un sensor, actuador o componente (aislamiento o desacoplo).
- **Identificación de fallas:** Se determina el tamaño, tipo o naturaleza de la falla.

El diagnóstico es independiente del objetivo de control, se puede ver como una protección adicional a un sistema que ya tiene un control implementado.

Al inicio, el diagnóstico de fallas se realizaba mediante redundancia física, es



Figura 2.1: Sistema de diagnóstico de fallas

decir, el uso de elementos repetidos en el sistema. De esta manera se comparaba el funcionamiento de los diferentes elementos y se llegaba a una conclusión para determinar la existencia de fallas. Sin embargo, no siempre es posible utilizar redundancia física en la práctica debido al costo, tamaño y peso de los elementos [8].

Posteriormente, en la década de los setenta, el concepto de redundancia analítica fue introducido, el cual hace referencia al conocimiento del modelo matemático, a las señales de entrada y salida del sistema en cuestión. La madurez del trabajo con modelos puede apreciarse en los libros [2], [5]. A pesar de que empezó hace muchos años, los métodos basados en modelos siguen siendo populares en la comunidad de diagnóstico [9].

Existen también métodos que combinan simplicidad y eficiencia, su principal característica es que utilizan datos históricos del proceso. Entre ellos están: métodos basados en inteligencia artificial, lógica difusa y redes neuronales [10].

No obstante, la idea del presente trabajo surge gracias al Prof. Steven X. Ding en conjunto con su grupo de trabajo formado en el 2004 y sus aportes con métodos basados en datos. El resumen de los principales métodos reportados a la fecha se encuentra en [7]. La característica de ellos es que combinan ventajas de los métodos basados en modelos y de los métodos basados en datos para realizar diagnóstico. Actualmente, los métodos que utilizan datos de entrada y salida del proceso para generar residuos continúan en desarrollo.

Si sumamos la llegada de la cuarta revolución industrial o industria 4.0, la cual busca mejorar los procesos en las industrias integrado nuevas tecnologías de proce-

samiento de datos, digitalización de procesos, una mejor interacción entre software y hardware, entre otras [11], los métodos basados en datos para la generación de residuos recibirán mayor atención en los años siguientes.

A pesar de que ya existen algunos algoritmos propuestos para obtener residuos utilizando solamente datos de entrada y salida, muchos son en tiempo discreto. Existe una técnica basada en observadores para la generación de residuos a partir de datos, presentada en [7]. Recientemente se utilizó el algoritmo de dicha técnica y se hizo una modificación para desacoplar fallas de no interés [12].

En esta tesis se pretende extender los resultados obtenidos para aislamiento de fallas en sistemas discretos al caso continuo, el enfoque utilizado se basa en observadores de salida para la obtención de residuos desacoplados de las fallas de no interés. Se hace uso del algoritmo propuesto en [6], el cual utiliza una primer etapa de filtrado para prescindir de la dificultad de diferenciación, posteriormente se hace uso de la técnica de subespacios con el fin de calcular las matrices del observador. Esto permitirá aislar fallas mediante la utilización de datos con falla y sin falla del proceso en cuestión.

2.2 TERMINOLOGÍA UTILIZADA EN EL DIAGNÓSTICO DE FALLAS

Al entrar en un campo de estudio, es fundamental conocer la terminología que se utiliza para así tener un mejor entendimiento de lo que se desea estudiar, a continuación se presenta la principal terminología utilizada por la comunidad de diagnóstico, la mayoría fueron extraídas de [13], [14].

Con el fin de distinguir los conceptos de falla, avería y disfunción, las definiciones son las siguientes:

- **Falla:** Cambio de un parámetro fuera de los márgenes de tolerancia.
- **Avería:** Interrupción permanente de la capacidad de un sistema para realizar una función requerida bajo las condiciones de operación especificadas.
- **Disfunción:** Irregularidad intermitente en el cumplimiento de una función deseada del sistema.

Clasificación de las fallas por lugar de ocurrencia durante la operación del sistema:

- **Fallas en los sensores:** Estas fallas actúan directamente sobre las mediciones del proceso.
- **Fallas en los actuadores:** Estas fallas ocasionan cambios en los actuadores. Dado este tipo de falla, puede afectar la controlabilidad de la planta.
- **Fallas en los componentes:** Estas fallas se usan para indicar un mal funcionamiento en los componentes internos del sistema, es por eso que son llamadas "fallas internas".

Tipos de fallas más comunes:

- **Falla multiplicativa:** Es la manera natural de modelar fallas. Se presenta mediante un término en el cual se está multiplicando al estado o a la entrada (cambio del valor de un parámetro).
- **Falla aditiva:** En este caso las fallas son modeladas como entradas adicionales al proceso.

Clasificación de las fallas por dependencia temporal:

- **Abrupta:** Es donde la falla se manifiesta con un cambio repentino y puede ser modelado mediante una función tipo escalón.

- **Incipiente:** Esta falla se manifiesta como un cambio de magnitud en las variables del sistema que va aumentando paulatinamente con respecto al tiempo y puede modelarse mediante una función tipo rampa con un tiempo de ocurrencia desconocido.
- **Intermitente:** Es cuando la falla aparece y desaparece sin un patrón determinado, se considera que no tiene una evolución determinada en el tiempo.

Requerimientos en el diagnóstico de fallas:

- **Seguridad:** Ausencia de peligro.
- **Fiabilidad:** Es la probabilidad del sistema de ejecutar sus funciones normales durante un intervalo de tiempo.
- **Disponibilidad:** Es la probabilidad de que un sistema opere cuando sea necesario.

Pasos para el diagnóstico de fallas:

- **Generación de residuos.** Es la obtención de señales que dependen únicamente de las fallas. En el caso ideal, un residuo será cero si no existen fallas presentes y diferente de cero si se presenta una falla.
- **Evaluación de residuo:** Es la extracción de la información sobre la falla. Saber su magnitud, tipo o naturaleza.

El *residuo* es un indicador de falla. Algunas propiedades deseables para el residuo son las siguientes:

- Tener valores cercanos a cero para los modos de operación normal.

- Que algún residuo alcance valores diferentes a cero para los modos de operación con fallas.
- Ser muy sensible a fallas.
- Respuesta rápida a posibles fallas.

2.3 GENERACIÓN RESIDUAL

En un sistema de diagnóstico de fallas, la generación residual es la primer etapa, consiste en el cálculo de las diferencias entre las variables medidas y las estimadas del modelo en cuestión.

Una gran variedad de esquemas se encuentran disponibles para generación residual [1], [2], [5]. En esta sección se incluyen únicamente dos técnicas de generación de residuos:

- Generador de Residuo basado en Relaciones de Paridad (PRRG)
- Observador de Diagnóstico (DO)

Aunque el presente trabajo aborda el tema de generación de residuos para sistemas lineales en tiempo continuo, las bases fueron desarrolladas en tiempo discreto, y se utilizará el mismo principio de ambas técnicas para la construcción del generador de residuos continuo a partir de muestras.

2.3.1 GENERADOR DE RESIDUO BASADO EN RELACIONES DE PARIDAD (PRRG)

Esta técnica para sistemas dinámicos fue desarrollada en [15], es ampliamente aceptada debido a su simplicidad computacional y a su fácil implementación, ya que

una de sus ventajas en la generación de residuos recae en que la dinámica de las señales de residuo de acuerdo a las fallas y entradas desconocidas es presentada en forma de ecuaciones algebraicas. La idea principal es revisar la paridad (consistencia) de las ecuaciones matemáticas del sistema construidas a partir únicamente de datos medibles. A continuación, se presenta la descripción tomada de [2].

Consideramos un sistema lineal en tiempo discreto

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) + E_d d(k) + E_f f(k) \\y(k) &= Cx(k) + Du(k) + F_d d(k) + F_f f(k)\end{aligned}\tag{2.1}$$

donde $x(k)$ es el estado de dimensión n , $u(k)$ es el vector de entradas de dimensión p , $d(k)$ es el vector de perturbaciones, $f(k)$ es el vector de fallas. Las matrices $A, B, C, D, E_d, E_f, F_d$ y F_f son constantes, conocidas y de dimensiones apropiadas. Suponer que el par (A, C) es observable y que $\text{rango}(C) = m$.

Con la finalidad de construir el generador de residuos suponemos primero que el efecto de las perturbaciones y las fallas es cero, es decir, $d(k) = 0$ y $f(k) = 0$. Expresando $y(k-s)$, con $s > 0$, en términos de $x(k-s)$, $u(k-s)$ y $y(k-s+1)$, en términos de $x(k-s)$, $u(k-s)$, $u(k-s+1)$ resulta:

$$\begin{aligned}y(k-s) &= Cx(k-s) + Du(k-s) \\y(k-s+1) &= Cx(k-s+1) + Du(k-s+1) \\&= CAx(k-s) + CBu(k-s) + Du(k-s+1).\end{aligned}$$

Repetiendo este proceso se obtiene la representación de salida para el tiempo actual k .

$$\begin{aligned}y(k-s+2) &= CA^2x(k-s) + CABu(k-s) + CBu(k-s+1) + Du(k-s+2) \\&\vdots \\y(k) &= CA^s x(k-s) + CA^{s-1} Bu(k-s) + \dots + CBu(k+1) + Du(k).\end{aligned}$$

Podemos introducir la siguiente notación

$$y_s(k) = \begin{bmatrix} y(k-s) \\ y(k-s+1) \\ \vdots \\ y(k) \end{bmatrix}; \quad u_s(k) = \begin{bmatrix} u(k-s) \\ u(k-s+1) \\ \vdots \\ u(k) \end{bmatrix}$$

$$H_{o,s} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^s \end{bmatrix}; \quad H_{u,s} = \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1}B & \cdots & CB & D \end{bmatrix} \quad (2.2)$$

donde $H_{o,s}$ es llamada *matriz de observabilidad extendida*. Entonces se puede re-escribir el conjunto de ecuaciones desarrolladas en forma compacta como:

$$y_s(k) = H_{o,s}x(k-s) + H_{u,s}u_s(k) \quad (2.3)$$

La ecuación (2.3) es conocida como la relación de paridad. Note que los vectores de entradas $u_s(k)$ y de salidas $y_s(k)$ se forman con las mediciones y se suponen conocidas. Las matrices $H_{o,s}$ y $H_{u,s}$ son constantes y formadas con las matrices del sistema, las cuales se suponen conocidas. La única variable desconocida es $x(k-s)$.

La idea básica de la relación de paridad es utilizar el hecho de que si $s \geq n$ entonces $\text{rango}(H_{o,s}) = n$ es menor que el número de renglones de $H_{o,s}$, con lo cual existe al menos un vector renglón $\nu_s \neq 0$ de dimensión $(s+1)m$ tal que

$$\nu_s H_{o,s} = 0 \quad (2.4)$$

El generador de residuos basado en relaciones de paridad (PRRG) es construido por:

$$r(k) = \nu_s(y_s(k) - H_{u,s}u_s(k)) = \nu_s H_{o,s}x(k-s) = 0 \quad (2.5)$$

Los vectores ν_s que satisfacen (2.4) son llamados vectores de paridad y al conjunto de vectores de paridad que satisfacen:

$$P_s = \{\nu_s \mid \nu_s H_{o,s} = 0\} \quad (2.6)$$

es llamado espacio de paridad de orden s -ésimo.

