



Cinvestav

**CENTRO DE INVESTIGACIÓN Y
ESTUDIOS AVANZADOS DEL INSTITUTO
POLITÉCNICO NACIONAL**

Unidad Zacatenco

DEPARTAMENTO DE CONTROL AUTOMÁTICO

**Aplicaciones del algoritmo PSO en problemas de
identificación paramétrica y sintonización**

Tesis que presenta

M. en T.A. Ricardo Alan Cortez Vega

Para obtener el grado de

Doctor en Ciencias

en la Especialidad de

Control Automático

Director de la Tesis

Dr. Rubén Alejandro Garrido Moctezuma

Ciudad de México

Marzo de 2022

Índice general

1	Introducción	7
1.1	Algoritmos metaheurísticos	8
1.1.1	Problema de optimización numérica Mono-Objetivo	9
1.1.2	Problemas de optimización numérica Multi-Objetivo (MO)	9
1.2	Identificación de parámetros	11
1.2.1	Identificación en señales con baja riqueza espectral	12
1.2.2	Identificación en presencia de ruido de medición	13
1.3	Sintonización de observadores de estado extendido	14
1.4	Objetivo General	15
1.5	Objetivos Específicos	16
1.6	Estructura de la tesis	16
1.7	Publicaciones	16
1.7.1	Artículo de revista	16
1.7.2	Artículos de congreso internacional	17
1.7.3	Artículos de congreso nacional	17
2	Algoritmo PSO como identificador de parámetros en sistemas dinámicos	18
2.1	Algoritmo PSO clásico	18
2.2	Variantes del algoritmo PSO	21
2.2.1	Algoritmo PSO con Ponderación de Velocidad Variable (PSO-PVV)	21
2.2.2	Algoritmo PSO con Factor de Constricción Variable (PSO-FCV)	22
2.2.3	Algoritmo PSO de Orden Fraccionario Regulado (PSO-OFR)	22
2.3	Identificación de parámetros como un problema de optimización	24
2.4	Implementación de la identificación de parámetros mediante el algoritmo PSO	25
2.4.1	Adquisición de datos	25
2.4.2	Optimización numérica	26
2.4.3	Validación de los parámetros estimados mediante la realización de análisis estadísticos	27
2.4.4	Validación de los parámetros identificados mediante la implementación de una ley de control	30

3	Efecto de la Riqueza Espectral en la identificación de parámetros empleando el algoritmo PSO	33
3.1	Condición de Excitación Persistente (EP) en los algoritmos clásicos de identificación de parámetros	33
3.2	Medición de la Riqueza Espectral de las señales de excitación en la identificación de parámetros	35
3.3	Algoritmo PSO-RE enfocado a la identificación de parámetros en sistemas dinámicos	37
4	Prueba del algoritmo PSO-RE mediante un caso de estudio	41
4.1	Preliminares	41
4.1.1	Modelo del servomecanismo	41
4.1.2	Algoritmos de identificación	42
4.1.3	Planteamiento del problema de optimización	44
4.1.4	Planteamiento experimental	45
4.1.5	Señales de excitación	47
4.2	Análisis de los parámetros identificados	51
4.2.1	Estadística no paramétrica	51
4.2.2	Análisis estadístico de los parámetros	54
4.3	Implementación de los parámetros identificados en una ley de control en lazo cerrado	55
4.4	Conclusiones	58
5	Identificación paramétrica en condiciones de ruido utilizando el algoritmo PSO	62
5.1	Identificación de parámetros con ruido de medición como problema de optimización numérica	62
5.1.1	Ejemplo numérico	64
5.2	Sintonización de filtros continuos como un problema de optimización Multi-Objetivo	69
5.3	Algoritmos de optimización para problemas Multi-Objetivo	72
5.3.1	Multi-Objective Particle Swarm Optimization (MOPSO)	73
5.3.2	Non-Dominated Sorting Genetic Algorithm II (NSGA-II)	73
5.4	Optimización Multi-Objetivo de la frecuencia de corte de un filtro continuo aplicado a señales con ruido	75
5.4.1	Modelo del servomecanismo	78
5.4.2	Identificación de la frecuencia de corte y el orden de un filtro Butterworth mediante algoritmos de optimización Multi-Objetivo	79
5.4.3	Planteamiento experimental	80
5.4.4	Frentes de Pareto resultantes de la optimización del filtro	81
5.5	Caso de estudio: Identificación de parámetros de un servomecanismo empleando señales con ruido de medición	85
5.5.1	Identificación de parámetros empleando algoritmos PSO	85
5.5.2	Planteamiento del problema de optimización	86
5.5.3	Planteamiento experimental	87

5.5.4	Análisis de los parámetros identificados	88
5.5.5	Implementación de los parámetros identificados en una ley de control por realimentación aplicada al servomecanismo	97
5.6	Conclusiones	99
6	Aplicación de funciones de penalización en la sintonización de observadores de estado extendido utilizando PSO	102
6.1	Observador de Estado Extendido (OEE)	103
6.2	Sintonización de las ganancias del OEE como un problema de optimización	104
6.2.1	Sintonización considerando cotas inferiores (CI)	105
6.2.2	Sintonización considerando cotas inferiores y superiores (CIS)	106
6.2.3	Sintonización utilizando funciones de penalización (FP)	106
6.3	Caso de estudio: Péndulo actuado sujeto a perturbaciones exógenas	109
6.3.1	Planteamiento experimental	109
6.3.2	Análisis estadístico	111
6.3.3	Desempeño del Observador de Estado Extendido	112
6.4	Conclusiones	115
7	Conclusiones	117
7.1	Trabajo Futuro	118
	Bibliografía	120

Agradecimientos

Agradezco al CONACYT por el apoyo dado mediante la beca **585250** para la realización de los estudios dentro del programa.

Agradezco al CINVESTAV por la oportunidad de realizar mis estudios de doctorado dentro de la institución.

Agradezco a mi asesor el **Dr. Rubén Alejandro Garrido Moctezuma** por el apoyo para el desarrollo del tema de la tesis y el ser capaz de vislumbrar las oportunidades de investigación en un tema que ha tenido poco desarrollo anteriormente.

Agradezco al **Dr. Efrén Mezura Montes** por su apoyo dentro del área de los algoritmos metaheurísticos y el mantener la mente abierta en los problemas relacionados al Control Automático.

Agradezco el apoyo de mi familia y amigos durante el proceso de realización del programa de doctorado.

Resumen

La presente tesis tiene como finalidad el estudio de las aplicaciones del algoritmo Particle Swarm Optimization (PSO) en problemas de identificación paramétrica, sintonización de filtros y selección de ganancias para Observadores de Estado Extendido (OEE). A pesar de que estos algoritmos se han empleado para solucionar estos problemas, los trabajos que existen no toman en cuenta aspectos que ya han sido estudiados de forma previa en el área del Control Automático empleando métodos clásicos no heurísticos de identificación y sintonización.

El primer aspecto es el efecto de la Riqueza Espectral (RE) de las señales de excitación utilizadas en la identificación de parámetros de un sistema dinámico. En este trabajo se muestra que la variabilidad de los parámetros estimados producidos por el algoritmo PSO crece cuando se utilizan señales de excitación con un valor bajo de RE. Se propone un nuevo algoritmo denominado PSO de Riqueza Espectral (PSO-RE), este modifica su dinámica respecto al valor de la RE de la señal usada para excitar el sistema a identificar con el fin de disminuir la variabilidad de los parámetros estimados cuando el valor de la RE es pequeño.

El segundo aspecto es el efecto del ruido de medición en la identificación de parámetros de un sistema dinámico. En este caso se propone la sintonización de los filtros usados para atenuar el ruido en las mediciones empleadas en la identificación paramétrica mediante técnicas de Optimización Multi-Objetivo.

El último aspecto es la sintonización de las ganancias de un OEE mediante el uso de Funciones de Penalización (FP) dentro de la función de costo de un algoritmo PSO. Estas dependen de los polos del polinomio característico del observador y permiten descartar aquellas ganancias que dan como resultado una dinámica inestable del observador. El uso de las FP permite que las ganancias calculadas produzcan una baja variabilidad y un bajo nivel de error de observación cuando son implementadas en el OEE.

Abstract

The purpose of this thesis is to study the applications of the Particle Swarm Optimization (PSO) algorithm in parametric identification, filter tuning and selection of the gains for Extended State Observers (ESO). Although the PSO algorithms have been used to solve these problems, the existing works do not consider issues that have been previously studied in the area of Automatic Control using classical non-heuristic methods of identification and tuning.

The first issue is the effect of the Spectral Richness (SR) of the excitation signals used in the parameter identification of a dynamic system. In this work it is shown that the variability of the estimated parameters produced by the PSO algorithm increases when excitation signals with a low RE value are used. A new algorithm called the Spectral Richness PSO (PSO-RE) is proposed. It modifies its dynamics with respect to the SR value of the signal used to excite the system to be identified in order to decrease the variability of the estimated parameters when the SR is small.

The second issue is the effect of measurement noise on the parameter identification of a dynamic system. In this case, the tuning of the filters that are employed to attenuate the noise in the measurements used in the parametric identification is performed using Multi-Objective Optimization techniques where the optimization is carried out by means of a PSO algorithm.

The last issue is the gain tuning of an OEE through the use of Penalty Functions (PFs) employed within the cost function of a PSO algorithm. The PFs depend on the poles of the observer characteristic polynomial and allow discarding those gains producing unstable observer dynamics. The use of PFs allows the calculated gains to produced low variability and low level of observation error when implemented in the OEE.

Capítulo 1

Introducción

La solución de los problemas del área del Control Automático usualmente se encuentra limitada por la necesidad de encontrar soluciones analíticas o en dado caso plantear algoritmos, leyes de control o leyes de adaptación, los cuales deben cumplir con criterios como la estabilidad y la optimalidad.

Dentro de esta área de estudio existe varios problemas que serán estudiados en este trabajo, los cuales son:

- **Identificación de parámetros:** Consiste en la estimación de los valores numéricos de los parámetros de un modelo que describe a un sistema físico, de tal forma que se minimice de acuerdo a un criterio de desempeño, la diferencia entre los datos experimentales y el modelo simulado con los valores estimados. El modelo y los parámetros estimados pueden ser utilizados para diseñar y proponer leyes de control.
- **Sintonización de filtros continuos:** La sintonización de filtros consiste en la selección de la frecuencia de corte y el orden del filtro, con el fin de atenuar componentes frecuenciales no deseadas en una señal. En el caso del Control Automático, la sintonización de filtros para llevar acabo el acondicionamiento de señales influye en el desempeño de controladores, observadores y en técnicas de identificación de parámetros.
- **Sintonización de ganancias en observadores:** La selección de ganancias de un observador consiste en determinar sus valores tales que se obtenga un convergencia rápida a los estados pero sin la presencia de sobretiros u oscilaciones. Las ganancias seleccionadas deben asegurar que la dinámica del observador de como resultado un sistema exponencialmente estable.

El empleo de algoritmos metaheurísticos, en particular la Optimización por Enjambre de Partículas (PSO- Particle Swarm Optimization) ofrece algunas ventajas

en la solución de los problemas antes mencionados, como lo es evitar la necesidad de conocer el estado completo del sistema dinámico. Además, algunos de los aspectos considerados al resolver estos problemas empleando métodos clásicos no se han abordado en el caso de los algoritmos metaheurísticos.

1.1. Algoritmos metaheurísticos

Los algoritmos metaheurísticos son métodos computacionales que permiten solucionar problemas de optimización. Un problema de optimización consiste en encontrar un valor óptimo $x_* \in \Omega$ que minimice una función de desempeño $J(\cdot)$ con un número de operaciones razonable siendo Ω el conjunto de soluciones factibles. La metodología utilizada por este tipo de algoritmos consiste en la evaluación de soluciones $x \in \Omega$ durante múltiples iteraciones con la finalidad de que cada iteración k proporcione una solución más cercana a x_* . Por lo que una solución $x(k)$ es utilizada por el algoritmo metaheurístico para calcular la solución de la siguiente iteración $x(k+1)$, tal que:

$$J(x(k)) \geq J(x(k+1)) \quad (1.1)$$

La generación de nuevas soluciones es llevada a cabo por medio un conjunto de reglas o expresiones matemáticas. Estas suelen consistir en la emulación de fenómenos observados en la naturaleza tales como el proceso evolutivo de organismos biológicos [1], la dinámica en la búsqueda de comida realizada por las colonias de hormigas [2], el comportamiento social de los individuos en una población representado como un enjambre de partículas [3], la respuesta del sistema inmunológico humano a un estímulo desconocido [4], entre otros. Dependiendo del problema de optimización se selecciona el tipo de algoritmo metaheurístico [5] que proporcione una relación adecuada del tiempo de ejecución y la capacidad de obtener soluciones óptimas.

Los problemas de optimización numéricos pueden ser divididos en dos clases: El primero de ellos son los problemas Mono-Objetivo donde se tiene una única función de desempeño que se desea minimizar y el algoritmo produce una única solución. Por otra parte, existen los problemas Multi-Objetivo donde se emplean cuando al menos dos funciones de desempeño que se encuentran en conflicto una con la otra, por lo que es necesario buscar un conjunto de soluciones que presenten compromisos entre las funciones para posteriormente decidir el tipo de compromiso que se considerará para seleccionar la solución.

1.1.1. Problema de optimización numérica Mono-Objetivo

Un problema de optimización numérica consiste en encontrar una solución óptima $x_* \in \Omega_s \subset \mathbb{R}^n$ tal que se minimice un índice de desempeño $J(\tau)$ con $\tau \in \Omega_s \subset \mathbb{R}^n$. El conjunto Ω_s contiene todas las soluciones factibles τ que cumplen con un grupo de restricciones definidas como [6]:

$$\Omega_s := \{\tau \in \mathbb{R}^n | g_1(\tau) \leq 0, \dots, g_{r_d}(\tau) \leq 0, h_1(\tau) = 0, \dots, h_{r_i}(\tau) = 0\} \quad (1.2)$$

donde las funciones $g_i(\tau)$ corresponden a las r_d restricciones de desigualdad y las funciones $h_i(\tau)$ son las r_i restricciones de igualdad a las que se encuentra sujeto el problema de optimización.

El proceso de optimización supone la existencia de un valor óptimo τ_* que minimiza $J(\tau)$ tal que:

$$J(\tau_*) \leq J(\tau) \quad \tau_* \in \Omega_s \quad \forall \quad \tau \in \Omega_s \quad (1.3)$$

Dentro de los algoritmos de optimización numérica Mono-Objetivo se encuentra el algoritmo de Evolución Diferencial [7]. En éste se emulan los cambios entre los algoritmos de una población respecto al paso generacional además, en este se incluyen operaciones como lo son la mutación y la cruce entre los individuos para generar nuevas soluciones. Otro ejemplo es el algoritmo de Colonia de Hormigas [8], donde las soluciones son calculadas basándose en la dinámica de búsqueda de alimento de un grupo de hormigas. Se emplea la información previa recolectada por otros individuos para seguir un camino hacia soluciones óptimas previamente conocidas para utilizarla con el fin de incrementar la diversidad de las nuevas soluciones. Un comportamiento similar se muestra en el caso de los algoritmos de Colonias de Abejas [9], presentando algunas diferencias en la forma en que las soluciones se mueven en el espacio Ω . Uno de los algoritmos que reporta una mayor rapidez de convergencia en las soluciones y facilidad de implementación es el algoritmo de Optimización por Enjambre de Partículas (PSO-Particle Swarm Optimization) [10], el cual se encuentra inspirado en la forma en que los pájaros comparten información respecto a sus fuentes de alimentación. Este será el algoritmo utilizado en este trabajo por la simplicidad de las expresiones matemáticas que emplea.

1.1.2. Problemas de optimización numérica Multi-Objetivo (MO)

El problema de optimización numérica Multi-Objetivo (MO) corresponde a la búsqueda de un vector de variables $v = [v_1, v_2, \dots, v_d]$ que deben minimizar un vector \mathcal{J} de $m \geq 2$ funciones objetivo, donde cada una de ellas se encuentra en conflicto con

las otras [11]:

$$\min(\mathcal{J}(v)) = [\mathcal{J}_1(v), \dots, \mathcal{J}_m(v)]^T \quad (1.4)$$

El conjunto de soluciones factibles del problema de optimización numérica MO es denotado por Ω_M y se define como:

$$\begin{aligned} \Omega_M := \{v \in \mathbb{R}^m \mid & g_{m1}(v) \leq 0, \dots, g_{mr_i}(v) \leq 0, \\ & h_{m1}(v) = 0, \dots, h_{mr_e}(v) = 0, \\ & v_k^{\min} \leq v_k \leq v_k^{\max} \quad k = 1, \dots, d\} \end{aligned} \quad (1.5)$$

donde $g_{mi}(v)$ y $h_{mi}(v)$ corresponden a las r_i condiciones de desigualdad e igualdad respectivamente. Las cotas de diseño de las variables v se encuentran dadas por $v_k = [v_k^{\min}, v_k^{\max}]$.

Con la finalidad de entender el problema de optimización numérica MO es necesario dar las siguientes definiciones:

- **Dominancia de Pareto:** Dadas m funciones de desempeño $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$. Entonces, se puede decir que el vector de decisión $\vec{x} = [x_1, \dots, x_d]^T$ domina a un vector $\vec{y} = [y_1, \dots, y_d]^T$, lo cual se denota por:

$$\vec{x} \prec \vec{y} \quad (1.6)$$

si y solo si, $f_i(\vec{x}) \leq f_i(\vec{y})$ para todo $i \in \{1, \dots, m\}$ y $f_i(\vec{x}) < f_i(\vec{y})$ para al menos un f_i .

- **Optimalidad de Pareto:** Un vector de variables de decisión \vec{x}^* pertenece al conjunto de óptimos de Pareto \mathcal{F} si no existe otra solución $\vec{x}^* \in \mathcal{F} \Rightarrow \nexists \vec{x} \prec \vec{x}^*$. Si \vec{x}^* es un óptimo de Pareto, entonces el vector objetivo $\vec{f}(\vec{x}^*)$ también es un óptimo de Pareto.

$$\vec{x}^* \in \mathcal{F} \Rightarrow \nexists \vec{x} \prec \vec{x}^* \quad (1.7)$$

- **Conjunto óptimo de Pareto:** El conjunto que contiene a todos los vectores de decisión que son óptimos según Pareto se denomina Conjunto Óptimo de Pareto (COP). Para un problema de optimización numérica MO dado, el COP denotado por \mathcal{P}^* está definido como:

$$\mathcal{P}^* = \{\vec{x} \in \mathcal{F} \mid \nexists \vec{y} \in \mathcal{F} : \vec{y} \prec \vec{x}\} \quad (1.8)$$

- **Frente de Pareto:** Para el COP \mathcal{P}^* que corresponde a los vectores de decisión \vec{x} se genera un Frente Óptimo de Pareto (FOP). Dado un vector objetivo $\vec{f}(\vec{x})$

y un COP \mathcal{P}^* , el FOP se define como:

$$\mathcal{PF}^* = \{\vec{f}(\vec{x}) | \vec{x} \in \mathcal{P}^*\} \quad (1.9)$$

Para resolver problemas Multi-Objetivo existen múltiples algoritmos enfocados en generar el frente de Pareto correspondiente. Dentro de estos se destacan dos. El primero es el MOPSO [12] que es una ampliación del algoritmo PSO al caso Multi-Objetivo y que permite una rápida convergencia de las soluciones para generar el frente de Pareto. El otro algoritmo es el NSGA-II [13], el cual corresponde a un algoritmo genético que contiene mecanismos de dispersión de soluciones para que éstas no se concentren en un único punto del frente.

1.2. Identificación de parámetros

La tarea de identificación paramétrica consiste en estimar los valores numéricos de los parámetros de un modelo, tal que el comportamiento de éste sea lo más cercano posible a una serie de datos experimentales [14] donde la cercanía se define de acuerdo a un criterio de desempeño. La idea detrás de este proceso es tener un conocimiento más profundo del sistema o fenómeno con el que se está tratando, de forma que se pueda realizar una predicción acertada de su comportamiento futuro o la implementación de estrategias de control.

Sin embargo, la forma en que estos parámetros son calculados puede variar respecto al tipo de algoritmo o técnica que se utilice. Una de las técnicas más utilizadas para llevar a cabo esta tarea son los algoritmos de Mínimos Cuadrados (MC) [14, 15]. Este se basa en la regresión lineal de un modelo y en un índice de desempeño cuadrático. El cálculo de los parámetros mediante esta técnica depende de la resolución de un conjunto de ecuaciones lineales sobre-determinadas, de forma que es necesaria la implementación numérica de la inversa de una matriz lo que puede producir errores de cálculo si ésta se encuentra mal condicionada [16]. Otra técnica ampliamente utilizada es el algoritmo Gradiente, que se usa en la identificación de parámetros en algoritmos de control adaptable [17]. Este se basa en una regresión lineal como es el caso del algoritmo de MC pero utiliza un algoritmo gradiente descendente y los estimados son obtenidos mediante la resolución de un conjunto de ecuaciones diferenciales. El algoritmo de Mínimos Cuadrados Promedio (MCP) ha sido utilizado para el entrenamiento de redes neuronales [18] y en el diseño de filtros adaptables [19]. De forma más reciente el algoritmo Closed-Loop Input Error (CLIE) [20] ha sido propuesto para la identificación de parámetros de sistemas electromecánicos. Este se basa en un

algoritmo de tipo gradiente y la resolución de un conjunto de ecuaciones diferenciales, pero la forma en que se plantea el regresor permite identificar los parámetros de un modelo no-lineal.

La identificación de parámetros también se ha realizado mediante el uso de algoritmos metaheurísticos [21, 22, 23]. Es importante mencionar que la implementación de este tipo de algoritmos evita la problemática de utilizar la estructura del modelo a identificar para generar un regresor o de probar analíticamente la convergencia de los parámetros. En general, estos algoritmos solo requieren el conocimiento de la entrada aplicada al sistema y la salida que este genera.

En particular, en el caso de sistemas electromecánicos existen diversas referencias relacionadas a la identificación mediante algoritmos metaheurísticos incluyendo el algoritmo PSO [24, 25, 26, 27] aplicado a servomecanismos, el algoritmo Cuckoo [28] y el algoritmo de metaheurística Caótica[29] aplicado a motores síncronos, el Optimizador Caótico de Ballena [30] aplicado a motores de imán permanente, y la Evolución Diferencial [31] aplicada a sistemas hidráulicos. Mas aún, los algoritmos metaheurísticos Multi-Objetivo han sido utilizado como idea principal para sintonizar controladores, un tópico estrechamente relacionado con la identificación de parámetros [32, 33]. De forma interesante el algoritmo PSO ha sido utilizado en la literatura para resolver problemas de identificación donde no se puede generar una regresión lineal de los parámetros como es el caso de los paneles foto-voltaicos [21].

1.2.1. Identificación en señales con baja riqueza espectral

La Riqueza Espectral (RE) [34] corresponde al número de líneas espectrales que posee una señal. En el contexto de identificación paramétrica, la relación entre la RE y la condición de Excitación Persistente es presentada en [35], donde se muestra que dependiendo del número de líneas espectrales en la señal de excitación se puede asegurar la convergencia de los parámetros estimados a los valores reales. El efecto de estas condiciones en la identificación de parámetros mediante técnicas del área de Control Automático ha sido ampliamente documentado [36]. Sin embargo, en el caso de las técnicas metaheurísticas no se ha estudiado el impacto que la Riqueza Espectral en los estimados obtenidos. Se han desarrollado algoritmos de Control Adaptable [37] que permiten la convergencia de los parámetros estimados ante la falta de condiciones de Excitación Persistente, sin embargo, estos solo aseguran la convergencia del error de seguimiento de trayectoria y no de los parámetros estimados. En [38] se muestra un algoritmo de identificación de parámetros que funciona en ausencia de la condición de Excitación Persistente del modelo de una plataforma reconfigurable compuesta por tres robots acuáticos, que son capaces de cambiar de posición sobre la superficie

del agua para generar diferentes formas. Este enfoque utiliza un número de datos finitos para llevar a cabo el proceso de identificación de parámetros. Una comparación del comportamiento de los algoritmos de identificación de parámetros cuando no se cumple la condición de Excitación Persistente, se presenta en [39] donde se demuestra que ciertos algoritmos como el Aprendizaje Concurrente [40] presentan convergencia en tiempo finito de los parámetros identificados sin que se cumpla la condición de Excitación Persistente.

1.2.2. Identificación en presencia de ruido de medición

El problema de identificación de parámetros cuando existe ruido de medición se ha estudiado en el área de Control Automático. En el trabajo realizado en [41] se considera al ruido como una señal acotada y se emplea una formulación recursiva del algoritmo de Mínimos Cuadrados para realizar la tarea de identificación. La manera en que el ruido afecta la identificación de parámetros se estudia en [42], donde se propone el uso de un identificador adaptable capaz de obtener valores de los parámetros estimados cercanos a los parámetros del sistema si la señal de excitación se selecciona adecuadamente. Este esquema funciona incluso si existe ruido de medición en la salida medible.

El efecto del ruido de medición durante el proceso de identificación de parámetros mediante un algoritmo de Mínimos Cuadrados adaptables con factor de olvido aplicado al caso de la identificación de baterías de ion litio se estudia en [43], donde se muestra que el error de estimación de los parámetros estimados del modelo que representan la capacidad de carga de la batería de ion-litio se incrementa cuando existe ruido de medición. La robustez de un identificador de parámetros con un controlador en lazo cerrado en presencia de ruido para un oscilador no lineal se estudia en [44]. Se muestra que el ruido de medición produce los parámetros estimados con saltos entre los dominios de atracción de dos soluciones estables que coexisten entre sí. En [45] se estudia la identificación de parámetros mediante un identificador basado en la teoría de Modos Deslizantes en presencia de ruido de medición, se reporta un desempeño aceptable en la identificación de parámetros si la magnitud del ruido está acotada y este es filtrado mediante el uso de un integrador.

En el caso de los algoritmos metaheurísticos se ha estudiado cómo la presencia de ruido en los datos afecta la calidad de los parámetros estimados. Se propone en [46] el uso del algoritmo de Evolución Diferencial empleando una población virtual y un modelo probabilístico para el cálculo de nuevos individuos de la población cuando existe ruido en los datos empleados en el algoritmo. En [47] se utiliza un algoritmo de Colonia de Abejas que emplea un método adaptable de selección del tamaño de

muestras y emplea un análisis estadístico de probabilidad de la función de desempeño para el cálculo de nuevas soluciones cuando se tiene ruido en los datos utilizados. Un algoritmo PSO que modifica el comportamiento de las partículas cuando se tiene presencia de ruido en los datos se propone en [48], en este enfoque se realiza un cambio en la frecuencia de muestreo de los datos que son introducidos en el PSO con la finalidad de atenuar el ruido de alta frecuencia. Los estudios mencionados previamente respecto a los algoritmos metaheurísticos no consideran la identificación de parámetros de un sistema dinámico.

Una comparación de la identificación de parámetros para un sistema caótico de orden fraccionario cuando la salida posee distribuciones de ruido Gaussiano, de Cauchy y de Levy se realiza en [49]. Se muestra que los estimados obtenidos con una señal con ruido Gaussiano son los que presentan un menor error de estimación paramétrica que los parámetros obtenidos en los casos que el ruido tiene una distribución de Cauchy o de Levy. En [50] se evalúan varios algoritmos metaheurísticos para realizar la identificación de parámetros de un sistema fotovoltaico, se muestra que si la magnitud de ruido en la señal es superior al 5% respecto de la magnitud de la señal ideal, los parámetros tales como la resistencia del panel fotovoltaico y la corriente de saturación inversa del diodo tienen un error relativo superior al 73% y 210% respectivamente. La Relación Señal Ruido (RSR) ha sido empleada en [51, 52] para evaluar la calidad de las señales utilizadas en la identificación de parámetros de un sistema dinámico utilizando el algoritmo PSO. En [53] se realiza la identificación de parámetros de un motor síncrono de imán permanente utilizando el algoritmo PSO, para resolver el ruido de medición se realiza previamente un ajuste de los datos a una curva exponencial, que elimina algunas de las componentes de alta frecuencia de ruido de medición presentes en la señal de salida.

1.3. Sintonización de observadores de estado extendido

Los estados de un sistema dinámico no siempre se encuentran disponibles para su medición, ya sea por razones económicas o por la naturaleza misma del sistema. Para tener acceso a dichos estados se utilizan los observadores, los cuales producen una estimación del estado a partir de mediciones disponibles.

El estimador de estados clásico por excelencia es el Observador de Luenberger [54], para su construcción este requiere conocimiento del modelo del sistema dinámico y de sus parámetros. Esto no siempre es posible lo cual hace necesario emplear esquemas alternativos para enfrentar este problema. En [55] utilizan la teoría de Modos Deslizantes mediante la técnica Super-Twisting para estimar los estados de un

sistema mecánico de segundo orden, sin embargo, los estados estimados presentan el fenómeno de castaño y el observador es altamente sensible al ruido de alta frecuencia que existe en la salida del sistema. Un observador que permite identificar la perturbación que afecta a un sistema se presenta en [56] al considerar una aproximación de la dinámica de la perturbación basada en un polinomio temporal de orden $p - 1$. En [57] se muestran los Observadores de Estado Extendido (OEE) los cuales estiman los estados del sistema y la perturbación que lo afecta mediante el uso de la teoría de los Integradores Proporcionales Generalizados. Estos son utilizados en el control por Rechazo Activo de Perturbaciones.