Para ver el efecto de las fallas y perturbaciones en el residuo ahora se remueve la restricción utilizada ($d(k) = 0$ y $f(k) = 0$). A continuación se repite el procedimiento utilizado para el caso sin falla. De esta forma se obtienen las siguientes matrices:

$$f_s(k) = \begin{bmatrix} f(k-s) \\ f(k-s+1) \\ \vdots \\ f(k) \end{bmatrix}; \quad d_s(k) = \begin{bmatrix} d(k-s) \\ d(k-s+1) \\ \vdots \\ d(k) \end{bmatrix}$$

$$H_{f,s} = \begin{bmatrix} F_f & 0 & \cdots & 0 \\ CE_f & F_f & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1}E_f & \cdots & CE_f & F_f \end{bmatrix}; \quad H_{d,s} = \begin{bmatrix} F_d & 0 & \cdots & 0 \\ CE_d & F_d & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1}E_d & \cdots & CE_d & F_d \end{bmatrix}$$

Entonces las ecuaciones para $y_s(k)$ puede escribirse como:

$$y_s(k) = H_{o,s}x(k-s) + H_{u,s}u_s(k) + H_{f,s}f_s(k) + H_{d,s}d_s(k)$$

así el generador de residuos queda como:

$$r_s(k) = \nu_s((H_{f,s}f_s(k) + H_{d,s}d_s(k))), \quad \nu_s \in P_s \quad (2.7)$$

de esta manera solo queda en función de las posibles fallas y perturbaciones del sistema, y el unico parámetro de diseño es el vector de paridad.

Aunque el diseño de un generador residual basado en las ecuaciones de paridad puede construirse de una manera relativamente sencilla, la implementación en línea no es la ideal debido a que los valores pasados de entrada, así como los de salida necesitan ser almacenados. Esta es una de las razones por las cuales el enfoque de

espacio de paridad es aplicado principalmente a sistemas dinámicos discretos y es por esto que es requerida otra herramienta matemática que compense esta desventaja.

2.3.2 OBSERVADOR DE DIAGNÓSTICO (DO)

El observador de diagnóstico (DO) es, junto con el método de espacio de paridad, uno de los generadores de residuos más investigados [7] debido a su estructura flexible y compatibilidad que guardan entre ambos, además que es similar al observador de Luenberger.

Mientras que para control se estima lo que no se puede medir, este generador de residuo es un observador de salida y es descrito por:

$$\begin{aligned} z(k+1) &= Gz(k) + Hu(k) + Ly(k) \\ r(k) &= Vy(k) - Wz(k) - Qu(k) \end{aligned} \quad (2.8)$$

donde $z \in \mathfrak{R}^s$, denota el orden del observador, puede ser igual, menor o mayor que el orden del sistema n .

Asumir que $G_{yu}(z) = C(zI - A)^{-1}B + D$, entonces G, H, L, Q, V y W junto con una matriz de transformación $T \in \mathfrak{R}^{s \times n}$ tiene que satisfacer las llamadas ecuaciones de Luenberger,

1. G es estable.
2. $A - GT = LC, \quad H = TB - LD.$
3. $VC - WT = 0, \quad Q = VD.$

con las cuales el sistema (2.8) proporciona un vector residual, que es

$$\forall u, x(0), \lim_{k \rightarrow \infty} r(k) = 0$$

. Por lo tanto, el observador de diagnóstico será el generador de residuo Fig. (2.2).

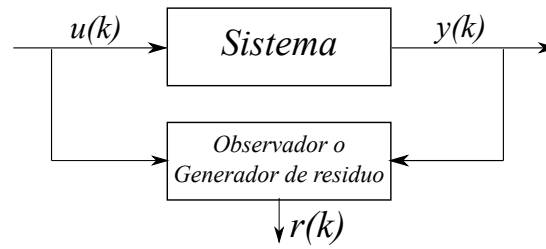


Figura 2.2: Generador de residuo basado en observador

2.4 ALGORITMO DISCRETO

Diferentes esquemas de algoritmos para detección de fallas han sido propuestos en la literatura [7], en esta sección se muestra que para un sistema discreto puede ser construido un generador de residuos combinando las herramientas PRRG (lazo abierto) y DO (lazo cerrado) antes mencionadas a partir de datos entrada-salida, gracias a la relación que guardan al poseer dinámicas idénticas bajo la condición de que sus valores propios sean *ceros*, esto en el caso de sistemas lineales invariantes en el tiempo [2]. La metodología de identificación por subespacios es parte fundamental al momento de querer identificar las matrices del observador.

La idea base del algoritmo para detección proviene de [7], sin embargo, la modificación para lograr localización de fallas en sistemas discretos fue propuesta en [12].

2.4.1 GENERADOR DE RESIDUOS EN TIEMPO DISCRETO

Siguiendo el concepto del espacio de paridad para la construcción del generador de residuos, consideramos un sistema discreto lineal e invariante en el tiempo (LTI):

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned} \tag{2.9}$$

donde el enfoque de espacio de paridad es como sigue

$$r(k) = v_s [Y - HU] = \Psi^\perp \begin{bmatrix} Y \\ U \end{bmatrix} = 0 \quad (2.10)$$

tenemos que $Y = \begin{bmatrix} y^T(k-s) & y^T(k-s+1) & \cdots & y^T(k) \end{bmatrix}^T$,
 $U = \begin{bmatrix} u^T(k-s) & u^T(k-s+1) & \cdots & u^T(k) \end{bmatrix}^T$ y la matriz H se define como:

$$H = \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & & 0 \\ \vdots & & \ddots & \vdots \\ CA^{s-1}B & CA^{s-2}B & \cdots & D \end{bmatrix}$$

El espacio nulo izquierdo Ψ^\perp es introducido. Usando ideas de las técnicas de identificación por subespacios [16], el sistema es representado como

$$Y_f = \Gamma_s X_{k-s} + H_{u,s} U_f \quad (2.11)$$

donde

$$U_f = \begin{bmatrix} u(k-s) & u(k-s+1) & \cdots & u(k-s+N-1) \\ u(k-s+1) & u(k-s+2) & \cdots & u(k-s+N) \\ \vdots & \vdots & \ddots & \vdots \\ u(k) & u(k+1) & \cdots & u(k+N-1) \end{bmatrix}$$

$$H_{u,s} = \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & & 0 \\ \vdots & & \ddots & \vdots \\ CA^{s-1}B & CA^{s-2}B & \cdots & D \end{bmatrix}, \quad \Gamma_s = \begin{bmatrix} C \\ \vdots \\ CA^s \end{bmatrix}$$

y X_{k-s} es el vector de estados pasados.

La matriz Y_f se construye de manera similar a U_f , pero reemplazando $u(\cdot)$ por $y(\cdot)$ y reciben el nombre de matrices de Hankel futuras. Una vez que fueron definidas, la ecuación (2.11) puede ser representada como

$$\begin{bmatrix} U_f \\ Y_f \end{bmatrix} = \begin{bmatrix} I & 0 \\ H_{s,u} & \Gamma_s \end{bmatrix} \begin{bmatrix} U_f \\ X_{k-s} \end{bmatrix} \quad (2.12)$$

Para detectar fallas, el residuo puede ser construido con la ayuda del espacio nulo izquierdo:

$$r(k) = \Psi^\perp \begin{bmatrix} U_f \\ Y_f \end{bmatrix} = 0 \quad (2.13)$$

esto para el caso en el que $f = 0$, es decir, al momento que ocurra una falla, la relación de paridad (2.13) no se cumplirá y por lo tanto la falla será detectada. Es importante mencionar que para la construcción del residuo con el espacio nulo izquierdo solamente datos entrada-salida son requeridos. Ahora, utilizando lo anterior, es posible construir un generador de residuos basado en observador [7]. Introduciendo entonces

$$\begin{aligned} z(k+1) &= A_d z(k) + B_d u(k) + L_d y(k) \\ r(k) &= G_d y(k) - C_d z(k) - D_d u(k) \end{aligned} \quad (2.14)$$

donde

$$\begin{aligned} A_d &= \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix}, \quad \Psi^\perp = \begin{bmatrix} \Psi_u^\perp & \Psi_y^\perp \end{bmatrix} \\ L_d &= - \begin{bmatrix} \Psi_y^\perp(1:m) \\ \vdots \\ \Psi_y^\perp((s-1)m+1:sm) \end{bmatrix} \\ B_d &= - \begin{bmatrix} \Psi_u^\perp(1:\ell) \\ \vdots \\ \Psi_y^\perp((s-1)\ell+1:s\ell) \end{bmatrix} \\ G_d &= \Psi_y^\perp((sm+1):(s+1)m) \\ D_d &= \Psi_u^\perp(s\ell+1:(s+1)\ell) \\ C_d &= \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.15)$$

m representa la dimensión del vector de salida y ℓ la del vector de entrada.

Puede distinguirse que habrá una primera etapa de diseño seguida de una segunda etapa de implementación.

2.4.2 AISLAMIENTO DE FALLAS EN SISTEMAS DISCRETOS

El hecho de utilizar únicamente datos entrada-salida otorga la posibilidad de almacenar los datos con falla, es decir, si datos con falla son recolectados para la etapa de diseño, es posible diseñar generadores de residuos insensibles a esas fallas, siempre y cuando dos fallas no afecten el mismo canal.

Si se consideran ahora una falla aditiva \bar{f} en (2.9)

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + E_1\bar{f}(k) \\ y(k) &= Cx(k) + Du(k) + F_1\bar{f}(k) \end{aligned} \quad (2.16)$$

Permite una representación distinta a (2.12) en una ventana de tiempo

$$\begin{bmatrix} U_f \\ Y_f \end{bmatrix} = \begin{bmatrix} I & 0 \\ H_{s,u} & \Gamma_s \end{bmatrix} \begin{bmatrix} U_f \\ X_{k-s} \end{bmatrix} + \begin{bmatrix} 0 \\ H_{s,\bar{f}}\bar{F}_{\bar{f}} \end{bmatrix} \quad (2.17)$$

donde

$$H_{s,\bar{f}} = \begin{bmatrix} F_1 & 0 & \cdots & 0 \\ CE_1 & F_1 & & 0 \\ \vdots & & \ddots & \vdots \\ CA^{s-1}E_1 & CA^{s-2}E_1 & \cdots & F_1 \end{bmatrix}$$

Debido a que la falla esta afectado en los datos entrada y salida, el residuo que se obtiene se vuelve insensible a esa falla:

$$\begin{aligned} r(k) &= \Psi^\perp \begin{bmatrix} U_f \\ Y_f \end{bmatrix} = 0 \\ &= \Psi^\perp \left\{ \begin{bmatrix} I & 0 \\ H_{s,u} & \Gamma_s \end{bmatrix} \begin{bmatrix} U_f \\ X_{k-s} \end{bmatrix} + \begin{bmatrix} 0 \\ H_{s,\bar{f}}\bar{F}_{\bar{f}} \end{bmatrix} \right\} \end{aligned} \quad (2.18)$$

Por lo tanto es posible diseñar un banco de generadores de residuos con tantas combinaciones como se desée. A esto se le conoce como el *Esquema de observadores*

generalizados [1]. La única información requerida en este caso son los datos con falla para el diseño.

Si se quieren aislar dos o más fallas, la información de esas fallas en datos de entrada-salida será requerida también.