En la literatura se ha propuesto el uso de algoritmos metaheurísticos para la sintonización de las ganancias de los observadores como se puede observar en [58] y en [59], donde esta tarea se realiza utilizando el algoritmo de Colonia de Hormigas y el algoritmo de Palomas respectivamente. La función de desempeño utilizada toma en cuenta el sobretiro, el tiempo de convergencia y el error en estado estacionario. De igual manera, en [60] se muestra la sintonización conjunta de los quince parámetros de un OEE y de un control por Rechazo Activo de Perturbaciones mediante el algoritmo de Selección Caótica de Clonación de Nubes a través de una función de desempeño que considera el error de estimación y el error de seguimiento de trayectoria. En [61] se sintoniza un OEE mediante el algoritmo PSO considerando una función de desempeño basada en el valor absoluto del error ponderado multiplicado por el tiempo, de tal manera que la ponderación del error se incrementa con el transcurso del tiempo. La estabilidad del OEE ha sido tomada en cuenta de manera indirecta en [62], donde se propone una parametrización de las ganancias del observador respecto a la frecuencia natural no amortiguada de un sistema de segundo orden. De esta forma el algoritmo PSO ajusta este parámetro para asegurar que las soluciones obtenidas sean estables. Este esquema genera polos repetidos en el polinomio característico del observador lo que limita su flexibilidad.

1.4. Objetivo General

Estudiar las aplicaciones del algoritmo PSO en problemas de identificación paramétrica, sintonización de filtros y sintonización de ganancias en Observadores de Estado Extendido.

1.5. Objetivos Específicos

- Estudiar el efecto de la Riqueza Espectral en el desempeño del algoritmo PSO como identificador de parámetros de un sistema dinámico y proponer una variante del algoritmo PSO que disminuya la variabilidad de las soluciones cuando existe una baja Riqueza Espectral.
- Proponer la sintonización de un filtro como un problema de optimización Multi-Objetivo con la finalidad de acondicionar señales con ruido de medición utilizadas en un esquema de identificación de parámetros.
- Formular una función de desempeño que tome en cuenta las condiciones de estabilidad de un Observador de Estado Extendido mediante la implementación de funciones de penalización.

1.6. Estructura de la tesis

La estructura de la tesis presentada es la siguiente. En el Capítulo 2 se plantea la utilización del algoritmo PSO como identificador de parámetros en sistemas dinámicos al considerar esta tarea como un problema de optimización. El Capítulo 3 describe el efecto de la Riqueza Espectral en la identificación de parámetros utilizando el algoritmo PSO. El Capítulo 4 desarrolla el planteamiento de una nueva variante del algoritmo PSO que disminuye la variabilidad de las soluciones obtenidas cuando se aplica una señal con baja Riqueza Espectral como referencia. En el Capítulo 5 se presenta la identificación de parámetros en presencia de ruido de medición y la sintonización de filtros para el acondicionamiento de señales como un problema de optimización Multi-Objetivo. En el Capítulo 6 se plantea la sintonización de Observadores de Estado Extendido mediante el algoritmo PSO considerando la utilización en la función de desempeño de funciones de penalización dependientes de los polos del observador. Finalmente, el Capítulo 7 presenta las conclusiones de la tesis.

1.7. Publicaciones

1.7.1. Artículo de revista

Actualmente se encuentran sometido el siguiente artículo:

- R. Cortez, R. Garrido, E. Mezura-Montes. **Spectral Richness PSO algorithm for parameter identification of dynamical systems under non-**

ideal excitation conditions. Applied Soft Computing. Sometido el 26 de Junio de 2021.

1.7.2. Artículos de congreso internacional

- R. Cortez-Vega, J. Maldonado, Rubén Garrido. **Parameter Identification using PSO under measurement noise conditions.** 6th International Conference on Control Decision and Information Technologies. París, Francia, 23 al 26 de Abril (2019):103-108
- R. Cortez, Rubén Garrido. **Stable Tuning of Extended State Observers using PSO and Penalty Functions** 2021 IEEE International Conference on Mechatronics and Automation (ICMA). Takamatsu, Japón, 8 al 11 de Agosto (2021): 471-476

1.7.3. Artículos de congreso nacional

- R. Cortez-Vega, Rubén Garrido. **Identificación paramétrica de un motor de CD utilizando el algoritmo de evolución diferencial.** Memorias del Congreso Nacional de Control Automático. San Luis Potosí, México, 10 a 12 de Octubre (2018): 90—95.

Capítulo 2

Algoritmo PSO como identificador de parámetros en sistemas dinámicos

El algoritmo PSO (Particle Swarm Optimization por su siglas en inglés) [10] es una metaheurística utilizada para resolver problemas de optimización numéricos mediante un proceso de búsqueda en el espacio de soluciones factibles Ω_s . En este capítulo se presentarán el algoritmo PSO clásico y sus variantes que se utilizarán dentro de este trabajo. Además, se expresará la manera en que se debe plantear el problema de identificación de parámetros en un sistema dinámico como un problema de optimización y la forma en que éste se debe implementar empleando el algoritmo PSO para asegurar la validez de los resultados.

2.1. Algoritmo PSO clásico

La idea principal en la que se basa este algoritmo es en la imitación del comportamiento social de un grupo de ρ individuos denominado enjambre. A los individuos se les denomina partículas y la posición de cada una de ellas corresponde a una solución potencial de un problema de optimización [63].

Cada partícula posee un comportamiento dinámico correspondiente a un sistema discreto de segundo orden (ver Fig. 2.1), que se describe mediante las siguientes ecuaciones en diferencias:

$$\begin{aligned}\omega_i(k+1) &= \omega_i(k) + \phi_1(k)p(x_{i,*}(k) - x_i(k)) + \phi_2(k)l(x_*(k) - x_i(k)) \\ x_i(k+1) &= \omega_i(k+1)\end{aligned}\tag{2.1}$$

donde los vectores $x_i(k), \omega_i(k) \in \mathbb{R}^n$ corresponden a la posición y la velocidad de la partícula i durante la iteración k respectivamente. La posición $x_i(k)$ corresponde a una solución del problema de optimización. Las posiciones iniciales de cada una de

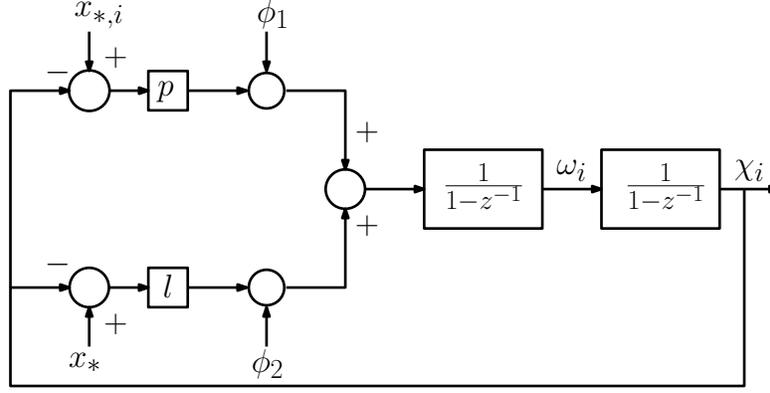


Figura 2.1: Representación del sistema discreto de segundo orden que corresponde a la dinámica de la partícula i en el algoritmo PSO. El término z^{-1} corresponde al operador retardo.

las partículas se generan de forma aleatoria tal que $x_i(0) \in \Omega_s$.

Las señales aleatorias ϕ_1 y ϕ_2 se encuentran acotadas en el intervalo $(0, 1]$. Los términos $p, l \in \mathbb{R}^+$ corresponden a los valores de ponderación que se le otorga a la diferencia de la posición $x_i(k)$ de la partícula respecto a su mejor posición previa $x_{*,i}(k)$ y a la mejor posición del enjambre $x_*(k)$ respectivamente.

Toda solución $x_i(k)$ es usada para evaluar el índice de desempeño $J(x_i(k))$ y basándose en las evaluaciones se calcula el término $x_{i,*}$ también conocido como **pbest**. Este corresponde a posición de la partícula i que ha generado un menor valor al ser evaluada en la función de desempeño:

$$x_{*,i}(k) = \underset{s \in [1, \dots, k]}{\operatorname{argmin}} \{J(x_i(s))\} \quad (2.2)$$

De forma análoga se puede calcular el término $x_*(k)$ conocido como **gbest** el cual corresponde a la solución mejor evaluada en todo el enjambre:

$$x_*(k) = \underset{g \in [1, \dots, \rho]}{\operatorname{argmin}} \{J(x_g(k))\} \quad (2.3)$$

Con la finalidad de mantener las soluciones calculadas mediante las ecuaciones (2.1) dentro del conjunto Ω_s es necesario implementar un mecanismo de manejo de restricciones. En este caso se propone que si $x_i(k) \notin \Omega_s$ entonces la velocidad $\omega_i(k)$ se reduce a la mitad de su valor original y se vuelve a calcular una posición de $x_i(k)$. Este proceso se repite hasta que se cumple la condición de que $x_i(k) \in \Omega_s$.

Para implementar el algoritmo (2.1) se proponen dos criterios de paro: El primero corresponde al número máximo de iteraciones que éste debe ejecutar el cual se denota como k_{max} , el segundo criterio es un valor de la función de desempeño mínimo α tal que

una vez que una partícula sea evaluada y obtenga un valor menor a éste, el algoritmo se detiene. La implementación del algoritmo PSO se describe como pseudo-código en el Algoritmo 1.

Algoritmo 1: Optimización por Enjambre de Partículas (PSO)

Datos: $J(\cdot)$ and Ω_s
Resultado: x_*

- 1 Generar soluciones aleatorios $x_i(1) \in \Omega_s \quad i = 1, \dots, r$;
- 2 Evaluar las soluciones $x_i(1)$ usando $J(\cdot)$;
- 3 Obtener los términos $x_{i,*}(1)$ and $x_*(1)$;
- 4 Calcular las velocidades $\omega_i(2)$ y las posiciones $x_i(2)$;
- 5 Colocar $k = 2$;
- 6 **Mientras** $k < k_{max}$ **hacer**
- 7 Evaluar las soluciones $x_i(k)$ usando $J(\cdot)$;
- 8 Obtener los términos $x_{i,*}(k)$ and $x_*(k)$;
- 9 **Si** $J(x_*(k)) \geq \alpha$ **entonces**
- 10 Calcular las velocidad $\omega_i(k)$ y las posiciones $x_i(k)$;
- 11 **Mientras** $x_i(k) \notin \Omega_s$ **hacer**
- 12 Disminuir la velocidad tal que $\omega_i(k) = \frac{\omega_i(k)}{2}$;
- 13 Calcular una nueva posición $x_i(k)$ using $\omega_i(k)$;
- 14 **Fin**
- 15 **En otro caso**
- 16 **Detenerse** ;
- 17 **Fin**
- 18 $k = k + 1$;
- 19 **Fin**

Con la finalidad de mejorar los resultados arrojados por el algoritmo PSO se han propuesto variaciones en su topología las cuales producen cambios en la velocidad de convergencia y en la capacidad de búsqueda dentro del espacio de soluciones. El concepto de topología para el algoritmo PSO está relacionado con la manera en cómo las partículas interactúan entre sí. Esto se refleja en el algoritmo PSO en la forma en que se define el término $x_*(k)$ [64]. La definición utilizada en (2.3) corresponde a una topología del tipo **G-best** donde todos los individuos en el enjambre comparten información respecto a la mejor posición obtenida por el enjambre de forma que todas las partículas se ven atraídas a la misma posición. Este tipo de comportamiento mejora la velocidad de convergencia pero implica una menor capacidad de búsqueda en el espacio de soluciones.

Existe una segunda topología denominada **L-best** [65] la cual divide el enjambre

en v cantidad de vecindarios de tal forma que el término (2.3) se reescribe como:

$$x_*(k) = \underset{g \in v_s}{\operatorname{argmin}} \{J(x_{*,g}(k))\} \quad (2.4)$$

donde los vecindarios se definen como $v_s = [x_{l_s}, x_{u_s}]$, siendo l_s y u_s las cotas de las partículas que pertenecen al vecindario s . Este tipo de topología incrementa la diversidad de las soluciones y permite una mayor capacidad de búsqueda en el conjunto de soluciones factibles, pero a su vez aumenta el número de iteraciones requeridas para que el algoritmo logre la convergencia a un resultado.

2.2. Variantes del algoritmo PSO

Existen variantes del algoritmo PSO original que modifican su desempeño en el proceso de optimización. Estas se basan en una modificación de las expresiones matemáticas que definen el cálculo de la velocidad de las partículas. El trabajo realizado en esta tesis contempla cuatro principales variantes: PSO con factor de inercia, PSO con factor de inercia dinámico, PSO con término de restricción y PSO con término de restricción dinámico.

2.2.1. Algoritmo PSO con Ponderación de Velocidad Variable (PSO-PVV)

El algoritmo PSO con factor de inercia [66] es una variante del algoritmo original donde la velocidad ω_i se ve modificada al añadir un término de inercia, β de tal forma que el algoritmo se reescribe como:

$$\begin{aligned} \omega_i(k+1) &= \beta\omega_i(k) + \phi_1(k)p(x_{i,*}(k) - x_i(k)) + \phi_2(k)l(x_*(k) - x_i(k)) \\ x_i(k+1) &= \omega_i(k+1) \end{aligned} \quad (2.5)$$

donde $\beta \in (0, 1]$. Los valores cercanos a 1 permiten que la búsqueda de soluciones se realice de forma global, mientras que valores más pequeños favorecen la búsqueda local.

Este algoritmo ha sido mejorado mediante la sustitución del término constante β por uno variante en el tiempo que depende del número de iteraciones que ha realizado el algoritmo. La expresión propuesta es la siguiente:

$$\beta(k) = \beta_{\max} - \frac{k}{k_{\max}}(\beta_{\max} - \beta_{\min}) \quad (2.6)$$

donde β_{\max} y β_{\min} son las cotas superior e inferior para el término $\beta(k)$. La idea

principal de esta modificación es que durante las iteraciones iniciales se realice una búsqueda global pero conforme estas se incrementen se ponga énfasis en realizar la búsqueda de forma local.

2.2.2. Algoritmo PSO con Factor de Constricción Variable (PSO-FCV)

El algoritmo PSO con término de constricción [67] también es una modificación del algoritmo original en el término ω_i , pero a diferencia del algoritmo con factor de inercia, el término de constricción γ añadido afecta a todos los términos de la ecuación que genera la velocidad. La dinámica de cada partícula en este caso corresponde a:

$$\begin{aligned}\omega_i(k+1) &= \gamma [\omega_i(k) + \phi_1(k)p(x_{i,*}(k) - x_i(k)) + \phi_2(k)l(x_*(k) - x_i(k))] \\ x_i(k+1) &= \omega_i(k+1)\end{aligned}\quad (2.7)$$

donde $\gamma \in (0, 1]$. El término de constricción evita que la velocidad ω_i crezca de forma descontrolada y produzca soluciones x_i que abandonen el conjunto Ω_s .

De forma similar al caso del algoritmo con factor de inercia, se ha propuesto un algoritmo PSO con término de constricción dinámico substituyendo un valor constante de γ por uno $\gamma(k)$ dependiente del número de iteraciones. Este término se define como:

$$\gamma(k) = (\gamma_{\max} - \gamma_{\min}) \frac{k_{\max} - k}{k_{\max}} + \gamma_{\min} \quad (2.8)$$

donde γ_{\max} y γ_{\min} son las cotas superior e inferior del término de constricción. Esta estructura permite que la velocidad de las partículas disminuya conforme la cantidad de iteraciones k se acerca a k_{\max} .

2.2.3. Algoritmo PSO de Orden Fraccionario Regulado (PSO-OFR)

El algoritmo PSO de Orden Fraccionario Regulado (PSO-OFR) [68] usa la distancia entre las partículas para ajustar la velocidad de orden fraccionario que se proponen para las partículas.

La dinámica de las partículas que produce este algoritmo se encuentra dada por:

$$\begin{aligned}
\omega_i(k+1) &= \zeta(k)\omega_i(k) + \frac{1}{2}\zeta(k)(1-\zeta(k))\omega_i(k-1) \\
&\quad + \frac{1}{6}\zeta(k)(1-\zeta(k))(2-\zeta(k))\omega_i(k-2) \\
&\quad + \frac{1}{24}\zeta(k)(1-\zeta(k))(2-\zeta(k)) \\
&\quad \quad (3-\zeta(k))\omega_i(k-3) \\
&\quad + \phi_1(k)p(k)(x_{i^*}(k) - x_i(k)) \\
&\quad + \phi_2(k)l(k)(x_*(k) - x_i(k)) \\
x_i(k+1) &= x_i(k) + \omega_i(k+1)
\end{aligned} \tag{2.9}$$

donde $p(k)$ y $l(k)$ son calculados como:

$$\begin{aligned}
p(k) &= (p_{\min} - p_{\max}) \left(\frac{k_{\max} - k}{k_{\max}} \right) + p_{\max} \\
l(k) &= (l_{\min} - l_{\max}) \left(\frac{k_{\max} - k}{k_{\max}} \right) + l_{\max}
\end{aligned} \tag{2.10}$$

El término $\zeta(k)$ indica el orden fraccionario de la velocidad y es regulado de forma adaptable basándose en el estado del enjambre:

$$\zeta(k) = 0.9 - \frac{1}{1 + e^{E_f(k)}} \frac{k}{k_{\max}} \tag{2.11}$$

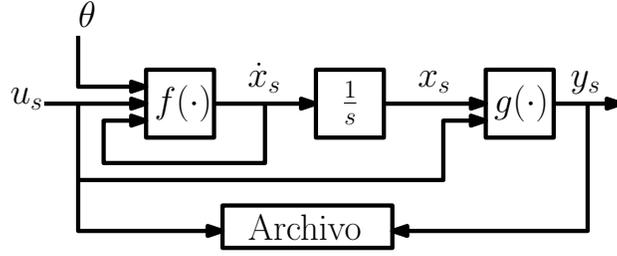
donde E_f es un factor evolutivo que refleja el estado de la evolución de enjambre y se calcula como:

$$E_f(k) = \frac{d_{gb}(k) - d_{\min}(k)}{d_{\max}(k) - d_{\min}(k)} \tag{2.12}$$

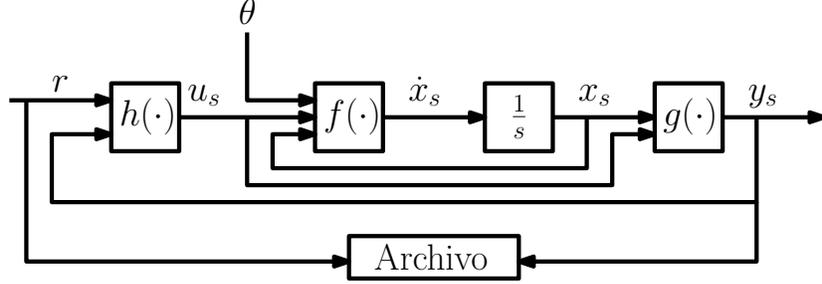
aquí, los términos d_{\max} y d_{\min} representan las distancias máxima y mínima entre las partículas del enjambre respectivamente. La distancia entre las partículas es calculada como:

$$d_i(k) = \frac{1}{\rho - 1} \sum_{j=1, j \neq i}^{\rho} \sqrt{\sum_{k=1}^m (x_i(k) - x_j(k))^2} \tag{2.13}$$

El término d_{gb} representa la distancia promedio $d_i(k)$ entre x_* definida en (2.3) y $x_j(k)$, $j \neq i$, $\forall j = 1, \dots, \rho$.



(a) Proceso de adquisición de datos para un sistema estable en lazo abierto.



(b) Proceso de adquisición de datos utilizando una ley de control que establezca un sistema inestable en lazo abierto.

Figura 2.2: Proceso de adquisición de datos de un sistema dinámico.

2.3. Identificación de parámetros como un problema de optimización

Considérese el siguiente sistema dinámico [69]:

$$\begin{aligned}\dot{x}_s &= f(t, x_s, u_s, \theta) \\ y_s &= g(t, x_s, u_s)\end{aligned}\tag{2.14}$$

donde $x_s \in \mathbb{R}^n$ son los estados del sistema, $y_s \in \mathbb{R}$ es la salida, $u_s \in \mathbb{R}$ corresponde a la señal de entrada, $\theta \in \mathbb{R}^m$ es un vector de parámetros constantes y t es el tiempo.

Considérese también el modelo del sistema (2.14):

$$\begin{aligned}\dot{x}_m &= f(t, x_m, u_m, \hat{\theta}) \\ y_m &= g(t, x_m, u_m)\end{aligned}\tag{2.15}$$

siendo $x_m \in \mathbb{R}^n$, $y_m \in \mathbb{R}$ y $u_m \in \mathbb{R}$ el estado, la salida y la entrada del modelo respectivamente. El vector $\hat{\theta}$ corresponde a un estimado de los parámetros θ del modelo (2.14).

Dado un conjunto de mediciones de señales del modelo (2.14), el problema de

identificación paramétrica se define como la estimación del vector de parámetros $\hat{\theta}$ a través de la minimización de un criterio de optimización que depende del error de modelo:

$$e_m = y_s - y_m \quad (2.16)$$

Una estructura usualmente utilizada para identificar parámetros es la configuración de modelos en paralelo [70]. La idea de esta aproximación es que la señal de excitación se aplica al sistema y al modelo. Dada una señal de excitación r , ésta puede aplicarse al sistema (2.14) siempre y cuando éste sea estable en lazo abierto, de forma que las entradas se definirían como $u_s = r$ y $u_m = r$ (ver Fig 2.2a). Si el sistema no es estable en lazo abierto es necesario aplicar una ley de control en realimentación como se observa en la Fig. 2.2b tal que el sistema en lazo cerrado sea estable.

Por ejemplo, si el estado x_s está disponible a través de mediciones, entonces la ley de control se propone como:

$$u_s = h(r, x_s) \quad (2.17)$$

la cual depende de la señal de excitación r .

Es importante mencionar que la ley de control también se aplica al modelo durante la simulación dinámica del mismo como se verá en párrafos subsecuentes. Esta ley tiene la misma estructura que la ley de control aplicada al sistema, es decir:

$$u_m = h_m(r, x_m) \quad (2.18)$$

en donde se emplea x_m en lugar de x_s .

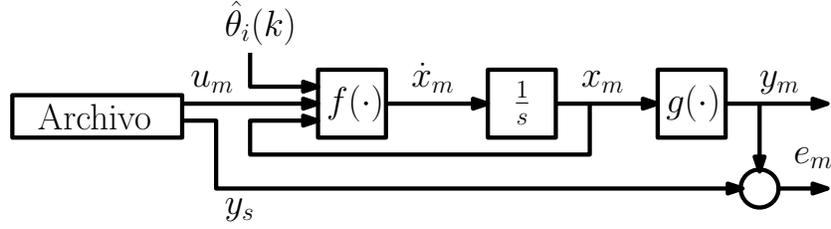
2.4. Implementación de la identificación de parámetros mediante el algoritmo PSO

La metodología utilizada para implementar el algoritmo PSO como identificador de parámetros consta de tres etapas:

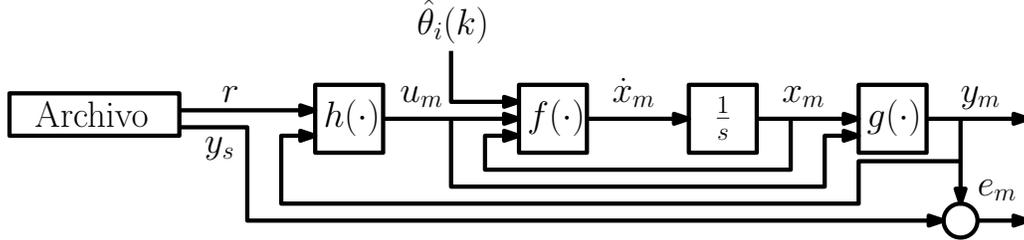
2.4.1. Adquisición de datos

Si el sistema a identificar es estable en lazo abierto se aplica una señal de excitación $r = u_s$ que alimente al sistema. Esta señal junto con la salida del mismo son almacenadas en un archivo para su uso posterior como se muestra en la Fig. 2.2a.

En el caso de que el sistema sea inestable se requiere emplear la ley de control (2.17) tal que el sistema sea estabilizado sin conocer previamente sus parámetros. Una vez implementada la ley de control debe aplicarse una señal de excitación r .



(a) Simulación del sistema dinámico estable en lazo abierto.



(b) Simulación del sistema dinámico sin estabilidad en lazo abierto y que requiere la implementación de una ley de control.

Figura 2.3: Simulación del sistema dinámico utilizando los estimados $\hat{\theta}_i(k)$

Esta señal junto con la señal de salida y_s se almacenan en un archivo de datos como se puede ver en la Fig. 2.2b.

2.4.2. Optimización numérica

El primer paso para ejecutar la optimización es emplear cotas conocidas a priori sobre los parámetros a identificar para expresarlas como restricciones. Estas permiten definir el conjunto de soluciones factibles Ω_s . Las cotas se pueden determinar a partir de las restricciones físicas del sistema. Por ejemplo, en el caso de muchos sistemas electromecánicos sus parámetros son positivos, hecho que permite establecer la restricción de que los parámetros a identificar sean siempre positivos.

El segundo paso consiste en proponer una función de desempeño cuya evaluación requiere que para cada partícula del algoritmo PSO se ejecute una simulación utilizando el modelo (2.15) como se puede ver en la Fig. 2.3. Note que la partícula $x_i(k)$ corresponde al vector de parámetros estimados $\hat{\theta}_i(k)$. Es importante mencionar que al modelo se aplica la referencia r adquirida en la etapa de adquisición de datos, por lo que es necesario que la simulación se ejecute con la misma frecuencia de muestreo que la adquisición de datos realizada. En el caso en que el modelo (2.15) sea inestable será necesario ejecutar la simulación dinámica aplicándole la ley de control (2.18) como se mencionó previamente.

Al ejecutar la simulación se almacenan los datos y_m para que se calcule el error e_m en cada instante de tiempo. Como ejemplo, la función de desempeño J se define

como:

$$J(\hat{\theta}_i(k)) = \sum_{j=1}^R |e_m(j)| \quad \hat{\theta}_i(k) \in \Omega_s \subset \mathbb{R}^m \quad R = \frac{T}{t_s} \quad (2.19)$$

donde $|\cdot|$ corresponde al valor absoluto, T es el segmento de tiempo durante el cual se llevó a cabo la simulación y t_s es el tiempo de muestreo utilizado.

Una vez que el algoritmo PSO detiene su operación por medio de alguno de los criterios de paro k_{max} ó α , la mejor solución obtenida por el mismo se considerará como el estimado $\hat{\theta}^*$, siendo esta la mejor solución para el problema de identificación de parámetros.

2.4.3. Validación de los parámetros estimados mediante la realización de análisis estadísticos

Con la finalidad de evaluar la variación de los parámetros estimados cuando se implementa un algoritmo metaheurístico es necesario ejecutar el proceso de identificación en múltiples ocasiones para verificar la repetibilidad de los resultados, debido a que se debe asegurar que aún con la presencia de términos estocásticos. A su vez, el implementar múltiples ejecuciones de un algoritmo bajo las mismas condiciones de operación permite implementar pruebas estadísticas no-paramétricas. Estas pruebas permiten determinar si existen diferencias significativas entre dos grupos de datos [71, 72] y con ello generar conclusiones respecto a los resultados.

Primeramente, para analizar el comportamiento de la variabilidad de los resultados se utilizará la desviación estándar σ la cual se define como:

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (m_i - \mu)^2} \quad (2.20)$$

donde m_i es un elemento del conjunto de muestras, N es el total de muestras en el conjunto y μ es el promedio de la muestras cuya definición es:

$$\mu = \frac{1}{N} \sum_{i=1}^N m_i \quad (2.21)$$

La desviación estándar σ representa la dispersión de los datos y en el caso de la identificación de parámetros es deseable que este valor sea pequeño. Lo que significa que si se ejecuta un algoritmo, éste estimará parámetros cercanos en todas las ocasiones a pesar de las señales estocásticas que afecten a los datos.

Por otro lado, las gráficas de caja se utilizan en Estadística Descriptiva para

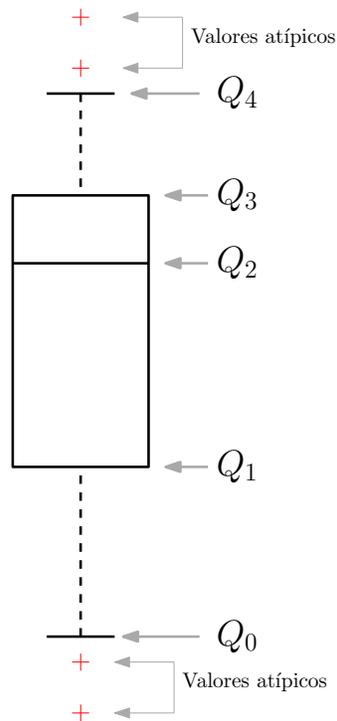


Figura 2.4: Diagrama de caja utilizado en estadística descriptiva.

visualizar la forma en que están distribuidos los datos. La estructura de estas gráficas se puede observar en la Fig. 2.4 en donde se observan los siguientes valores:

- Q_1 : Primer cuartil. El 25 % de los datos son menores a este valor.
- Q_2 : Segundo cuartil o mediana. El 50 % de los datos es menor que o igual a este valor.
- Q_3 : Tercer cuartil. El 75 % de los valores son menores o iguales a dicho valor.
- IQR : La distancia intercuartil se define como $IQR = Q_3 - Q_1$
- Q_0 : El cuartil cero corresponde a $Q_1 - 1.5 \times IQR$.
- Q_4 : El cuartil cuatro corresponde a $Q_3 + 1.5 \times IQR$.
- Valores atípicos: Aquellos valores superiores a Q_4 o inferiores a Q_0 .