Entonces el algoritmo para lograr aislamiento en sistemas discretos es como sigue:

2.4.2.1 ALGORITMO DISCRETO PARA AISLAMIENTO DE FALLAS

1. *Considerar un conjunto de datos con la información de falla i .*
2. *Formar las matrices de Hankel futuras Y_f y U_f .*
3. *Calcular Ψ^\perp .*
4. *Construir el generador de residuos basado en observador utilizando ecuaciones en (2.15).*

2.5 IDENTIFICACIÓN DE SISTEMAS

Un modelo dinámico puede ser usado para describir muchos fenómenos, tales como: biológicos, económicos, industriales, etc. Los hay de diferentes tipos: mentales, verbales, intuitivos, sin embargo, los de mayor interés en nuestro caso son los modelos matemáticos. Estos pueden ser descritos como ecuaciones diferenciales (para sistemas continuos) o en diferencias (para sistemas discretos). Nos son útiles porque describen el comportamiento dinámico del sistema en función del tiempo y por lo tanto, se puede experimentar con ellos [16].

La identificación es un complemento al modelado de sistemas. Mientras que mediante modelado se obtienen modelos analíticos a partir de leyes físicas, median-

te identificación representan el camino experimental (Fig. (2.3)) y dichos modelos pueden ser:

- **No paramétricos:** Se obtienen mediante identificación sin tener en cuenta ninguna estructura o el modelado.
- **Paramétricos:** Se obtienen los parámetros de la estructura del modelo.

Las técnicas de identificación de sistemas han recibido gran interés en el área de control, debido a que no siempre es posible contar con el modelo matemático que mejor describe al sistema, y en algunos casos, su comportamiento no es el más cercano.

La identificación de sistemas surge como una necesidad de crear modelos matemáticos sencillos y precisos para la experimentación con sistemas complejos a partir de mediciones que son interpretadas como observaciones ruidosas [17].

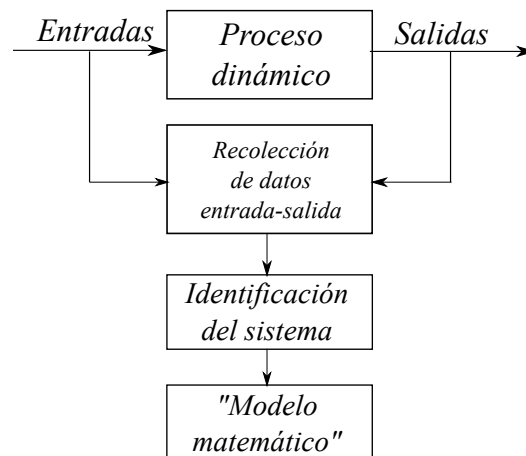


Figura 2.3: Obtención de un modelo matemático a partir de identificación

2.5.1 PROCEDIMIENTO GENERAL DE IDENTIFICACIÓN

En términos generales, el proceso de identificación involucra los siguientes pasos:

- **Recolección de datos.** Los datos entrada-salida son obtenidos mediante la excitación del sistema con una señal de entrada para posteriormente ser tratados y evaluados, de esta manera se eliminan ruidos indeseados o cualquier otro tipo de imperfección que impida obtener el mejor modelo.
- **Obtención de un conjunto de modelos.** Se selecciona un conjunto de modelos de los cuales se seleccionará el más adecuado, si de antemano se conoce que el modelo es paramétrico, se determina su estructura. Este punto se facilita considerablemente si se posee cierto conocimiento sobre las leyes físicas que rigen el proceso.
- **Selección del mejor modelo.** La selección se basa principalmente en que tanto se parece el modelo identificado al proceso real al momento de reproducir su funcionamiento.
- **Validación del modelo.** Por último queda validar si el modelo seleccionado tiene un buen desempeño para la aplicación en cuestión. Si se llega a la conclusión de que el modelo no es válido, se deben considerar los siguientes aspectos:
 - a) Los datos entrada-salida utilizados no proporcionan información suficiente, por lo que es necesario recolectar nuevos datos.
 - b) La estructura seleccionada no es la que mejor describe al sistema.

En el caso de que el modelo sea aceptado, se dice que es una buena representación del proceso original.

2.5.2 IDENTIFICACIÓN POR SUBESPACIOS

Los métodos por subespacios son una rama de algoritmos de identificación de sistemas en el espacio de estados [16].

Algunas características de las técnicas de subespacios son las siguientes:

- Combina herramientas de teoría de sistemas, álgebra lineal numérica y geometría (proyecciones).
- No se requiere de procedimientos iterativos de optimización, lo que implica que no hay problema con mínimos locales, convergencia e inicialización.
- Provee modelos confiables de sistemas LTI multivariables directamente de datos entrada-salida.
- No se requiere de una realización particular (forma canónica) lo cual mejora el condicionamiento numérico.
- Se requiere de una capacidad computacional modesta en comparación con los métodos tradicionales de error de predicción.
- Los algoritmos asociados están implementados de forma eficiente en MATLAB.
- La identificación por subespacios utiliza la idea diferente a los métodos tradicionales de error de predicción.

2.5.2.1 PROBLEMA DE IDENTIFICACIÓN

En identificación por subespacios, los modelos matemáticos son descritos por las siguientes ecuaciones en diferencias

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) + \omega(k) \\ y(k) &= Cx(k) + Du(k) + \nu(k)\end{aligned}\tag{2.19}$$

Los vectores $\omega(k)$ y $\nu(k)$ son señales no medibles, $\nu(k)$ es denominado ruido en la medición y $\omega(k)$ ruido en el proceso, cuyas sucesiones son consideradas con promedio igual a cero, es decir, como ruido blanco en estado estacionario.

El primer paso en los métodos de subespacio es estimar el subespacio expandido por las columnas de la matriz de observabilidad extendida (2.2), de allí el nombre. Las diferencias en las distintas técnicas recaen en la manera en que es obtenida la matriz de observabilidad extendida y en cómo es usada para estimar las matrices del sistema.

Entonces el problema matemático de identificación puede formularse como: Dado el conjunto de mediciones de entrada y salida u_1, \dots, u_s , y y_1, \dots, y_s , determinar el orden n del sistema y las matrices A, B, C , y D que mejor ajusten los datos en algún sentido bien definido.

Para un mejor entendimiento del algoritmo éste puede encontrarse en [16]. En este trabajo se identificarán las matrices del observador de diagnóstico en lugar de las matrices del sistema (2.19).

CAPÍTULO 3

GENERACIÓN DE RESIDUOS EN TIEMPO CONTINUO

3.1 DESCRIPCIÓN DEL CAPÍTULO

La generación residual es el primer paso para realizar diagnóstico de fallas. Como se vió anteriormente, un generador de residuos en tiempo discreto basado en datos puede ser construido utilizando la filosofía basada en modelos.

Para la construcción de un generador de residuos en tiempo continuo se utiliza el mismo principio, sin embargo, el primer objetivo será establecer una relación entre las matrices del sistema filtrado con las matrices del sistema original, de esta forma el problema de diferenciación puede ser superado.

En este capítulo se presenta un esquema para generación de residuos en tiempo continuo basado en observador directamente de muestras entrada-salida, el algoritmo utilizado como base se encuentra en [6]. Posteriormente, se hace uso del algoritmo en el sistema de tres tanques para probar su eficiencia.

3.2 GENERACIÓN DE RESIDUOS

3.2.1 RELACIÓN SISTEMA FILTRADO-SISTEMA ORIGINAL

Considerando el sistema continuo lineal invariante en el tiempo (LTI)

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) + E_f f(t) \\ y(t) &= Cx(t) + Du(t) + F_f f(t)\end{aligned}\tag{3.1}$$

donde $x \in \mathfrak{R}^n$ es el vector de estado, $u \in \mathfrak{R}^{k_u}$ el vector de entradas de control, $y \in \mathfrak{R}^m$ el vector de salidas medidas, $f \in \mathfrak{R}^{k_f}$ el vector de fallas, $A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times k_u}$, $C \in \mathfrak{R}^{m \times n}$, $D \in \mathfrak{R}^{m \times k_u}$, $E_f \in \mathfrak{R}^{n \times k_f}$, $F_f \in \mathfrak{R}^{m \times k_f}$ son matrices constantes de dimensiones compatibles. Se asume además que el sistema es controlable y observable.

Recordando la relación de paridad descrita en (2.4) y la matriz $H_{u,s}$ descrita en (2.2) se definen los vectores ν_s y $\rho_s = \nu_s H_{u,s}$ como

$$\nu_s = \begin{bmatrix} \nu_{s,0} & \nu_{s,1} & \cdots & \nu_{s,s} \end{bmatrix}, \nu_{s,i} \in \mathfrak{R}^{1 \times m}\tag{3.2}$$

$$\rho_s = \begin{bmatrix} \rho_{s,0} & \rho_{s,1} & \cdots & \rho_{s,s} \end{bmatrix}, \rho_{s,i} \in \mathfrak{R}^{1 \times k_u}\tag{3.3}$$

La intención es utilizar la teoría de los sistemas discretos en sistemas continuos, por lo que se mostrará posteriormente como puede ser obtenido el vector de paridad para estos sistemas.

Se sabe que la diferenciación de datos en el dominio del tiempo puede ser perjudicial en los sistemas de control, ya que, de realizarse en la práctica puede introducir ruidos indeseables en el sistema en cuestión, además que los cálculos posteriores se vuelven numéricamente inestables.

Con el objetivo de prescindir de tal procedimiento, un enfoque basado en el dominio de la frecuencia fue introducido en [18]. Ya que nuestro objetivo es en el dominio del

tiempo, se presenta a continuación otro enfoque que hace uso de un filtro pasa-bajas.

Se define el siguiente filtro pasa-bajas

$$\omega = \frac{1}{1 + \tau s} \quad (3.4)$$

donde τ es un número positivo cualquiera que se elige libremente.

Del sistema (3.1) se hace la transformada de Laplace con $f(t) = 0$ y $x(0) = 0$.

$$sX(s) = AX(s) + BU(s) \quad (3.5)$$

$$Y(s) = CX(s) + DU(s) \quad (3.6)$$

Despejando s de (3.4) y sustituyendo en (3.5)

$$X(s) - \omega X(s) = \tau\omega AX(s) + \tau\omega BU(s)$$

Dado que también $\omega = 1 - \tau s\omega$

$$\begin{aligned} X(s) + \tau\omega sX(s) &= X(s) + \tau\omega AX(s) + \tau\omega BU(s) \\ &= (I + \tau\omega A)X(s) + \tau\omega BU(s) \\ \rightarrow X(s) &= (I + \tau\omega A - \tau\omega sI)X(s) + \tau\omega BU(s) \end{aligned}$$

Simplificando gracias a la misma igualdad de ω nos queda

$$X = (I + \tau A)\omega X(s) + \tau B\omega U(s) \quad (3.7)$$

Multiplicando ambos lados de la ecuación de salida (3.6) por ω^s

$$\omega^s Y(s) = C(\omega^s X(s)) + D(\omega^s U(s)) \quad (3.8)$$

De manera similar al espacio de paridad, se obtienen las salidas anteriores del sistema

$$\omega^{s-1} Y(s) = C(\omega^{s-1} X(s)) + D(\omega^{s-1} U(s)) \quad (3.9)$$

sustituyendo (3.7) resulta

$$\omega^{s-1}Y(s) = C(I + \tau A)\omega^s X(s) + C\tau B\omega^s U(s) + D(\omega^{s-1}U(s)) \quad (3.10)$$

Por lo tanto, de manera recursiva puede obtenerse

$$\begin{aligned} Y(s) = & C(I + \tau A)^s \omega^s X(s) + C(I + \tau A)^{s-1} \tau B \omega^s U(s) + \dots + \\ & C\tau B \omega U(s) + DU(s) \end{aligned} \quad (3.11)$$