Como se puede observar, se espera que la distancia IQR tenga el menor valor posible, lo que corresponde a una menor variabilidad posible. A su vez se espera que existan pocos o ningún valor atípico, lo que significaría que los datos se encuentran lo suficientemente compactos como grupo.

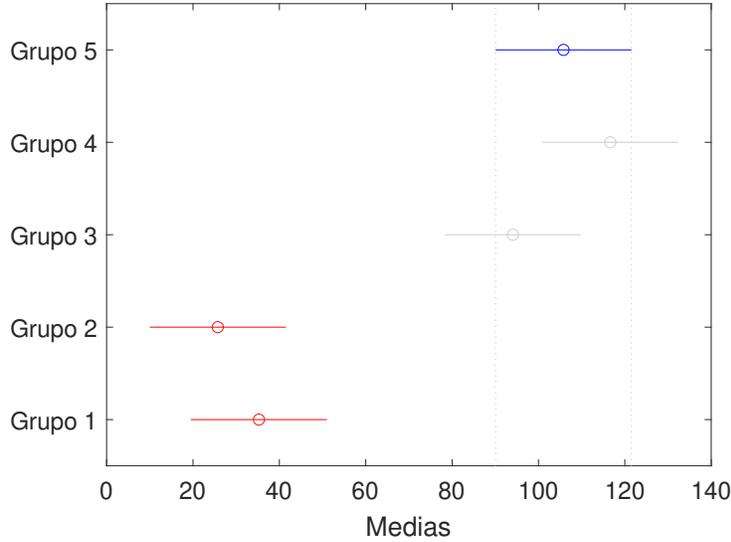


Figura 2.5: Ejemplo de una prueba de Bonferroni.

El uso de la estadística descriptiva no es suficiente para asegurar que existan diferencias significativas entre los dos grupos de resultados. También es necesario implementar una serie de pruebas de estadística no paramétrica para verificar estas aseveraciones. Para determina la existencia de dichas diferencias la prueba más simple es la prueba de Friedman [73], el cual arroja un *p-valor* que es la probabilidad de que dos o más conjuntos bajo análisis sean similares. Para verificar que existan diferencias significativas entre los grupos se requiere que $p - valor < 0.05$, esto implica que existe un 95 % de probabilidad de que estos sean diferentes.

La prueba de Friedman permite identificar si existen diferencias significativas entre varios grupos al mismo tiempo, pero no es capaz de determinar si las diferencias significativas se presentan entre todos los grupos o si sólo existen diferencias entre dos grupos particulares y los demás son similares. Por lo anterior, es necesario implementar una prueba adicional **post-hoc** que permite la comparación múltiple. Esta realiza una comparación entre las medias obtenidas por una prueba de ANOVA para determinar diferencias entre los grupos [72]. Las pruebas desarrolladas en este trabajo consideran primeramente la obtención de dichas medias mediante la prueba de Kruskal-Wallis para posteriormente aplicar la prueba de Bonferroni [74].

Un ejemplo del resultado de una prueba de Bonferroni se presenta en la Fig. 2.4 donde se puede notar que existen cinco grupos de datos y se desea saber si existen diferencias significativas entre ellos.

Como puede observarse en el ejemplo, en el eje horizontal se tienen los valores de las medias obtenidas por la prueba de Kruskal-Wallis. La anchura de los resultados

El error de seguimiento de trayectoria es $e_t = r - x_{s1}$ y $e^{(i)} = \frac{d^i e}{dt^i} = r^{(i)} - x_{s(i+1)}$, $i = 1, \dots, n - 1$.

Sustituyendo la ley de control (2.23) en (2.22) produce la dinámica del error siguiente:

$$e_t^{(n)} + a_{n-1}e_t^{(n-1)} + \dots + a_1\dot{e}_t + a_0e_t = 0 \quad (2.24)$$

donde $e_t^{(n)} = r^{(n)} - \dot{x}_{sn}$. Se seleccionan los coeficientes $a_{n-1}, a_{n-2}, \dots, a_1, a_0$ de tal manera que el punto de equilibrio de la dinámica del error $[e_t, \dot{e}_t, \dots, e_t^{(n-1)}]^T = [0, 0, \dots, 0]^T$ sea exponencialmente estable. Bajo estas condiciones el error de seguimiento e_t converge exponencialmente a cero.

En la práctica el vector θ generalmente es desconocido y consecuentemente es necesario calcular la ley de control (2.23) empleando los estimados $\hat{\theta}^*$ obtenidos mediante el algoritmo PSO:

$$u_s = \frac{1}{\hat{\theta}_2^*} \left(-f_1(x_s, \hat{\theta}_1^*) - r^{(n)} + a_{n-1}e_t^{(n-1)} + \dots + a_1\dot{e}_t + a_0e_t \right) \quad (2.25)$$

donde $\hat{\theta} = [\hat{\theta}_1^\top, \hat{\theta}_2]^\top$, $\hat{\theta}_1 \in \mathbb{R}^{m-1}$. Además, se hace la suposición de que el algoritmo de identificación produce estimados positivos, i.e. $\hat{\theta}_2 \in \mathbb{R}^+$.

Al realizar la substitución de (2.25) en (2.22) se obtiene la dinámica del error:

$$e_t^{(n)} + a_{n-1}e_t^{(n-1)} + \dots + a_1\dot{e}_t + a_0e_t = \eta \quad (2.26)$$

$$\eta = - \left(f_1(x_s, \hat{\theta}_1^*) - f_1(x_s, \theta_1) \right) - u \left(\hat{\theta}_2^* - \theta_2 \right) \quad (2.27)$$

Si $\hat{\theta}^* = \theta$, entonces la ecuación (2.26) corresponderá a (2.24), por lo que el error de seguimiento e_t convergerá exponencialmente a cero. De otra forma, el error de seguimiento de trayectoria convergerá a una vecindad cercana al punto de equilibrio $[e_t, \dot{e}_t, \dots, e_t^{(n-1)}]^T = [0, 0, \dots, 0]^T$ cuyo tamaño dependerá de la magnitud de η .

En consecuencia, el tamaño del error de de seguimiento e_t puede servir como medida de la cercanía de $\hat{\theta}^*$ respecto de θ . Un valor menor del error paramétrico $\hat{\theta}^* - \theta$ implicaría que la vecindad a la que converge e_t disminuye y por lo tanto un menor error de seguimiento se encontraría ligado a un menor error paramétrico.

Una manera de evaluar la pertinencia de los parámetros estimados $\hat{\theta}^*$ es mediante el cálculo de un conjunto de índices de desempeño que cuantifiquen el error de seguimiento de trayectoria e_t , la amplitud de la señal de control u_s y sus derivada respecto al tiempo. Estos índices dan una idea general de la calidad de los parámetros estimados y la presencia de grandes errores de seguimiento o bien, la existencia de castaño

en la señal de control implicarían que los parámetros estimados no son adecuados.

Para evaluar la calidad de seguimiento de la trayectoria se propone la Integral del Error Cuadrático (IEC) definida como:

$$IEC = \int_0^t e_t^2 dt \quad (2.28)$$

La evaluación de la magnitud de la señal de control en un periodo de tiempo se realiza mediante la Integral del valor Absoluto del Control (IAC):

$$IAC = \int_0^t |u_s| dt \quad (2.29)$$

Finalmente, para evaluar el fenómeno del castaño se utiliza la Integral de la variación del valor Absoluto de la Variación del Control (IAVC) definido como:

$$IAVC = \int_0^t \left| \frac{du_s(\tau)}{d\tau} \right| dt \quad (2.30)$$

Capítulo 3

Efecto de la Riqueza Espectral en la identificación de parámetros empleando el algoritmo PSO

La condición de Excitación Persistente (EP) y la Riqueza Espectral (RE) de las señales de excitación aplicadas con propósitos de identificación paramétrica han sido tópicos importantes en el análisis de los algoritmos clásicos de identificación de parámetros. Sin embargo, en el caso de los algoritmos metaheurísticos empleados en la identificación de parámetros como en el caso del algoritmo PSO, estos aspectos prácticamente no han recibido atención.

Debido a esta situación, en este capítulo se estudia la calidad de los parámetros estimados bajo diferentes condiciones de RE de la señal de excitación. Para ello se propone una metodología que permite estimar la RE en las señales de referencia aplicadas a un sistema dinámico. De forma adicional, se propone un nuevo algoritmo PSO el cual modifica su mecanismo de búsqueda con base a la RE de la señal de referencia. Esto permite una mejora en los parámetros estimados cuando el valor de la RE es bajo respecto a algoritmos PSO que no toman en cuenta este aspecto.

3.1. Condición de Excitación Persistente (EP) en los algoritmos clásicos de identificación de parámetros

La identificación de parámetros suele realizarse utilizando algoritmos basados en regresiones lineales como es el caso del algoritmo de Mínimos Cuadrados (MC) [15, 14]. Un ejemplo de regresión lineal es:

$$y_s = \theta^T \phi \tag{3.1}$$

donde el término $\phi \in \mathbb{R}^n$ es llamado el vector regresor y se encuentra compuesto por señales que se obtienen a través de mediciones. Por otra parte, el vector $\theta \in \mathbb{R}^n$ corresponde a los parámetros desconocidos que se suponen constantes. Finalmente, el término y_s es la salida del sistema que se encuentra bajo identificación.

Para ilustrar la importancia de los conceptos de EP y RE en los algoritmos de identificación basados en regresiones lineales, considérese ahora el caso de los algoritmos de identificación recursivos los cuales producen un estimado variante en el tiempo $\hat{\theta}(t) \in \mathbb{R}^n$ en el tiempo de θ . Entonces, la convergencia de $\hat{\theta}(t)$ hacia θ es asegurada si la siguiente condición de EP se cumple [17]:

$$\sigma_2 I \geq \int_{t_0}^{t_0+\delta} \phi(\tau)\phi^\top(\tau)d\tau \geq \sigma_1 I \quad \forall \quad t_0 > 0 \quad (3.2)$$

donde σ_1 , σ_2 y δ son constantes positivas. Esta condición implica que la matriz producida por la integral de $\phi(\tau)\phi^\top(\tau)$ respecto al tiempo es definida positiva para cualquier intervalo δ .

Es importante mencionar que la condición (3.2) debe cumplirse no sólo en el caso de los algoritmos de MC, sino también en otros algoritmos de identificación paramétrica incluidos aquellos empleados dentro de controladores adaptables. Sin embargo, no se encuentra directamente relacionada con las propiedades espectrales que la señal de excitación r debe cumplir para asegurar la convergencia de los parámetros estimados $\hat{\theta}(t)$ hacia los parámetros reales θ en un esquema de identificación paramétrica.

En el caso de los sistemas Lineales Invariantes en Tiempo (LIT), existen condiciones en el dominio de la frecuencia para la señal de excitación r que aseguran la condición EP del regresor ϕ , tal como se describen a continuación [17, 70].

Definición 1 Sea $R(t)_r$ la auto-covarianza de una señal $r \in \mathbb{R}$, entonces r es estacionaria si el siguiente límite es satisfecho uniformemente en τ_0 :

$$R_r(t) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{\tau_0}^{\tau_0+T} r(\alpha)r(t + \alpha)d\alpha \quad (3.3)$$

La Medición Espectral $S_r(\omega)$ de r se define como la transformada de Fourier de $R_r(t)$:

$$S_r(\omega) = \int_{-\infty}^{\infty} e^{-j\omega\alpha} R_r(\tau)d\alpha \quad (3.4)$$

Definición 2 Sea $r \in \mathbb{R}$ una señal estacionaria, entonces r es llamada de Suficiente Riqueza Espectral (SRE) de orden n si su Medición Espectral $S_r(\omega)$ tiene al menos n puntos de soporte.

Por ejemplo, la señal:

$$\sigma = \sum_{i=1}^m B_i \sin(\omega_i t) \quad (3.5)$$

con $\omega_i \neq \omega_j, i \neq j$, es de SRE de orden $2n$ porque contiene n frecuencias y su Medición Espectral contiene al menos $2m$ puntos de soporte. Cada uno de los términos $\sin(\omega_i t)$ en la sumatoria presenta dos puntos de soporte, uno en $-\omega_i$ y otro en ω_i .

En la mayor parte de los esquemas de control adaptable es posible relacionar el vector regresor ϕ con la señal de excitación r mediante el uso de una función de transferencia $H(s)$ que sea estable y propia:

$$\mathcal{L}\{\phi\} = H(s)\mathcal{L}\{r\} \quad (3.6)$$

donde la notación $\mathcal{L}\{\cdot\}$ es el operador de Laplace con $s \in \mathbb{C}$ un número complejo.

La expresión presentada anteriormente permite establecer una relación entre la condición EP de ϕ y la condición de SRE para r establecida en el siguiente teorema:

Teorema 1 [75] *Considérese (3.6) con $r \in \mathbb{R}$, $\phi \in \mathbb{R}^n$. Se supone que los términos $H(j\omega_1) \dots H(j\omega_n)$ son linealmente independientes para cada $\omega_1 \dots \omega_n \in \mathbb{R}, \omega_i \neq \omega_j, i \neq j$. Entonces, r es SRE de orden n si y sólo si ϕ cumple con la condición de EP.*

De acuerdo al resultado anterior, una señal de excitación r con una SRE de orden n produce un vector regresor ϕ que cumple con la condición EP, lo cual asegura la convergencia del vector de parámetros estimados $\hat{\theta}$ al valor real de θ [17].

3.2. Medición de la Riqueza Espectral de las señales de excitación en la identificación de parámetros

Los algoritmos metaheurísticos resuelven el problema de identificación paramétrica sin emplear regresiones lineales. En consecuencia, no es posible establecer una relación analítica similar a la descrita en el Teorema 1, entre el contenido espectral de la señal de excitación r y la convergencia de los parámetros estimados. Por ello es necesario analizar de manera heurística la relación entre la calidad de los parámetros estimados obtenidos mediante un algoritmo metaheurístico y la RE de la señal de excitación r .

La RE de la señal r se relaciona con el número de frecuencias que contiene su espectro, y cuando se aplica a un sistema dinámico la salida de éste puede tener un espectro diferente al de la señal de entrada como consecuencia de los modos del sistema que son excitados [76].

Para evaluar si los datos obtenidos mediante mediciones de entrada-salida de un sistema son útiles para un proceso de identificación de parámetros empleando un algoritmo PSO se propone el procedimiento siguiente. Primero se aplican señales de excitación al sistema que contengan valores diferentes de RE, lo cual produce un conjunto de datos para cada señal. Empleando cada conjunto se implementa un proceso de identificación paramétrica mediante un algoritmo PSO. Posteriormente se evalúa la calidad de los parámetros estimados tanto desde el punto de vista estadístico como de su pertinencia al ser aplicados a una ley de control. Lo anterior permite evaluarlos en función de la RE de la señal de excitación r .