De las ecuaciones anteriores puede definirse la siguiente ecuación de entradas y salidas

$$\begin{bmatrix} \omega^s Y(s) \\ \omega^{s-1} Y(s) \\ \vdots \\ Y(s) \end{bmatrix} = \begin{bmatrix} C \\ C(I + \tau A) \\ \vdots \\ C(I + \tau A)^s \end{bmatrix} \omega^s X(s) + \begin{bmatrix} D & 0 & \dots & 0 \\ \tau CB & D & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \tau C(I + \tau A)^{s-1} B & \dots & \tau CB & D \end{bmatrix} \begin{bmatrix} \omega^s U(s) \\ \omega^{s-1} U(s) \\ \vdots \\ U(s) \end{bmatrix} \quad (3.12)$$

A su vez pueden definirse

$$H_{o\tau} = \begin{bmatrix} C \\ C(I + \tau A) \\ \vdots \\ C(I + \tau A)^s \end{bmatrix}; \quad H_{s\tau} = \begin{bmatrix} D & 0 & \dots & 0 \\ \tau CB & D & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \tau C(I + \tau A)^{s-1} B & \dots & \tau CB & D \end{bmatrix} \quad (3.13)$$

Recordando que la ecuación (3.12) se encuentra en el dominio de Laplace, se obtiene su equivalente en el dominio del tiempo

$$\begin{bmatrix} \mathcal{L}^{-1}(\omega^s Y(s)) \\ \mathcal{L}^{-1}(\omega^{s-1} Y(s)) \\ \vdots \\ \mathcal{L}^{-1}(Y(s)) \end{bmatrix} = H_{o\tau} \mathcal{L}^{-1}(\omega^s X(s)) + H_{s\tau} \begin{bmatrix} \mathcal{L}^{-1}(\omega^s U(s)) \\ \mathcal{L}^{-1}(\omega^{s-1} U(s)) \\ \vdots \\ \mathcal{L}^{-1}(U(s)) \end{bmatrix} \quad (3.14)$$

donde $\mathcal{L}^{-1}(\omega^i U(s))$ y $\mathcal{L}^{-1}(\omega^i Y(s))$ representan la transformada inversa de Laplace de $\omega^j U(s)$ y $\omega^j Y(s)$.

Los vectores de entrada y salida $\mathcal{L}^{-1}(\omega^i U(s))$ y $\mathcal{L}^{-1}(\omega^i Y(s))$ con $i = 1, \dots, s$ pueden obtenerse filtrando las señales de entrada y salida utilizando un filtro pasabajas ω^i , respectivamente, como se muestra en Fig. (3.1). El parámetro τ podrá ser seleccionado libremente aunque es importante mencionar que para una cantidad de datos muy grande, su valor deberá ser pequeño y viceversa.

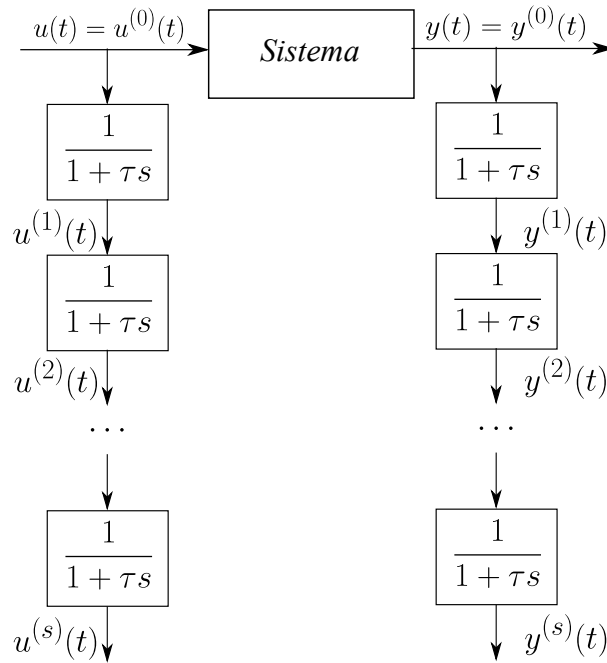


Figura 3.1: Obtención de matrices de datos

3.2.2 IDENTIFICACIÓN DEL ESPACIO NULO IZQUIERDO

Suponiendo que N muestras de las señales filtradas $\mathcal{L}^{-1}(\omega^i U(s)), \mathcal{L}^{-1}(\omega^i Y(s))$ son tomadas en tiempo t_1, \dots, t_N .

Se introduce por conveniencia la siguiente notación

$$y^{(i)}(t) = \mathcal{L}^{-1}(\omega^i Y(s)), \quad y^{(i)}(k) = y^{(i)}(t) |_{t=t_k}$$

$$\begin{aligned} u^{(i)}(t) &= \mathcal{L}^{-1}(\omega^i U(s)), & u^{(i)}(k) &= u^{(i)}(t) |_{t=t_k} \\ x^{(s)}(t) &= \mathcal{L}^{-1}(\omega^s X(s)), & x^{(s)}(k) &= x^{(s)}(t) |_{t=t_k} \end{aligned}$$

con $i = 0, 1, \dots, s$ y $k = 1, 2, \dots, N$. El número de columnas N debe seleccionarse de tal manera que todos los datos sean usados, del algoritmo discreto sabemos que normalmente es igual a $(Ndat - 1) - 2k + 1$. Si se considera la ecuación (3.14) con cada muestra, tenemos

$$Y_N = H_{o\tau} X_N + H_{s\tau} U_N \quad (3.15)$$

donde $Y_N \in \mathfrak{R}^{(s+1)m \times N}$, $U_N \in \mathfrak{R}^{(s+1)k_u \times N}$, $X_N \in \mathfrak{R}^{n \times N}$ están definidas como

$$\begin{aligned} U_N &= \begin{bmatrix} u^{(s)}(1) & u^{(s)}(2) & \dots & u^{(s)}(N) \\ u^{(s-1)}(1) & u^{(s-1)}(2) & \dots & u^{(s-1)}(N) \\ \vdots & \vdots & & \vdots \\ u^{(0)}(1) & u^{(0)}(2) & \dots & u^{(0)}(N) \end{bmatrix} \\ Y_N &= \begin{bmatrix} y^{(s)}(1) & y^{(s)}(2) & \dots & y^{(s)}(N) \\ y^{(s-1)}(1) & y^{(s-1)}(2) & \dots & y^{(s-1)}(N) \\ \vdots & \vdots & & \vdots \\ y^{(0)}(1) & y^{(0)}(2) & \dots & y^{(0)}(N) \end{bmatrix} \\ X_N &= \begin{bmatrix} x^{(s)}(1) & x^{(s)}(2) & \dots & x^{(s)}(N) \end{bmatrix} \end{aligned} \quad (3.16)$$

U_N y Y_N son las matrices de Hankel y son calculadas mediante programación. Se requiere que el conjunto de datos N sea recolectado como un vector fila o columna para su inicialización.

Acomodando las matrices de Hankel como

$$W_N = \begin{bmatrix} Y_N \\ U_N \end{bmatrix}, \quad Q_N = \begin{bmatrix} X_N \\ U_N \end{bmatrix} \quad (3.17)$$

$$H = \begin{bmatrix} H_{o\tau} & H_{s\tau} \\ O & I \end{bmatrix}$$

Para una cantidad suficiente de N datos, la matriz Q_N es de rango completo por filas. Por lo tanto, la matriz de datos W_N y la matriz H que representan la estructura del sistema, tienen el mismo espacio nulo izquierdo. Entonces la matriz H puede ser obtenida a partir de la matriz de datos W_N cuya programación es relativamente sencilla.

Realizando la descomposición en valores singulares (SVD) de la matriz de datos W_N

$$W_N = U\Sigma_s V \quad (3.18)$$

donde

$$U = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix}, \quad V = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}$$

$$\Sigma_s = \begin{bmatrix} S & O \\ O & O \end{bmatrix}$$

$U \in \mathfrak{R}^{(s+1)(m+k_u) \times (s+1)(m+k_u)}$, $V \in \mathfrak{R}^{N \times N}$ son matrices ortogonales, $\Sigma_s \in \mathfrak{R}^{(s+1)(m+k_u) \times N}$, $U_{11} \in \mathfrak{R}^{m(s+1) \times (k_u(s+1)+n)}$, $U_{12} \in \mathfrak{R}^{m(s+1) \times (m(s+1)-n)}$, $U_{21} \in \mathfrak{R}^{k_u(s+1) \times (k_u(s+1)+n)}$, $U_{22} \in \mathfrak{R}^{k_u(s+1) \times (m(s+1)-n)}$, $S \in \mathfrak{R}^{(k_u(s+1)+n) \times (k_u(s+1)+n)}$.

Teniendo que $\begin{bmatrix} U'_{12} & U'_{22} \end{bmatrix} W_N = 0$

$$\rightarrow \begin{bmatrix} U'_{12} & U'_{22} \end{bmatrix} \begin{bmatrix} H_{o\tau} & H_{s\tau} \\ O & I \end{bmatrix} \begin{bmatrix} X_N \\ U_N \end{bmatrix} = 0$$

$$\rightarrow \begin{bmatrix} U'_{12} & U'_{22} \end{bmatrix} \begin{bmatrix} H_{o\tau} & H_{s\tau} \\ O & I \end{bmatrix} = 0$$

Entonces podemos acomodar

$$U'_{12} H_{o\tau} = 0$$

$$U'_{12}H_{s\tau} + U'_{22} = 0$$

Como resultado, podemos identificar las matrices $H_{o\tau}^\perp$ y $H_{o\tau}^\perp H_{s\tau}$

$$\begin{aligned} H_{o\tau}^\perp &= U'_{12} \\ H_{o\tau}^\perp H_{s\tau} &= -U'_{22} \end{aligned} \quad (3.19)$$

Es importante recordar que para asegurar la existencia del espacio nulo izquierdo de W_N y H , es decir, que H no es de rango completo por filas, la variable s debe ser escogida suficientemente grande, normalmente se selecciona más grande que el orden del sistema. Sin embargo, se observa que la cantidad de filtros a utilizarse está relacionada con el orden de las matrices de datos y por lo tanto también lo estará con el orden del generador de residuos; por lo que, entre más grande sea s , mayor será la carga computacional.

3.2.3 REPRESENTACIÓN COMPUTACIONAL DEL ESPACIO NULO IZQUIERDO

Hasta ahora se ha demostrado que existe una relación entre la metodología que se sigue para los sistemas discretos y la metodología en sistemas continuos, que aunque no es directa, permite visualizar el camino que falta para alcanzar el objetivo. A continuación se demuestra que las matrices H_o^\perp , $H_o^\perp H_s$ pueden ser obtenidas a partir de las matrices $H_{o\tau}^\perp$, $H_{o\tau}^\perp H_{s\tau}$, para posteriormente indentificar los vectores ν_s y ρ_s .