Para aplicar el procedimiento anterior es necesario calcular el valor de RE de las señales de excitación r empleadas. Lo anterior se puede llevar a cabo mediante la definición de Medición Espectral $S_r(\omega)$ (3.4). Sin embargo, su cálculo analítico es posible sólo en casos simples. De forma alternativa, las componentes de frecuencia de r pueden obtenerse numéricamente mediante el uso de la Transformada Rápida de Fourier (Fast Fourier Transform-FFT) [77]. Esta familia de algoritmos dependen de la Transformada Discreta de Fourier y obtienen la amplitud del conjunto de señales senoidales que conforman una señal.

Defínase el conjunto de frecuencias $\Psi = [0, \psi]$ asociado a las componentes de frecuencia sinusoidales de la señal r que se pueden calcular computacionalmente. El término $\psi = f_s/2$ es la cota superior del ancho de banda del algoritmo FFT, siendo f_s la frecuencia de muestreo utilizada para la adquisición de la señal de excitación r . Lo anterior permite definir el conjunto:

$$\Sigma_r(\omega) = \{\Sigma_{r1}(\omega_1), \dots, \Sigma_{rm}(\omega_m)\}, \omega_i \in \Psi \quad (3.7)$$

que corresponde al espectro de la señal r producido por el algoritmo FFT. Cada elemento $\Sigma_{ri}(\omega_i)$ es la amplitud de la señal sinusoidal de frecuencia ω_i . Por lo tanto, el estimado \hat{r} de la señal r se calcula como:

$$\hat{r} = \sum_{i=1}^m \Sigma_{ri}(\omega_i) \sin \omega_i t \quad (3.8)$$

donde la Medición Espectral $S_{\hat{r}}(\omega)$ de \hat{r} tiene $2m$ puntos de soporte.

Por otra parte, al escoger la señal de excitación r se deben comparar varias señales y con el fin de asegurar una comparación justa entre ellas se normalizan sus espectros

de la manera siguiente:

$$\bar{\Sigma}_r(\omega) = \{\bar{\Sigma}_{r1}(\omega_1), \dots, \bar{\Sigma}_{rm}(\omega_m)\}, \omega_i \in \Psi \quad (3.9)$$

$$\bar{\Sigma}_{ri}(\omega_i) = \frac{\Sigma_{ri}(\omega_i)}{\max_{\omega_i \in \Psi}(\Sigma_{ri}(\omega_i))} \quad (3.10)$$

Es necesario notar que la FFT produce estimaciones numéricas discretas del espectro de la señal. Por esta razón se espera que existan residuos de líneas espectrales que deben ser descartados. Con esta finalidad se define el conjunto:

$$\tilde{\Sigma}_r(\omega) = \{\tilde{\Sigma}_{r1}(\omega_1), \dots, \tilde{\Sigma}_{r\chi}(\omega_\chi)\}, \omega_i \in \Psi \quad (3.11)$$

$$\tilde{\Sigma}_{ri}(\omega_i) = \begin{cases} 0 & \text{if } \bar{\Sigma}_{ri}(\omega_i) < \kappa \\ 1 & \text{if } \bar{\Sigma}_{ri}(\omega_i) \geq \kappa \end{cases} \quad (3.12)$$

donde el valor de umbral κ es definido como [78]:

$$\kappa = 1 - \pi Z \cot(\pi Z) \quad (3.13)$$

con $Z = \psi/N$ siendo N el número de muestras que componen a la señal r . Por lo tanto, $\tilde{\Sigma}_r(\omega)$ es llamado el espectro κ -normalizado de r y contiene $\chi \leq m$ líneas espectrales cuya magnitud es superior al umbral κ .

Lo anterior permite establecer la definición siguiente:

Definición 3 *La Riqueza Espectral (RE) de una señal r es la cardinalidad $\chi = |\tilde{\Sigma}_r(\omega)|$ del conjunto $\tilde{\Sigma}_r(\omega)$.*

Dado que la medición espectral de la señal \hat{r} es de 2χ , sólo el valor χ será utilizado para evaluar la RE de las señales de excitación utilizadas en los algoritmos de identificación paramétrica.

3.3. Algoritmo PSO-RE enfocado a la identificación de parámetros en sistemas dinámicos

Con la finalidad de disminuir la variabilidad de los parámetros estimados cuando se utilizan señales que presentan una RE baja, se propone un nuevo algoritmo PSO basado en la Riqueza Espectral de la señal de excitación r que se denominará en lo sucesivo como PSO-RE. Este algoritmo emplea el valor de la cardinalidad χ del conjunto $\tilde{\Sigma}_r(\omega)$. Los siguientes párrafos describen el algoritmo.

La dinámica de las partículas en este algoritmo es la siguiente:

$$\begin{aligned}
\omega_i(k+1) &= \omega_i(k) + \phi_1(k)p(\chi, k)(x_{i,*}(k) - x_i(k)) \\
&\quad + \phi_2(k)l(\chi, k)(x_*(k) - x_i(k)) \\
x_i(k+1) &= x_i(k) + \omega_i(k+1)
\end{aligned} \tag{3.14}$$

donde los términos $p(\chi, k)$ y $l(\chi, k)$ son ajustados mediante el valor χ y el número de iteraciones k que ha realizado el algoritmo. Debe hacerse notar que χ debe ser calculado previamente a la implementación del algoritmo PSO-RE como se describió en la subsección anterior.

El término $p(\chi, k)$ se encuentra asociado a la ponderación que se le da a las mejores posiciones locales de las partículas y se expresa como:

$$\begin{aligned}
p(\chi, k) &= \bar{p} + \left(\frac{\bar{p}}{(1 + e^{-\chi})} \right) \left(\frac{1}{1 + e^{-w_\chi(k-0.5k_{\max})}} \right) \\
&\quad + \frac{-\bar{p}p_c}{1 + e^{-w_\chi(k-0.5k_{\max})}}
\end{aligned} \tag{3.15}$$

donde \bar{p} es una constante positiva que determina el máximo valor que puede obtener $p(\chi, k)$, la constante p_c es un valor de ponderación positivo y w_χ es la ponderación que se da a la riqueza espectral. La constante k_{\max} es el número de iteraciones máximo que ejecuta el algoritmo.

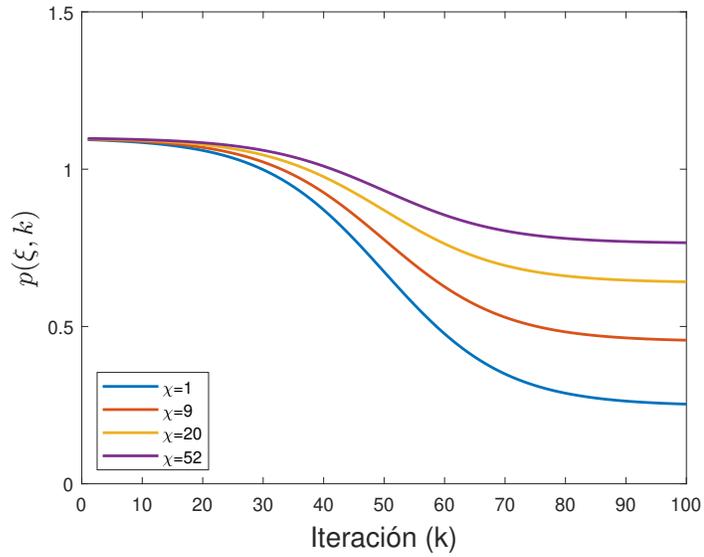
El término $l(\chi, k)$ es calculado como:

$$l(\chi, k) = \frac{\bar{l}e^{-\chi}}{1 + e^{-w_\chi(k-0.5k_{\max})}} + l_{\min} \tag{3.16}$$

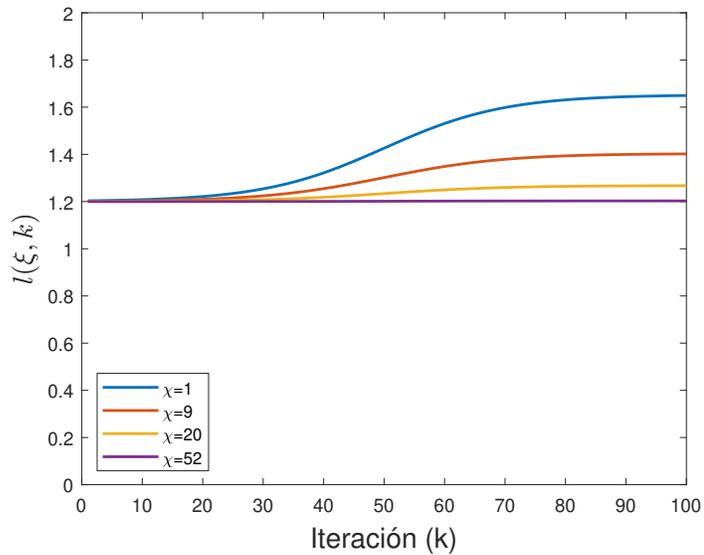
donde $l_{\min} \in \mathbb{R}^+$ es el mínimo valor que obtendrá $l(\chi, k)$ y \bar{l} determina el comportamiento de $l(\chi, k)$ cuando la señal de excitación utilizada presenta un valor pequeño de χ .

La Fig. 3.1 muestra la relación entre χ y los términos $p(\chi, k)$ y $l(\chi, k)$. Si la señal de excitación tiene un número de líneas espectrales alto, por ejemplo $\chi = 52$, ambas funciones permanecerán constantes cuando el número de iteraciones k se incremente. En el caso en que la señal de excitación posea un valor pequeño de χ , cuando el número de iteraciones k se incrementa el valor de $p(\chi, k)$ disminuye mientras que, el valor de $l(\chi, k)$ aumenta.

La idea principal del algoritmo PSO-RE es la reducción de la varianza de las soluciones cuando la señal r que excita al sistema tiene un número de líneas espectrales χ reducido. En consecuencia, si χ es pequeño es necesario forzar la convergencia del



(a) Comportamiento del término $p(\chi, k)$ respecto a la riqueza espectral de la señal χ y al número de iteraciones k .



(b) Comportamiento del término $l(\chi, k)$ respecto a la riqueza espectral de la señal χ y al número de iteraciones k .

Figura 3.1: Evolución dinámica de los términos $p(\chi, k)$ y $l(\chi, k)$.

algoritmo hacia una única solución. Con este fin, cuando el número de iteraciones k se incrementa el peso de las soluciones locales disminuye al mismo tiempo que se incrementa el peso que se le da a la mejor solución global. Este comportamiento produce la convergencia hacia una solución global. Por otra parte, si χ es elevado el

comportamiento del algoritmo PSO-RE es similar al de un algoritmo PSO clásico.

Capítulo 4

Prueba del algoritmo PSO-RE mediante un caso de estudio

El algoritmo PSO-RE presentado previamente se prueba mediante un caso de estudio que consiste en la identificación de parámetros de un modelo de un servomecanismo mediante datos experimentales generados empleando diferentes señales de excitación. Los resultados obtenidos con el algoritmo PSO-RE se comparan con aquellos obtenidos con varios algoritmos PSO publicados previamente. La pertinencia de los parámetros estimados se evalúa desde el punto de vista estadístico y mediante su empleo en el cálculo de una ley de control aplicada al servomecanismo la cual es implementada y validada experimentalmente.

4.1. Preliminares

4.1.1. Modelo del servomecanismo

La planta a identificar es un servomecanismo de Corriente Directa (DC) actuado mediante un amplificador electrónico de potencia. El modelo no lineal de dicho sistema es [20]:

$$\ddot{q} = -a\dot{q} + bu - c\text{sign}(\dot{q}) + d \quad (4.1)$$

donde q , \dot{q} y \ddot{q} son la posición, velocidad y aceleración angular respectivamente, mientras que u es la señal de control aplicada. El término a se relaciona a la fricción viscosa, b es la ganancia de entrada, el término c está relacionado a la fricción de Coulomb, $\text{sign}(\cdot)$ es la función signo y d es una perturbación constante generada por los voltajes parásitos en el amplificador de potencia. Se supone que los parámetros a , b , c y d son desconocidos, positivos y constantes.

Este sistema electromecánico no es estable en lazo abierto por lo que se estabiliza por medio de un controlador de realimentación Proporcional Derivativo (PD):

$$u = k_p e - k_d \dot{q} \quad (4.2)$$

donde el error de posición se define como $e = r - q$. Las constantes $k_p, k_d \in \mathbb{R}^+$ son las ganancias proporcional y derivativa respectivamente.

4.1.2. Algoritmos de identificación

La identificación de parámetros del modelo (4.1) se lleva a cabo mediante el algoritmo PSO-RE y para propósitos de comparación también se emplean los siguientes algoritmos PSO: PSO clásico (PSO-C) [10], PSO con Ponderación de Velocidad Variable (PSO-PVV) [66], PSO con Factor de Constricción Variable (PSO-FCV) [67] y PSO de Orden Fraccionario Regulado (PSO-OFR) [68].

Para este problema es necesario estabilizar el servomecanismo mediante el uso de (4.2) con lo cual se puede realizar la adquisición de datos como se presenta en la Fig. 4.1. Mediante el archivo de datos obtenido en esta etapa se puede realizar el proceso de identificación de parámetros utilizando los algoritmos PSO. Es importante mencionar que el proceso de identificación mediante algoritmos metaheurísticos considera una simulación dinámica del sistema, en el caso del servomecanismo esta simulación se presenta en la Fig. 4.2. El modelo del servomecanismo utilizado corresponde a:

$$\ddot{q}_m = -\hat{a}\dot{q}_m + \hat{b}u_m - \hat{c}\text{sign}(\dot{q}_m) + \hat{d} \quad (4.3)$$

donde la señal de control u_m se define como:

$$u_m = k_p(r - q_m) - k_d\dot{q}_m \quad (4.4)$$

Los valores de las ganancias k_p y k_d son los mismos que los que se utilizan en el controlador (4.2).

Adicionalmente, se plantea la implementación del algoritmo CLIE (Closed Loop Input Error) (CLIE) [20] el cual es un estimador de parámetros basado en una regresión lineal. Esto permite tener un grupo de parámetros estimados del servomecanismo obtenidos mediante una técnica tradicional con el fin de realizar comparaciones con los métodos que emplean el algoritmo PSO.

La forma en que se implementa el algoritmo CLIE se presenta en la Fig. 4.3. Se puede notar que esta implementación considera la existencia de un controlador PD que genera una señal de control u aplicada al servomecanismo. De manera simultánea, se

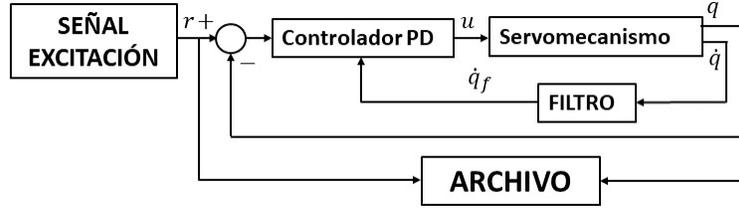


Figura 4.1: Adquisición de datos experimentales para el caso del servomecanismo con la finalidad de emplearlos en la identificación de parámetros mediante algoritmos PSO.

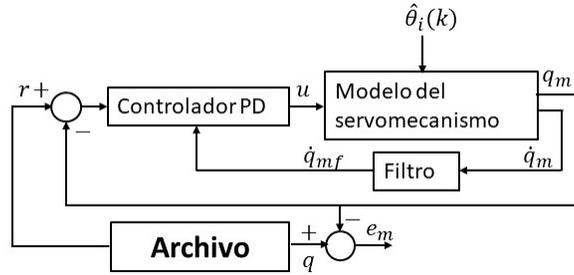


Figura 4.2: Simulación dinámica del servomecanismo utilizando datos experimentales realizada durante el proceso de identificación de parámetros mediante algoritmos PSO.

realiza una simulación dinámica en tiempo real del modelo del servomecanismo donde se utilizan los parámetros estimados. En el modelo del servomecanismo el controlador PD genera la señal de control simulada u_e misma que se utiliza en el algoritmo de identificación de parámetros.

El algoritmo CLIE produce los siguientes parámetros estimados $\hat{\theta}_{\text{clie}}$ de θ con:

$$\theta = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}; \hat{\theta}_{\text{clie}} = \begin{bmatrix} \hat{a}_{\text{clie}} \\ \hat{b}_{\text{clie}} \\ \hat{c}_{\text{clie}} \\ \hat{d}_{\text{clie}} \end{bmatrix} \quad (4.5)$$

donde el vector de parámetros estimados $\hat{\theta}_{\text{clie}}$ se obtiene mediante la siguiente ecuación diferencial:

$$\dot{\hat{\theta}}_{\text{clie}} = -\Gamma \phi \epsilon_u \quad (4.6)$$

El término $\epsilon_u = u_e - u$ es el error en la entrada entre el servomecanismo y su modelo, y la matriz $\Gamma \in \mathbb{R}^{4 \times 4}$ es diagonal con términos positivos tal que $\Gamma > 0$. La convergencia de $\hat{\theta}_{\text{clie}}$ hacia θ se asegura si ϕ cumple la condición de EP (3.2). El

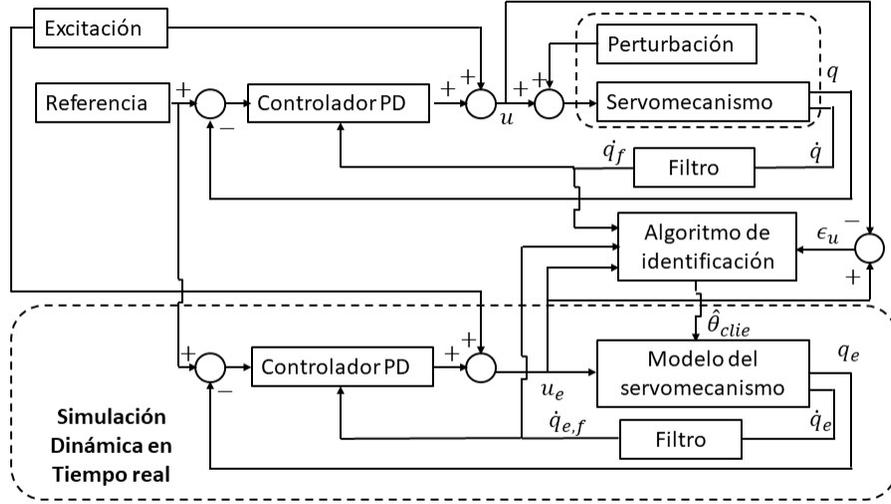


Figura 4.3: Implementación del algoritmo CLIE para la identificación de parámetros del servomecanismo.

vector regresor ϕ es definido como:

$$\phi = \begin{bmatrix} \dot{q}_e \\ -u_e \\ \text{sign}(\dot{q}) \\ -1 \end{bmatrix} \quad (4.7)$$

4.1.3. Planteamiento del problema de optimización

Para plantear la identificación de los parámetros a , b , c y d como un problema de optimización es necesario proponer la función de desempeño que se utilizará dentro del algoritmo PSO para cuantificar la validez de una solución, esta es propuesta como:

$$J(\hat{\theta}_i(k)) = \int_0^T |e_m(\tau)| d\tau; \hat{\theta}_i(k) \in \Omega_s \subset \mathbb{R}^m \quad (4.8)$$

donde Ω_s corresponde al conjunto definido mediante las restricciones sobre los valores que pueden tomar los parámetros del servomecanismo. Basados en las propiedades del modelo (4.1) los parámetros a , b y c son positivos y el término d asociado a la perturbación es constante. Lo anterior permite proponer las siguientes restricciones

que definen al conjunto Ω_s :

$$\begin{aligned}
 g_1 & : 0 < \hat{a} \\
 g_2 & : 0 < \hat{b} \\
 g_3 & : 0 < \hat{c} \\
 g_4 & : d_{\min} < \hat{d} < d_{\max}
 \end{aligned} \tag{4.9}$$

Debe mencionarse que el parámetro d se encuentra limitado por d_{\min} y d_{\max} debido a que el término corresponde a una perturbación constante acotada, por lo que estos valores son propuestos por el usuario. En el caso del servomecanismo las cotas propuestas son $d_{\min} = -10$ y $d_{\max} = 10$.

Notar que en el algoritmo PSO cada solución corresponde a un vector de parámetros estimados, es decir $x_* = \hat{\theta}_*$ el cual se define como:

$$\tilde{\theta}_* = \begin{bmatrix} \hat{a}_* \\ \hat{b}_* \\ \hat{c}_* \\ \hat{d}_* \end{bmatrix} \tag{4.10}$$

4.1.4. Planteamiento experimental

Para llevar a cabo el análisis del efecto de la RE en la identificación de parámetros se utilizó una plataforma experimental que se muestra en la Fig. 4.4. Esta se encuentra compuesta de un servomotor de DC modelo C34-L80-W40 conectado a un disco de inercia y un amortiguador magneto-reológico MRB-2107-3 de la compañía Lord, cuyo amortiguamiento se ajusta mediante un controlador Wonderbox del mismo fabricante que se encuentra deshabilitado durante todos los experimentos. La posición angular q del motor es medida utilizando un codificador óptico modelo ST50 con una resolución de 1024 PPR, y la velocidad angular \dot{q} es obtenida mediante un tacómetro modelo SA-7388F-1, ambos de la compañía Servotek. La medición de la velocidad angular se procesa mediante el siguiente filtro pasa-bajas:

$$G(s) = \frac{300}{s + 300} \tag{4.11}$$

el cual permite atenuar el ruido de medición de alta frecuencia. Debe mencionarse que este filtrado también se ha incluido en el esquema del sistema dinámico que será simulado para la implementación del algoritmo PSO como se muestra en la Fig. 4.2.

El algoritmo CLIE y la adquisición de datos experimentales empleada en los algo-

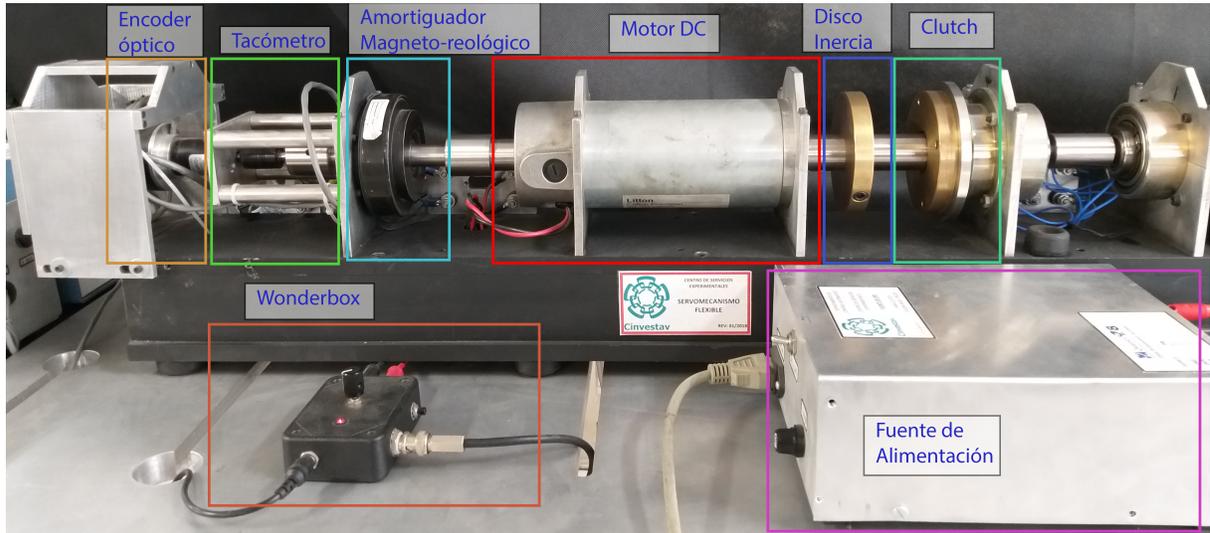


Figura 4.4: Plataforma experimental con amortiguador magneto-reológico.

ritmos PSO se han llevado a cabo utilizando el programa Matlab/Simulink R2011b de la compañía Mathworks y el programa Quarc de Quanser Consulting. Los programas se ejecutan en una computadora personal de 32 bits con un procesador Intel Core 2 con el sistema operativo Windows 7. Una tarjeta Q8 de Quanser Consulting permite la adquisición de datos.

El algoritmo CLIE se ejecuta empleando la ley de control (4.2) aplicada al servomecanismo y la ley de control PD (4.4) aplicada al modelo (4.3). Ambas leyes de control utilizan las ganancias $k_p = 34$ y $k_d = 0.65$. La ley de adaptación para el algoritmo CLIE emplea una matriz de ganancias $\Gamma = \text{diag}[3, 10, 1, 2]$. Cabe mencionar que el mismo algoritmo PD con las mismas ganancias se empleó en la adquisición de datos utilizada en los algoritmos PSO, con un tiempo de muestreo de 0.001s con un método numérico de integración de Euler.

Los algoritmos PSO implementados son codificados y ejecutados usando el programa Matlab 2019a ejecutado en un procesador de 64 bits AMD Ryzen 1600X. La sintonización de los algoritmos se ha realizado mediante la herramienta IRACE, la cual es implementada en el programa RStudio. El paquete IRACE [79] permite una sintonización automática de cada uno de los algoritmos PSO para asegurar que el estudio comparativo que se lleva a cabo entre los resultados sea justo. Aún más, la simulación dinámica del servomecanismo es realizada mediante el método numérico de Runge-Kutta con un tamaño de paso de 0.001s.

Todos los algoritmos PSO tienen $m = 20$ partículas y emplean un número máximo de iteraciones $k_{\max} = 200$. Los parámetros de los algoritmos implementados se encuentran en la Tabla 4.1. Para evaluar la repetibilidad de los resultados cada algo-

Algoritmo PSO	Parámetros
PSO	$p = 1.2617, l = 1.2447$
PSO-PVV	$p = 2.235, l = 1.3541,$ $\beta_{\max} = 0.98, \beta_{\min} = 0.62$
PSO-FCV	$p = 2.321, l = 1.8543,$ $\gamma_{\max} = 0.95, \gamma_{\min} = 0.70$
PSO-OFR	$p_{\min} = 1.253, p_{\max} = 1.874,$ $l_{\min} = 1.145, l_{\max} = 1.745$
PSO-RE	$\bar{p} = 1.627, w_{\chi} = 0.1,$ $l_{\min} = 1.654, \bar{l} = 0.62$ $p_c = 1.3$

Tabla 4.1: Parámetros de las variantes de PSO.

ritmo es ejecutado en 30 ocasiones para cada una de las señales de excitación descritas en los párrafos subsecuentes.

4.1.5. Señales de excitación

Las señales de excitación aplicadas son procesadas para obtener su espectro $\tilde{\Sigma}_r(\omega)$ utilizando una cota de $\kappa = 0.01$ calculada de acuerdo a (3.13).

Para impedir que algunos cambios drásticos en la dirección del motor se lleven a cabo y puedan producir desgaste mecánico, algunas de las señales de excitación son procesadas mediante el filtro pasa-bajas $F(s)$:

$$F(s) = \frac{10}{s + 10} \quad (4.12)$$

Las señales de excitación que se aplican son las siguientes:

- Onda senoidal: La onda senoidal mostrada en la Fig. 4.5a produce una excitación periódica y posee sólo una componente de frecuencia. Esta señal es generada mediante el bloque Signal Generator del entorno de Simulink con una frecuencia de 0.2 Hz y una amplitud de 1.5 radianes. Como se espera, en la Fig. 4.6a existe una única componente espectral.
- Onda cuadrada filtrada: La onda cuadrada filtrada mostrada en la Fig. 4.5b también proporciona una excitación periódica de la referencia y es generada mediante el bloque Signal Generator del entorno de Simulink con una frecuencia de 0.2 Hz y una amplitud de 1.5 radianes. Dicha señal es procesada mediante

el filtro 4.12. En la Fig. 4.6b se muestra que su espectro posee una componente frecuencial dominante y varias componentes armónicas con un menor valor.

- Señal caótica: Para generar esta señal se implementaron un par de osciladores de Duffing funcionando en régimen caótico mostrado en la Fig. 4.5c y cuyas dinámicas están descritas por [42]:

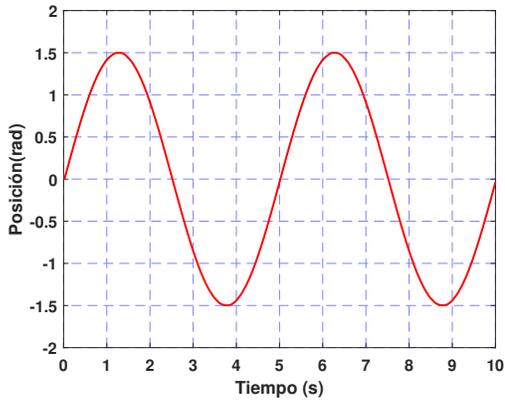
$$\begin{aligned}
 \dot{z}_1 &= \sigma_1 z_2 \\
 \dot{z}_2 &= \sigma_1 (z_1 - z_1^3) - 0.25 z_2 + \frac{1}{3} \sin(\sigma_1 t) \\
 \dot{z}_3 &= \sigma_2 z_4 \\
 \dot{z}_4 &= \sigma_2 (z_3 - z_3^3) - 0.25 z_4 + \frac{1}{3} \sin(\sigma_2 t) \\
 r &= A_1 z_1 + A_2 z_3
 \end{aligned} \tag{4.13}$$

donde $\sigma_1 = 3$, $\sigma_2 = 10$, $A_1 = 1$, $A_2 = 0.5$ considerando $z_1(0) = 0$, $z_2(0) = 0$, $z_3(0) = 0$ y $z_4(0) = 0$. Esta señal produce cambios suaves y con componentes de baja frecuencia como se puede observar en la Fig. 4.6c.