Para derivar la relación entre $H_{o\tau}^\perp$, $H_{o\tau}^\perp H_{s\tau}$ y H_o^\perp , $H_o^\perp H_s$, notamos que

$$H_{o\tau} = \begin{bmatrix} C \\ C + \tau CA \\ \vdots \\ C_s^0 C + C_s^1 \tau CA + \cdots + C_s^s \tau^s CA^s \end{bmatrix} = \begin{bmatrix} I & O & \cdots & O \\ C_1^0 I & C_1^1 \tau I & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ C_s^0 I & C_s^1 \tau I & \cdots & C_s^s \tau^s I \end{bmatrix} \begin{bmatrix} C \\ CA \\ \vdots \\ CA^s \end{bmatrix}$$

donde C_i^j representa el coeficiente binomial de i y j . Entonces se puede definir

$$P_m = \begin{bmatrix} I_{m \times m} & O & \cdots & O \\ C_1^0 I_{m \times m} & C_1^1 \tau I_{m \times m} & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ C_s^0 I_{m \times m} & C_s^1 \tau I_{m \times m} & \cdots & C_s^s \tau^s I_{m \times m} \end{bmatrix}$$

Por lo tanto,

$$H_{o\tau} = P_m H_{o,s} \quad (3.20)$$

Introduciendo la siguiente matriz

$$P_{k_u} = \begin{bmatrix} I_{k_u \times k_u} & O & \cdots & O \\ C_1^0 I_{k_u \times k_u} & C_1^1 \tau I_{k_u \times k_u} & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ C_s^0 I_{k_u \times k_u} & C_s^1 \tau I_{k_u \times k_u} & \cdots & C_s^s \tau^s I_{k_u \times k_u} \end{bmatrix}$$

Es posible realizar lo siguiente

$$\begin{aligned} H_{s\tau} P_{k_u} &= \begin{bmatrix} D & O & \cdots & O \\ \tau CB & D & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ \tau C(I + \tau A)^{s-1} B & \cdots & \tau CB & D \end{bmatrix} \times \\ &\quad \begin{bmatrix} I_{k_u \times k_u} & O & \cdots & O \\ C_1^0 I_{k_u \times k_u} & C_1^1 \tau I_{k_u \times k_u} & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ C_s^0 I_{k_u \times k_u} & C_s^1 \tau I_{k_u \times k_u} & \cdots & C_s^s \tau^s I_{k_u \times k_u} \end{bmatrix} \\ &= \begin{bmatrix} D & O & \cdots & O \\ \tau CB + D & \tau D & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ \tau C(\tau A + I)^{s-1} B + \cdots + D & \cdots & \tau^s CB + \tau^{s-1} C_s^{s-1} D & \tau^s D \end{bmatrix} \end{aligned}$$

$$= \begin{bmatrix} I_{m \times m} & O & \cdots & O \\ C_1^0 I_{m \times m} & C_1^1 \tau I_{m \times m} & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ C_s^0 I_{m \times m} & C_s^1 \tau I_{m \times m} & \cdots & C_s^s \tau^s I_{m \times m} \end{bmatrix} \times \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1}B & \cdots & CB & D \end{bmatrix} = P_m H_{u,s}$$

Puede observarse que las matrices $H_{o,s}$ y $H_{u,s}$ anteriormente definidas en (2.2) junto con las matrices $H_{o\tau}$ y $H_{s\tau}$ definidas en (3.13) están relacionadas por

$$H_{o,s}^\perp = H_{o\tau}^\perp P_m \quad (3.21)$$

$$H_{o,s}^\perp H_{u,s} = H_{o\tau}^\perp H_{s\tau} P_{k_u} \quad (3.22)$$

donde

$$P_j = \begin{bmatrix} I_{j \times j} & O & \cdots & O \\ C_1^0 I_{j \times j} & C_1^1 \tau I_{j \times j} & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ C_s^0 I_{j \times j} & C_s^1 \tau I_{j \times j} & \cdots & C_s^s \tau^s I_{j \times j} \end{bmatrix} \quad (3.23)$$

con $j = m, k_u$.

Para demostrarlo se utiliza una de las principales propiedades del complemento ortogonal, esto es $H_{o\tau}^\perp H_{o\tau} = 0$, por consiguiente $H_{o\tau}^\perp P_m H_{o,s} = 0$ debido a la igualdad (3.20). Entonces la ecuación (3.21) queda probada.

Para la ecuación $H_{s\tau} P_{k_u} = P_m H_{u,s}$ puede verse que añadiendo $H_{o\tau}^\perp$ en ambos lados, (3.22) se satisface, $H_{o\tau}^\perp H_{s\tau} P_{k_u} = H_{o\tau}^\perp P_m H_{u,s} = H_{o\tau}^\perp H_{u,s}$.

Es decir, de acuerdo a lo anterior si $H_{o\tau}^\perp$ y $H_{o\tau}^\perp H_{s\tau}$ son obtenidos, entonces $H_{o,s}^\perp$ y $H_{o,s}^\perp H_{u,s}$ pueden obtenerse también respectivamente a partir de (3.21) y (3.22).

Por lo tanto, un vector de paridad ν_s satisfaciendo (2.4) y a su vez $\rho_s = \nu_s H_{u,s}$ pueden seleccionarse de la siguiente manera

$$\nu_s = \alpha H_{o,s}^\perp, \quad \rho_s = \alpha H_{o,s}^\perp H_{u,s} \quad (3.24)$$

$\alpha \neq 0$ es un vector fila de dimensiones compatibles. De manera que para la implementación se sustituye (3.19), (3.21) y (3.22) en (3.24) resultando

$$\begin{aligned} \nu_s &= \alpha U'_{12} P_m \\ \rho_s &= -\alpha U'_{22} P_{k_u} \end{aligned} \quad (3.25)$$

Una forma de obtener el vector α es mediante una factorización QR [18]. Factorizando U'_{12} queda

$$U'_{12} = U_Q U_R$$

donde U_Q es una matriz unitaria y el bloque triangular superior derecho de U_R es cero. Entonces α puede ser seleccionado como la primera fila de U'_Q , de esta manera αU_{12} será la primera fila de U_R y por lo tanto tendrá la mayor cantidad de ceros debido a la relación $U_R = U'_Q U_{12}$.

3.2.4 GENERACIÓN DE RESIDUO A PARTIR DE DATOS BASADA EN OBSERVADOR

Se sabe que el espacio de paridad guarda una estrecha relación con el observador de diagnóstico [7]. Se considera un generador de residuos basado en observador

$$\begin{aligned} \dot{z}(t) &= Gz(t) + Ju(t) + Ly(t) \\ r(t) &= \omega z(t) + \rho u(t) + \nu y(t) \end{aligned} \quad (3.26)$$

donde las matrices

$$G = \begin{bmatrix} 0 & \cdots & 0 & g_1 \\ 1 & \cdots & 0 & g_2 \\ \vdots & & & \vdots \\ 0 & \cdots & 1 & g_s \end{bmatrix}; \quad J = \begin{bmatrix} \rho_{s,0} \\ \rho_{s,1} \\ \vdots \\ \rho_{s,s-1} \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_s \end{bmatrix} \rho_s; \quad L = - \begin{bmatrix} \nu_{s,0} \\ \nu_{s,1} \\ \vdots \\ \nu_{s,s-1} \end{bmatrix} - \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_s \end{bmatrix} \nu_{s,s}$$

$$w = \begin{bmatrix} 0 & \cdots & 0 & -1 \end{bmatrix}; \quad p = \rho_{s,s}; \quad v = \nu_{s,s} \quad (3.27)$$

junto con la matriz de transformación

$$T = \begin{bmatrix} \nu_{s,1} & \cdots & \nu_{s,s-1} & \nu_{s,s} \\ \nu_{s,2} & \cdots & \nu_{s,s} & 0 \\ \vdots & & & \vdots \\ \nu_{s,s} & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s-1} \end{bmatrix}$$

satisfacen las ecuaciones de Luenberger cuando G es estable

$$TA - GT = LC; \quad vC + wT = 0$$

$$TB - LD = J; \quad p + vD = 0 \quad (3.28)$$

La dinámica del residuo está descrita por las ecuaciones

$$\dot{e}(t) = Ge(t) + (LF_f - TE_f)f(t)$$

$$r(t) = we(t) + vF_f f(t) \quad (3.29)$$

donde $e(t) = z(t) - Tx(t)$. Se observa que la señal del residuo no depende de la entrada de control, por lo tanto, si existe una falla será diferente de cero y por lo tanto será detectada.

Los valores g_1, g_2, \dots y g_s son constantes de libre diseño que garantizan la estabilidad de G . Es sencillo ver que para la construcción de las matrices (3.27) que satisfacen (3.28), solo se necesita identificar ν_s y ρ_s , además de las constantes g_1, g_2, \dots y g_s . Entonces, la información de ν_s y ρ_s es suficiente para la construcción de un generador de residuos basado en observador (3.26).

3.2.4.1 ALGORITMO 1 (DISEÑO DE UN OBSERVADOR CONTINUO PARA DETECCIÓN DE FALLAS)

1. Establecer valores de $\tau > 0$, N , g_1, \dots, g_s y s .
2. Filtrar datos de entrada y salida $u(t)$ y $y(t)$ para obtener $u^{(i)}(t)$, $y^{(i)}(t)$, $i = 1, \dots, s$.
3. Tomar muestras de $u(t_k)$, $y(t_k)$, $u^{(i)}(t_k)$, $y^{(i)}(t_k)$, $k = 1, \dots, N$.
4. Construir W_N con Y_N y U_N .
5. Realizar descomposición en valores singulares (SVD) de la matriz W_N y obtener U_{12} y U_{22} .
6. Calcular matrices P_m y P_{k_u} de acuerdo a (3.23).
7. Calcular un vector $\alpha \neq 0$ y obtener vectores ν_s y ρ_s de acuerdo a (3.25).
8. Seleccionar g_1, g_2, \dots, g_s garantizando la estabilidad de G .
9. Formar G , J , L , w , p y v de acuerdo a (3.27).
10. Construir el generador de residuos (3.26).

3.2.5 EJEMPLO CON ALGORITMO 1

3.2.5.1 SISTEMA DE 3 TANQUES

Una vez que el algoritmo fue programado en la plataforma MATLAB, se procede a demostrar la eficiencia del mismo. El siguiente ejemplo trata de un modelo ampliamente utilizado para aplicaciones de control, el sistema de tres tanques [2], el cual consta de 2 bombas que llenan los tanques de los extremos respectivamente.

Éstos se encuentran interconectados entre sí, sólo el tanque 2 tiene salida de líquido como se muestra en la Fig. 3.2.

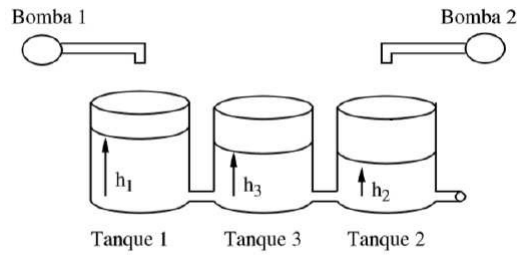


Figura 3.2: Sistema de 3 tanques

Como entradas se tienen los flujos de cada bomba mientras que las salidas son los niveles de cada tanque. Se considera el modelo linealizado de los 3 tanques.

$$\dot{x} = \begin{bmatrix} -0.0085 & 0 & 0.0085 \\ 0 & -0.0195 & 0.0084 \\ 0.0085 & 0.0084 & -0.0169 \end{bmatrix} x + \begin{bmatrix} 0.0065 & 0 \\ 0 & 0.0065 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}$$

$$y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x; \quad x = y = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}$$

donde h_1 , h_2 y h_3 son los niveles de cada tanque, Q_1 y Q_2 son las entradas de flujo provenientes de cada bomba.

3.2.5.2 RESULTADOS MEDIANTE SIMULACIÓN

Se simuló el sistema de los 3 tanques con el objetivo de tomar las mediciones de entrada y salida necesarias para el algoritmo.

Para este ejemplo se utilizaron 4 filtros ya que el orden del sistema es 3. El tiempo de simulación fue de 10,000s con un tiempo de muestreo de 1s. La variable

de tiempo τ se seleccionó aleatoriamente de 0.5s debido a que la cantidad de datos a manejar es grande, por lo que la diferencia entre $\mathcal{L}^{-1}(\omega^i U(s))$ y $\mathcal{L}^{-1}(\omega^{i+1} U(s))$ es más pequeña.

Las ecuaciones del generador de residuos resultante utilizando el algoritmo son las siguientes:

$$G = \begin{bmatrix} 0 & 0 & 0 & -1.6378 \\ 1 & 0 & 0 & -6.1819 \\ 0 & 1 & 0 & -8.2852 \\ 0 & 0 & 1 & -4.7609 \end{bmatrix}; \quad J = \begin{bmatrix} 0.0674 & -0.0663 \\ 0.0342 & -0.0340 \\ 0.0045 & -0.0047 \\ 0.0085 & -0.0084 \end{bmatrix}$$

$$L = \begin{bmatrix} 0.0089 & -0.0113 & -0.0356 \\ -0.0641 & -0.0482 & 1.1611 \\ -0.0545 & -0.0242 & 0.7525 \\ -0.0187 & -0.0040 & 0.1494 \end{bmatrix}$$

$$w = \begin{bmatrix} 0 & 0 & 0 & -1 \end{bmatrix}; \quad p = \begin{bmatrix} -0.0025 & 0.0024 \end{bmatrix} \quad v = \begin{bmatrix} 0.0133 & 0.0007 & -0.0368 \end{bmatrix}$$

Los resultados de simulación son los siguientes:

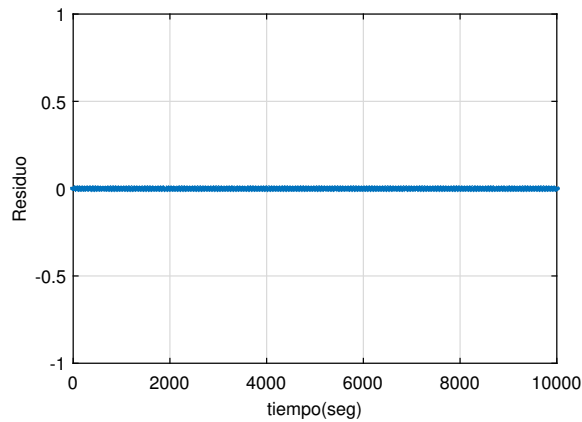


Figura 3.3: Caso sin falla

Se observa en la Fig. 3.3 que cuando no existen fallas presentes, este se mantiene en un comportamiento oscilatorio que prácticamente es cero.

Para el siguiente caso se presenta una falla en el sensor 2 al segundo 3000 Fig. 3.4.

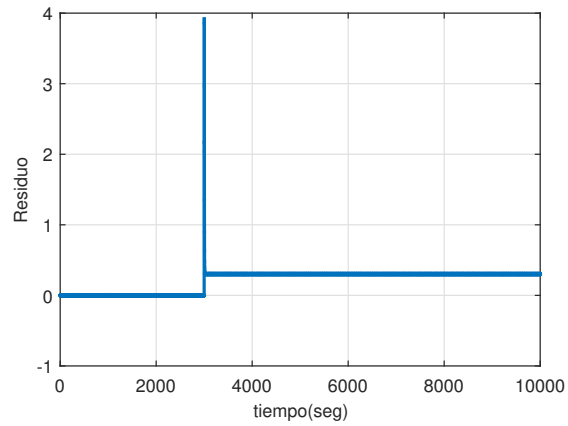


Figura 3.4: Caso con falla

Nuevamente se obtiene la gráfica del residuo en donde se aprecia que el residuo es diferente de cero, por lo que la falla fue correctamente detectada.

CAPÍTULO 4

OBSERVADORES CONTINUOS PARA AISLAMIENTO DE FALLAS

4.1 DESCRIPCIÓN DEL CAPÍTULO

En el diagnóstico de fallas, los métodos basados en datos han ganado gran popularidad en los últimos años. En ocasiones se vuelve casi imposible diseñar un sistema de diagnóstico de fallas para los métodos basados en modelos debido al incremento drástico en la complejidad de algunos sistemas. A diferencia de ellos, los enfoques basados en datos históricos del proceso ofrecen herramientas para extraer información útil en el diseño de un sistema de monitoreo.

Anteriormente vimos que para un sistema en tiempo discreto, es posible diseñar un algoritmo para aislar fallas utilizando las medidas del proceso disponibles, sin embargo, en sistemas en tiempo continuo, sigue siendo un problema sin resolver.

En este capítulo se presenta un enfoque de solución para diseñar un algoritmo capaz de detectar y aislar fallas en sistemas lineales en tiempo continuo, se busca demostrar que es posible extender los resultados obtenidos previamente en sistemas en tiempo discreto a los sistemas en tiempo continuo, bajo el supuesto de que diferentes conjuntos de mediciones se encuentran disponibles para la primer etapa de

diseño. Se muestra la justificación técnica de los resultados, así como experimentos para mostrar el funcionamiento del algoritmo.

4.2 DISEÑO DE UN GRUPO DE OBSERVADORES PARA AISLAMIENTO DE FALLAS

Normalmente cuando el objetivo de diagnóstico es detección, estamos hablando de un generador de residuos (en nuestro caso basado en observador) diseñado para detectar cualquier señal que rebase los límites de tolerancia permitidos por el sistema en su forma nominal (Fig. (4.1)), es decir, que para los enfoques basados en datos, el generador de residuos se diseña considerando únicamente datos nominales.

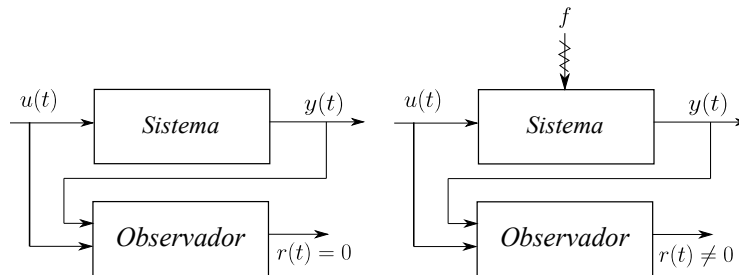


Figura 4.1: Esquema para de detección de fallas

Usando el mismo concepto de diseño, con la diferencia de que ahora datos con la información de ciertas fallas de interés son utilizados para la primer etapa de diseño, es posible construir un generador de residuos que no detecte 1 falla, es decir, que sea insensible a esa falla, pero sensible a cualquier otra falla. La idea se muestra en la Fig. (4.2) donde se tienen 2 fallas que se quieren desacoplar, entonces el generador de residuos se diseña insensible a la falla f_1 pero sensible a la falla f_2 .

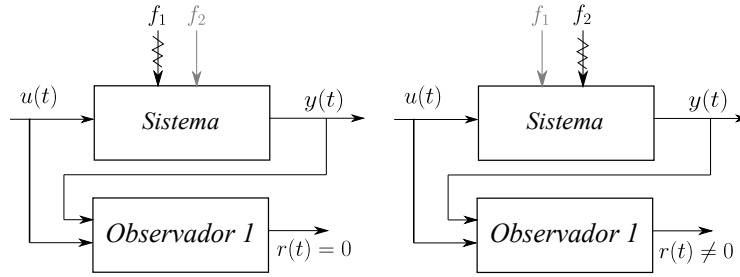


Figura 4.2: Esquema para aislamiento de fallas

De esta manera queda más clara la idea, los generadores de residuos se diseñarán insensibles a las fallas de interés, con la condición de que la información en datos de esas fallas se encuentre disponible para ello. El resultado será que tendremos un grupo de generadores de residuos estructurados de tal manera que cada uno tenga un comportamiento diferente ante cada falla, es decir, cuando ocurra una falla, el generador de residuo que fue diseñado con los datos de esa falla, dará como salida un residuo igual a cero, mientras que los demás generadores de residuos se verán afectados y sus residuos serán diferentes a cero. A esto se le conoce en la literatura como el *Esquema de Observadores Generalizados* (GOS) [1] y se muestra en la Fig. (4.3).

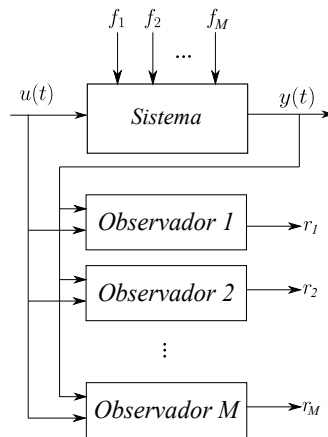


Figura 4.3: Esquema de observadores generalizados

Este arreglo nos ayudará para la localización de fallas. Cabe mencionar que solo fallas que estén desacopladas podrán ser localizadas, esto es, que no se encuentren

afectando el mismo canal.

4.3 AISLAMIENTO DE FALLAS EN SISTEMAS LINEALES CONTINUOS

4.3.1 AISLAMIENTO DE FALLAS

Mediante el aislamiento de fallas, nos es posible conocer en donde ha ocurrido una falla, ya sea en sensores, actuadores, componentes, etc. debido a que se elimina la dependencia de una falla con respecto a otra; esto será posible siempre y cuando las fallas que se quieran aislar, se encuentren desacopladas.

4.3.2 PROPUESTA DE SOLUCIÓN

La propuesta involucra el hecho de que se encuentran suficientes mediciones disponibles para la parte de diseño. Al trabajar únicamente con datos entrada-salida son pocas las opciones que se tienen para obtener información sobre las fallas, es por eso que suponemos que también se encuentran disponibles en los datos.

A pesar de que parece redundante el hecho de utilizar tantos observadores, cada uno contiene información específica sobre la falla que se quiere aislar, de esta manera se obtienen las características de robustez deseadas. Se propone jugar con los datos tantas veces como fallas existan más uno. Es evidente que el esfuerzo computacional es un precio a pagar para el diseño, sin embargo justo, por la cantidad de datos que se debe manejar.

Siguiendo el *Esquema de observadores generalizados* y utilizando únicamente mediciones entrada-salida afectadas por una falla, es posible el diseño de un genera-

dor de residuos insensible a esa falla pero sensible a otras fallas.

4.3.3 JUSTIFICACIÓN MATEMÁTICA

Se considera un sistema continuo lineal e invariante en el tiempo con fallas aditivas:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + E_f f(t) \\ y(t) &= Cx(t) + Du(t) + F_f f(t) \end{aligned} \quad (4.1)$$

donde $x \in \mathfrak{R}^n$ es el vector de estado, $u \in \mathfrak{R}^{k_u}$ el vector de entradas de control, $y \in \mathfrak{R}^m$ el vector de salidas medidas, $f \in \mathfrak{R}^{k_f}$ el vector de fallas, $A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times k_u}$, $C \in \mathfrak{R}^{m \times n}$, $D \in \mathfrak{R}^{m \times k_u}$, $E_f \in \mathfrak{R}^{n \times k_f}$, $F_f \in \mathfrak{R}^{m \times k_f}$ son matrices constantes conocidas y de dimensiones compatibles. Se asume además que el sistema (4.1) es controlable y observable.

El modelo de datos entrada-salida puede ser descrito por:

$$Y_N = H_{o\tau} X_N + H_{s\tau} U_N + H_{f\tau} F_N \quad (4.2)$$

Tomando datos de entrada-salida nominales y con falla, se construye el generador de residuos con ν_s tal que cancele el efecto de las fallas presentes:

$$r(t) = \nu_s(Y_N - H_{s\tau} U_N) = \nu_s(H_{o\tau} X_N + H_{f\tau} F_N) = 0 \quad (4.3)$$

donde

$$H_{f\tau} = \begin{bmatrix} F & 0 & \cdots & 0 \\ \tau CE & F & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \tau C(\tau A + I)^{s-1} E & \cdots & \tau CE & F \end{bmatrix}$$

y las matrices $H_{o\tau}$ y $H_{s\tau}$ fueron definidas previamente en (3.13). La primera parte de (4.3) es llamada residuo de implementación y la segunda parte, residuo de análisis. Por lo tanto, si el residuo es diseñado con los datos de la falla que se quiere aislar, ésta

será desacoplada de dicho residuo. La ecuación (4.3) también puede representarse como

$$r(t) = \nu_s \left(\begin{bmatrix} H_{o\tau} & H_{s\tau} \\ 0 & I \end{bmatrix} \begin{bmatrix} X_N \\ U_N \end{bmatrix} + \begin{bmatrix} H_{f\tau} F_N \\ 0 \end{bmatrix} \right) \quad (4.4)$$

Utilizando el residuo de análisis, se puede obtener una ecuación de residuo para las fallas f_1 a f_M .

$$r_1(f_2, \dots, f_M) = \nu_{s_1}(H_{o\tau}X_N + H_{f\tau}F_{1N}) = 0$$

$$r_2(f_1, f_3, \dots, f_M) = \nu_{s_2}(H_{o\tau}X_N + H_{f\tau}F_{2N}) = 0$$

⋮

$$r_M(f_1, f_2, \dots, f_{M-1}) = \nu_{s_M}(H_{o\tau}X_N + H_{f\tau}F_{M-1N}) = 0$$

Una vez obtenidos los residuos se generan las firmas características de cada falla (matriz de incidencia de fallas). Se muestra en la tabla (4.1) un ejemplo de M residuos con M fallas, con desacoplamiento de ciertas fallas de interés. donde “ \emptyset ”

	Falla 1 (f_1)	Falla 2 (f_2)	⋯	Falla M (f_M)
Residuo 1 (r_1)	\emptyset	✓		✓
Residuo 2 (r_2)	✓	\emptyset		✓
⋮			⋯	
Residuo M (r_M)	✓	✓		\emptyset

Tabla 4.1: Matriz de incidencia de fallas

representa que el residuo fue diseñado insensible a la falla correspondiente, mientras que “✓” representa sensibilidad a la falla. Entonces el residuo 1 (r_1) será insensible a la falla 1 pero sensible a las demás fallas, y así sucesivamente hasta la falla M.

Entonces es posible diseñar un banco de generadores de residuo insensibles a las fallas de interés utilizando únicamente datos entrada-salida basados en el razonamiento previo. Se requerirán los datos de todas las fallas que se quieran ser

analizadas. El cálculo de ν_s implicará la cancelación de cualquier falla que esté presente en los datos utilizados para el diseño, recordando que será posible cuando las fallas no esten acopladas.

4.3.3.1 ALGORITMO 2 (DISEÑO DE UN GENERADOR DE RESIDUO EN TIEMPO CONTINUO INSENSIBLE A LA FALLA M-ÉSIMA)

1. *Considerar el conjunto de datos con la información de la falla M-ésima.*
2. *Filtrar los datos de entrada y salida como se muestra en la Fig. 3.1.*
3. *Formar las matrices de Hankel U_N y Y_N .*
4. *Encontrar ν_s y ρ_s .*
5. *Construir el generador de residuos basado en observador siguiendo (3.26).*

4.4 EJEMPLO CON ALGORITMO 2

Para probar la fiabilidad del algoritmo propuesto, se probó en el modelo dq0 del motor síncrono de imanes permanentes (PMSM). El modelo no representa el tipo de sistemas para los que fue propuesto el algoritmo, ya que es un sistema no lineal, sin embargo, se demuestra que cuando el efecto no lineal de un punto de operación cambia, éste puede ser representado como una entrada adicional; además, cuando se trata de diferentes puntos de operación, el efecto no lineal afecta siempre el mismo canal de entrada, y usando la característica de desacoplo del algoritmo, es posible cancelar el efecto no lineal. Algunos resultados recientes de diagnóstico de fallas basado en observador para sistemas no lineales pueden encontrarse en [19].

4.4.1 MOTOR SÍNCRONO DE IMANES PERMANENTES (PMSM)

Se considera el modelo de un motor síncrono de imanes permanentes. Aunque no se requiere información del modelo explícito en ecuaciones diferenciales debido a que se trata únicamente con los datos, se revisarán las ecuaciones del modelo para justificar algunos aspectos del algoritmo. Se presenta un esquemático de un motor síncrono de imanes permanentes de 2 polos:

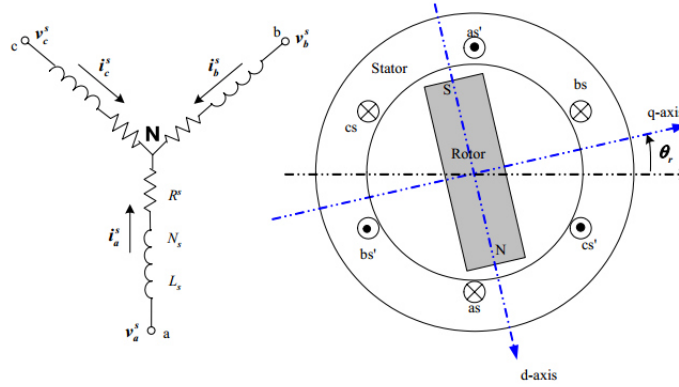


Figura 4.4: Modelo esquemático de un PMSM de 2 polos

El modelo en dq0 fue obtenido después de usar una transformación de park al modelo abc [20]. El modelo en espacio de estado es dado por:

$$\dot{x} = \begin{bmatrix} -\frac{R}{L_{dq}} & x_3 & 0 \\ -x_3 & -\frac{R}{L_{dq}} & -\frac{\Psi_m}{L_{dq}} \\ 0 & \frac{3P^2}{8J}\Psi_m & 0 \end{bmatrix} x + \begin{bmatrix} \frac{1}{L_{dq}} & 0 \\ 0 & \frac{1}{L_{dq}} \end{bmatrix} \begin{bmatrix} V_d \\ V_q \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -\frac{P\tau_L}{2J} \end{bmatrix} \quad (4.5)$$

donde $x \triangleq [i_d \ i_q \ \omega]^T$, i_d y i_q son las corrientes en secuencia d y q, ω es la velocidad angular del rotor, V_d y V_q son los voltajes de entrada, τ_L representa la carga de par, R es la resistencia, P es el número de polos, J es la inercia del motor, L_{dq} es la inductancia de q y d, secuencias que se asumen iguales por simplicidad.

Para poder visualizar el efecto de la no linealidad en el sistema lineal alrededor de un punto de equilibrio, se presenta el modelo linealizado de (4.5):

$$x_\ell = \bar{A}x_\ell + \bar{B}u + Q(x) \quad (4.6)$$

donde

$$Q(x) = \begin{bmatrix} x_{3OP}x_2 + x_{2OP}x_3 \\ -x_{3OP}x_1 - x_{1OP}x_3 \\ 0 \end{bmatrix}$$

y x_{1OP} , x_{2OP} , x_{3OP} corresponden a los valores de los estados en un punto de operación. Por lo tanto, se pueden desacoplar dos canales, sin embargo, para el modelo utilizado del motor síncrono, los valores de x_{1OP} se encuentran muy cercanos a cero, por lo que puede ser despreciado, de esta manera solo se requiere desacoplar el canal x_{2OP} .

El sistema fue simulado en la plataforma Simulink para la obtención de los datos.

4.4.2 EXPERIMENTOS REALIZADOS

Para el diseño de un banco de residuos se debe de tomar en cuenta la cantidad de fallas que se quieren analizar y el número de puntos puntos de operación que se van a utilizar. En este ejemplo, se seleccionaron 3 puntos de operación: 100 rpm, 80 rpm y 60 rpm. Los datos entrada-salida con falla correspondientes a cada punto de operación de obtienen por separado.

Se considera entonces el diseño de dos generadores de residuo, cada uno insensible a la otra falla. Se construye un conjunto de datos con la información recolectada de cada punto de operación. Para nuestro objetivo, los datos son concatenados considerando los diferentes puntos de operación, sin importar el orden en que son acomodados.

Para este ejemplo se utilizaron 4 filtros debido a que el orden del sistema es 3. El tiempo de simulación fue de 60s con un tiempo de muestreo de 0.01s. La variable de tiempo τ se seleccionó en este caso de 1.6s. La falla en el actuador es causada en la fuente voltaje, por lo que ambos voltajes de entrada se verán afectados ocasionando un mal suministro. La otra falla ocurre en el sensor de la salida, en este caso, de la velocidad angular ω .

La matriz de incidencia de fallas en la tabla (4.3) muestra los resultados de los residuos diseñados para la prueba.

	Falla 1 (f_1)	Falla 2 (f_2)
Residuo 1 (r_1)	\emptyset	\checkmark
Residuo 2 (r_2)	\checkmark	\emptyset

Tabla 4.2: Matriz de incidencia de fallas ejemplo 2

donde f_1 representa la falla en el actuador mientras que f_2 corresponde a la falla en el sensor.

Se toman en cuenta 3 escenarios de prueba diferentes para la construcción de un banco de residuos, el objetivo es mostrar la fiabilidad del algoritmo propuesto para aislamiento de fallas.

Escenarios	Descripción
Caso 1	Datos nominales
Caso 2	Datos con falla en sensor
Caso 3	Datos con falla en actuador

Tabla 4.3: Escenarios de prueba considerados

4.4.3 RESULTADOS DE SIMULACIÓN

Para el diseño de los dos generadores de residuo insensibles, se utilizan tanto el algoritmo 1 como el algoritmo 2. Las ecuaciones de los generadores de residuos resultantes utilizando el algoritmo son las siguientes:

Generador de residuos 1.

$$G_1 = \begin{bmatrix} 0 & 0 & 0 & -19.0568 \\ 1 & 0 & 0 & -38.4638 \\ 0 & 1 & 0 & -27.9577 \\ 0 & 0 & 1 & -8.7478 \end{bmatrix}; J_1 = \begin{bmatrix} 6.7169 & -25.4125 \\ 13.2264 & -50.0390 \\ 8.8311 & -33.4079 \\ 2.0645 & -7.8089 \end{bmatrix}$$

$$L_1 = \begin{bmatrix} 7.5133 & -21.7415 & 40.6773 \\ 14.7941 & -42.8110 & 80.0984 \\ 9.8768 & -28.5841 & 53.4817 \\ 2.3087 & -6.6825 & 12.5037 \end{bmatrix}$$

$$w_1 = \begin{bmatrix} 0 & 0 & 0 & -1 \end{bmatrix}; p_1 = \begin{bmatrix} 0.3672 & -1.3892 \end{bmatrix} \quad v_1 = \begin{bmatrix} 0.4107 & -1.1885 & 2.2235 \end{bmatrix}$$

Generador de residuos 2.

$$G_2 = \begin{bmatrix} 0 & 0 & 0 & -7.1040 \\ 1 & 0 & 0 & -18.1693 \\ 0 & 1 & 0 & -16.8569 \\ 0 & 0 & 1 & -6.7745 \end{bmatrix}; J_2 = \begin{bmatrix} -5.1509 & -0.5570 \\ -12.7922 & -1.3873 \\ -10.4435 & -1.2465 \\ -2.8979 & -0.3996 \end{bmatrix}$$

$$L_2 = \begin{bmatrix} 0.0231 & -17.0342 & 1.7159 \\ 0.0580 & -42.2541 & 4.2872 \\ 0.0626 & -34.7913 & 4.2154 \\ 0.0244 & -9.8032 & 1.5207 \end{bmatrix}$$

$$w_2 = \begin{bmatrix} 0 & 0 & 0 & -1 \end{bmatrix}; p_2 = \begin{bmatrix} -0.8194 & -0.0854 \end{bmatrix} \quad v_2 = \begin{bmatrix} 0.0033 & -2.6885 & 0.2526 \end{bmatrix}$$

Los resultados de implementación considerando los escenarios antes mencionados son los siguientes.

Caso 1.

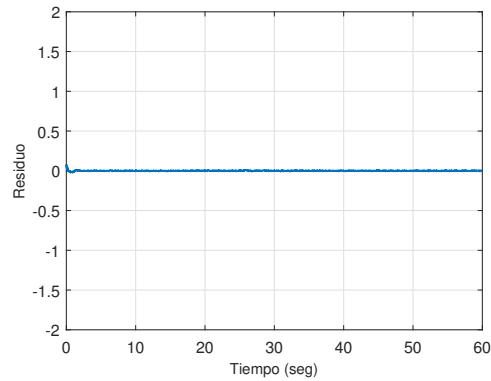


Figura 4.5: Residuo 1 en ausencia de falla

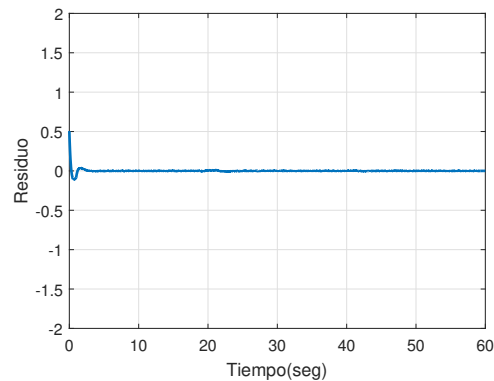


Figura 4.6: Residuo 2 en ausencia de falla

Se observa que en ambas gráficas, el residuo permanece cerca de cero, este es el comportamiento deseado para el caso sin falla, ya que el error de estimación de la salida es casi cero.

Caso 2.

Para ese caso se añade una falla en el sensor, se puede ver en las siguientes gráficas que el residuo 2 es insensible a esa falla mientras que en el residuo 1 puede

ser detectada fácilmente.

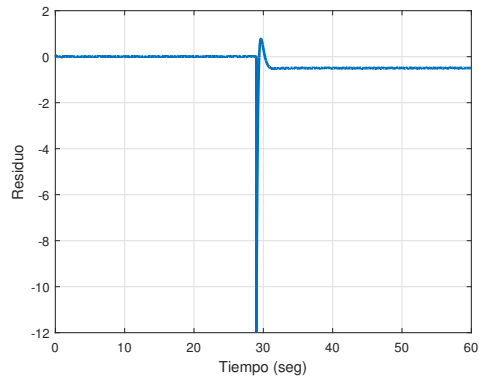


Figura 4.7: Residuo 1 con falla en sensor

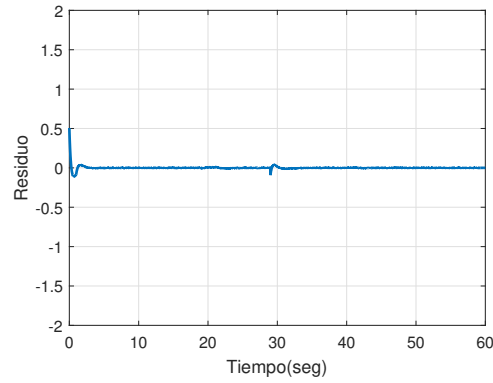


Figura 4.8: Residuo 2 con falla en sensor

Caso 3.

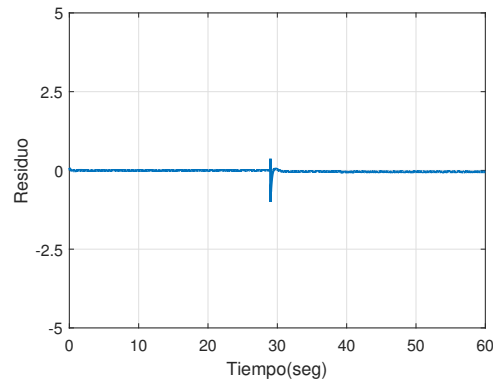


Figura 4.9: Residuo 1 con falla en actuador

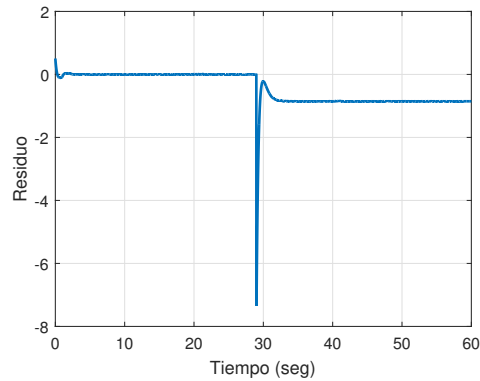


Figura 4.10: Residuo 2 con falla en actuador

En este caso se puede observar una respuesta similar a la del caso 2 pero ahora cambiando de residuo. Por lo tanto, la falla en el actuador puede detectarse observando las diferencias entre ambos residuos. El efecto que ocasiona la falla en el actuador en el residuo 1 es pequeña en comparación con el residuo 2, además de que no permanece en todo tiempo. De hecho, el pequeño efecto de la falla en el actuador en el residuo 1 y el pequeño efecto de la falla en el sensor en el residuo 2, se deben a la no linealidad del sistema que esta siendo desacoplada.

Los 3 casos demuestran la eficiencia y el funcionamiento del algoritmo propuesto.

CAPÍTULO 5

CONCLUSIONES Y TRABAJO A FUTURO

Se consideró un enfoque de aislamiento de fallas en sistemas lineales continuos a partir de datos. Se comprobó mediante ejemplos prácticos que el aislamiento puede alcanzarse siguiendo la idea para el caso discreto, siendo las principales diferencias el método de obtención de las matrices de datos y la obtención del espacio nulo izquierdo. La condición para la construcción del aniquilador izquierdo requiere que la información de las fallas esté presente en los datos, con magnitud suficientemente grande para su manifestación.

Se demuestra que la teoría de identificación por subespacios tiene sustento en la idea propuesta. Debido a esto, es importante mencionar que las señales de entrada deben ser obtenidas con la característica de excitación persistente para la correcta identificación de las matrices del observador.

Aunque el enfoque incluye un proceso implícito de muestreo, el hecho de trabajar con sistemas continuos, implica el análisis de un mayor número de muestras para el mismo sistema en su versión discreta, lo que incrementa la localizabilidad de las fallas.

El algoritmo cuenta con muchas posibilidades de ajuste, la variable de tiempo del filtro, y los valores propios de la matriz G del observador son la prueba de ello. Es importante recordar que el orden del sistema se verá afectado por la cantidad de

filtros que se utilicen, por lo que, entre más filtros requiera el diseño, mayor será la carga computacional. Trabajo futuro se requiere para identificar con mayor eficiencia el valor del parámetro τ del filtro, y los valores propios de la matriz G .

Las pruebas realizadas al motor síncrono de imanes permanentes demuestran que el algoritmo puede ser aplicado eficientemente a sistemas no lineales alrededor de un punto de equilibrio, es decir, a sistemas linealizados, ya que el efecto no lineal puede ser cancelado por la característica de aislamiento. Gracias a los resultados de este tema de tesis, se tiene el artículo “*Enfoque en Tiempo Continuo para Aislamiento de de Fallas a partir de Datos Muestreados*” aceptado para el Congreso Nacional de Control Automático (AMCA) 2018.

Fallas con magnitudes grandes y pequeñas pueden ser fácilmente detectadas utilizando el enfoque. Cabe mencionar que la magnitud de una falla puede ser obtenida por medio de la inversa del generador de residuos [21], utilizando la transferencia del residuo de implementación como ganancia, siempre y cuando se conozca el canal de entrada de la falla que se desea analizar.

Los métodos de diagnóstico de fallas que utilizan mediciones, continúan en desarrollo. Como trabajo futuro se pretende considerar un umbral en el residuo para evitar falsas alarmas, así como la implementación en un sistema físico.

BIBLIOGRAFÍA

- [1] P. M. Frank. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results. *Automatica*, 26(3):459–474, 1990.
- [2] S. X. Ding. *Model-based fault diagnosis techniques: design schemes, algorithms, and tools*. Springer Science & Business Media, 2008.
- [3] D. G. Luenberger. Observing the state of a linear system. *IEEE transactions on military electronics*, 8(2):74–80, 1964.
- [4] R. V. Beard. *Failure accomodation in linear systems through self-reorganization*. PhD thesis, Massachusetts Institute of Technology, 1971.
- [5] J. Chen and R. J. Patton. *Robust model-based fault diagnosis for dynamic systems*, volume 3. Springer Science & Business Media, 2012.
- [6] P. Zhang and S. X. Ding. A model-free approach to fault detection of continuous-time systems based on time domain data. *International Journal of Automation and Computing*, 4(2):189–194, 2007.
- [7] S. X. Ding. *Data-driven design of fault diagnosis and fault-tolerant control systems*. Springer, 2014.
- [8] E. Alcorta-Garcia. Detectando fallas mediante redundancia analítica. *Ingenierías*, 4(11):43, 2001.
- [9] A. Varga. Solving fault diagnosis problems. *Studies in Systems*, 2017.

-
- [10] P. Amann, E. Alcorta Garcia, B. Koeppen-Seliger, and P.M. Frank. Knowledge- and data-based models for fault diagnosis. *Systems Analysis Modelling Simulation*, 35(1):25–44, 1999.
- [11] H. Lasi, P. Fettke, H. Kemper, T. Feld, and M. Hoffmann. Industry 4.0. *Business & Information Systems Engineering*, 6(4):239–242, 2014.
- [12] I. E. Leal Leal A. Banda Urbina, E. Alcorta-Garcia. Aislamiento de fallas en sistemas lineales discretos a partir de mediciones. *AMCA*, pages 1–7, 2016.
- [13] R. Isermann. *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer Science & Business Media, 2006.
- [14] C. Verde, S. Gentil, and R. Morales-Menéndez. *Monitoreo y diagnóstico automático de fallas en sistemas dinámicos*. Editorial Trillas, 2013.
- [15] E. Y. Chow and A. S. Willsky. Analytical redundancy and the design of robust failure detection systems. *Automatic Control, IEEE Transactions on*, 29(7):603–614, 1984.
- [16] P. Van Overschee and B. L. De Moor. *Subspace identification for linear systems: Theory implementation applications*. Springer Science & Business Media, 2012.
- [17] L. Ljung. *System identification: theory for the user*. Englewood Cliffs, 1987.
- [18] P. Zhang. Design of observer-based fault detection systems for continuous-time systems from frequency domain data. In *Proceedings of the 16th IFAC World Congress*. Prague Czech Republic, 2005.
- [19] J. Zhang, A. K. Swain, and S. K. Nguang. *Robust Observer-Based Fault Diagnosis for Nonlinear Systems Using MATLAB®*. Springer, 2016.
- [20] L. Liu. *Robust Fault Detection and Diagnosis for Permanent Magnet Synchronous Motors*. PhD thesis, Florida State University, 2006.
- [21] E. Alcorta Garcia and J.C. Ruiz Garcia. Estimación de fallas: un enfoque basado en inversión del generador de residuos. *AMCA*, 2015.