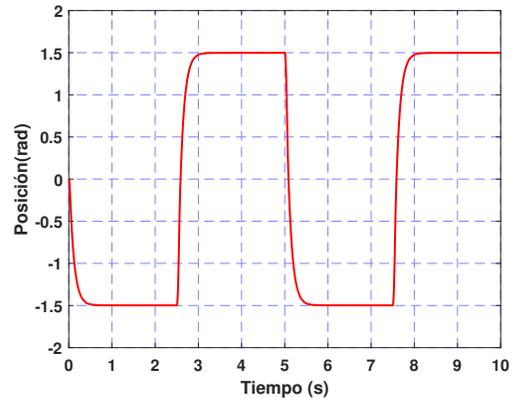
- Ruido blanco filtrado: Esta señal se genera mediante el bloque Band-Limited White Noise en el entorno de Simulink con una potencia de ruido de 0.1, un tiempo de muestreo de 0.01s y la aplicación del filtro 4.12 para generar la señal mostrada en la Fig. 4.5d. La Fig. 4.6d muestra que es la señal que tiene una mayor cantidad de componentes de frecuencia en comparación con a las anteriores.

La Riqueza Espectral χ de las señales para estas cuatro señales de excitación se presenta en la Tabla 4.2. Estos resultados muestran que las señales de onda senoidal y onda cuadrada filtrada son las que contienen un menor número de componentes de frecuencia en comparación con la señal caótica y el ruido blanco filtrado.

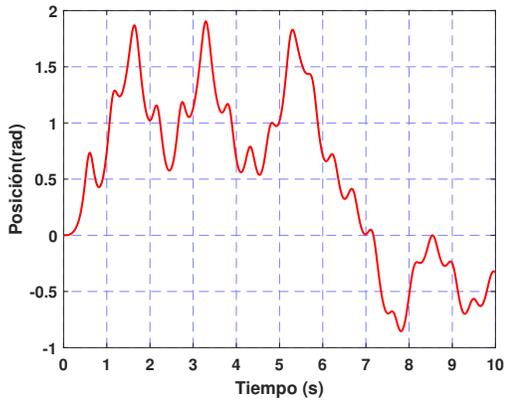
También debe ser mencionado que el espectro normalizado $\tilde{\Sigma}_r(\omega)$ de la señal de onda senoidal en la Fig. 4.6a contiene una sola frecuencia dominante y múltiples líneas espectrales con valores pequeños las cuales son consecuencia de la naturaleza numérica del algoritmo FFT. Esto muestra la necesidad de la aplicación de un umbral κ para obtener el espectro κ -normalizado de $\tilde{\Sigma}_r(\omega)$ tal que se descarten aquellas líneas espectrales que estén debajo del umbral.



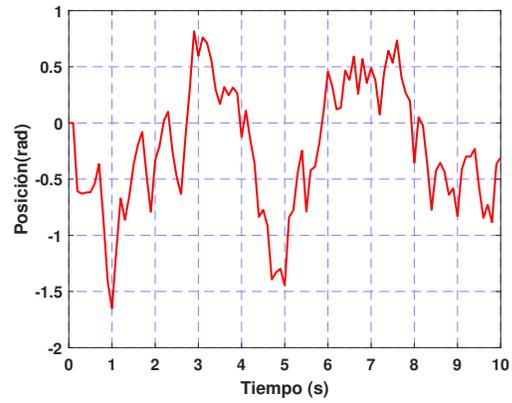
(a) Onda senoidal.



(b) Onda cuadrada filtrada.

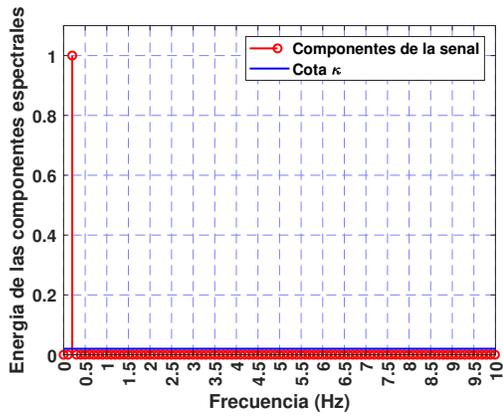


(c) Señal caótica.

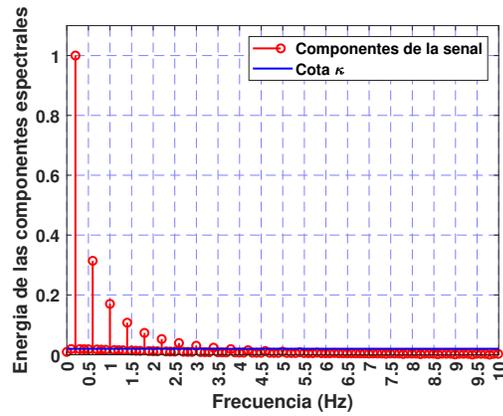


(d) Señal de ruido blanco filtrado.

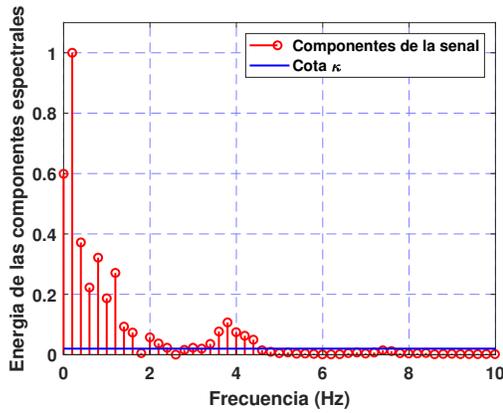
Figura 4.5: Señales de excitación aplicadas al servomecanismo.



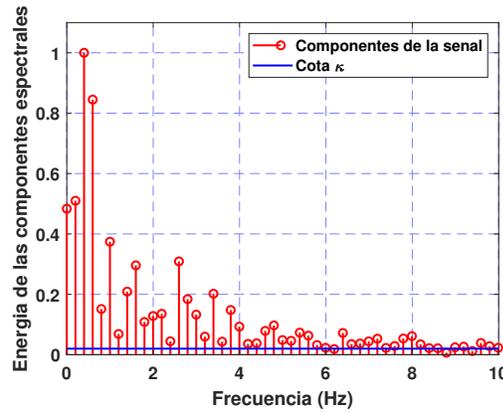
(a) FFT obtenida a partir de la onda senoidal.



(b) FFT obtenida a partir de una onda cuadrada filtrada.



(c) FFT obtenida a partir de la señal caótica.



(d) FFT obtenida a partir de ruido blanco filtrado.

Figura 4.6: Transformada Rápida de Fourier de las señales de excitación.

Señales de excitación	χ
Onda senoidal	1
Onda cuadrada filtrada	9
Señal caótica	20
Ruido blanco filtrado	52

Tabla 4.2: Riqueza Espectral χ de las señales aplicadas para la excitación del sistema en el proceso de identificación de parámetros.

4.2. Análisis de los parámetros identificados

Los parámetros identificados del servomecanismo obtenidos por los algoritmos PSO son mostrados en la Tabla 4.3, donde se presenta la mediana de los valores obtenidos mediante los 30 experimentos realizados. El desempeño del índice J mostrado en la misma tabla es la mediana de los valores obtenidos en 30 ejecuciones para cada combinación de algoritmo PSO con señal de excitación. Dentro de la misma tabla se presentan los valores de la desviación estándar σ de cada uno de los parámetros estimados.

4.2.1. Estadística no paramétrica

Con la finalidad de verificar que los algoritmos PSO generan resultados diferentes para diferentes señales de excitación y que los cambios observados no se deben a la naturaleza aleatoria de los algoritmos PSO que se implementan, es necesario realizar pruebas estadísticas no-paramétricas.

La primera prueba aplicada corresponde la prueba de Friedman sobre los valores de las funciones de desempeño obtenidas por los algoritmos metaheurísticos. Esta prueba evaluará si estos valores se modifican basándose en la Riqueza Espectral χ de la señal de excitación aplicada. Al ejecutarse la prueba de Friedman se obtuvo un p-valor de 8.27608×10^{-104} , lo cual implica que existen diferencias significativas entre los resultados obtenidos para cada señal de excitación sin importar el tipo de algoritmo PSO aplicado. Con la finalidad de verificar estos resultados la segunda prueba de Kruskal-Wallis produce un p-valor de 2.4229×10^{-104} seguido de una prueba de confianza de Bonferroni cuyos resultados se muestran en la Fig. 4.7. Se puede notar en la gráfica que las medias obtenidas mediante la prueba de Bonferroni son diferentes basándose en el tipo de señal de excitación aplicada. Más aún, los resultados que tienen una mayor similitud entre ellos son aquellos que presentan una alta riqueza espectral.

Por otra parte, para evaluar el comportamiento de los algoritmos PSO y verifi-

	Algoritmo	χ	J	\tilde{a}_*	\tilde{b}_*	\tilde{c}_*	\tilde{d}_*	$\sigma(a_*)$	$\sigma(b_*)$	$\sigma(c_*)$	$\sigma(d_*)$	I_t
1	PSO-C	1	2.448	0.4037	169.761	1.955	-0.051	1.031	28.120	2.042	1.629	379.393
2	PSO-C	9	2.957	0.5281	109.929	4.927	0.130	0.760	13.431	1.000	0.954	531.731
3	PSO-C	20	3.360	0.0111	188.804	2.223	-0.242	1.431	5.2283	1.190	0.491	340.155
4	PSO-C	52	4.995	0.0100	189.944	0.044	-0.350	0.039	1.7881	0.575	0.801	339.991
5	PSO-PVV	1	2.673	2.2333	160.409	2.369	-0.435	1.490	24.541	1.990	2.148	424.618
6	PSO-PVV	9	2.972	1.3761	108.697	5.020	0.143	1.082	17.727	1.093	0.576	509.153
7	PSO-PVV	20	3.366	0.4454	187.506	2.057	-0.249	1.637	4.1395	0.936	0.474	322.142
8	PSO-PVV	52	5.317	0.0720	188.097	1.579	-0.799	0.1918	4.2200	3.449	0.716	323.598
9	PSO-FCV	1	2.532	1.9820	163.234	2.563	-0.363	1.635	34.464	2.025	1.722	390.139
10	PSO-FCV	9	2.930	0.6570	114.761	5.353	0.334	0.654	22.283	0.919	0.678	499.786
11	PSO-FCV	20	3.247	0.0169	188.511	2.181	-0.136	0.038	4.2468	0.829	0.608	365.198
12	PSO-FCV	52	4.994	0.1013	189.737	0.206	-0.258	0.427	0.7770	0.550	0.613	368.085
13	PSO-OFr	1	2.592	2.0044	154.906	2.386	-0.025	1.557	32.6897	1.608	2.038	372.786
14	PSO-OFr	9	2.703	0.7834	103.965	5.079	0.164	0.694	8.9018	0.660	0.530	527.909
15	PSO-OFr	20	3.181	0.0001	189.962	2.079	-0.082	0.001	0.0123	0.714	0.264	331.214
16	PSO-OFr	52	4.694	0.0002	189.986	0.001	-0.279	0.001	0.0002	0.001	0.175	307.085
17	PSO-RE	1	2.577	2.1507	169.940	2.527	0.398	1.582	19.958	2.380	1.903	359.611
18	PSO-RE	9	2.704	0.6093	105.822	4.823	0.117	0.725	8.4602	0.634	0.548	461.545
19	PSO-RE	20	3.170	0.0006	189.993	2.291	-0.133	0.003	0.0352	0.165	0.193	328.684
20	PSO-RE	52	4.601	0.0005	189.952	0.004	-0.309	0.002	0.2200	0.015	0.504	308.911
	CLIE			\hat{a}_{clie}	\hat{b}_{clie}	\hat{c}_{clie}	\hat{d}_{clie}					
		52	—	3.23	178.62	1.15	-0.23	—	—	—	—	359.6678

Tabla 4.3: Resultados de la identificación de parámetros utilizando diversos algoritmos PSO y el algoritmo CLIE.

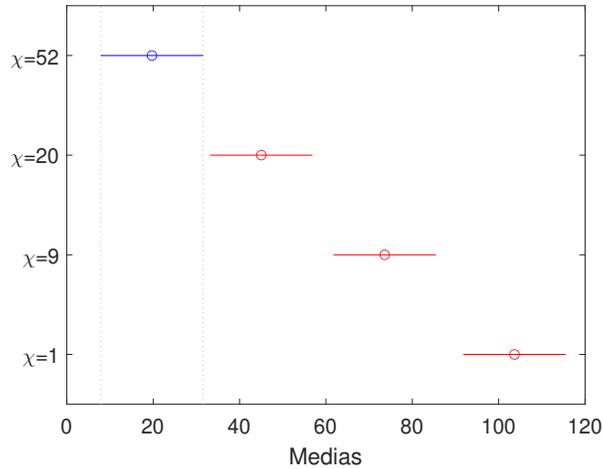


Figura 4.7: Prueba de Bonferroni para los valores de J generados por los algoritmos PSO respecto a la Riqueza Espectral χ de la señal de excitación.

Señal de excitación	P-valor
Onda senoidal	1.17×10^{-3}
Onda cuadrada filtrada	1.4×10^{-3}
Caótica	2.94×10^{-22}
Ruido blanco filtrado	4.88×10^{-27}

Tabla 4.4: P-valor obtenido de la prueba de Kruskal-Wallis para cada una de las señales de excitación aplicadas.

car que las variaciones de los parámetros estimados sean producidas por el tipo de algoritmo PSO aplicado y no por la naturaleza estocástica de la señal ϕ empleada en ellos, se emplea la prueba de Bonferroni la cual se basa en el uso de las medias obtenidas mediante la prueba Kruskal-Wallis para determinar la existencia de diferencias significativas. Se realizan cuatro pruebas de Bonferroni, cada una de ellas utiliza los resultados de la función de desempeño J obtenidos aplicando la misma señal de excitación para los diferentes algoritmos utilizados. Los p-valores de cada una de las pruebas se presentan en la Tabla 4.4 en donde se puede notar que éstos aumentan cuando se tiene un valor de χ bajo, esto significa que cuando la señal posee una baja riqueza espectral los resultados serán similares no importando qué algoritmo se utilice, mientras que si se tiene una señal de excitación con una riqueza espectral elevada el algoritmo utilizado si influirá en los resultados obtenidos para la identificación de parámetros.

Las pruebas de Bonferroni se presentan en la Fig. 4.8. En el caso de la onda senoidal los resultados obtenidos con los algoritmos PSO-OFR y PSO-RE son diferentes.

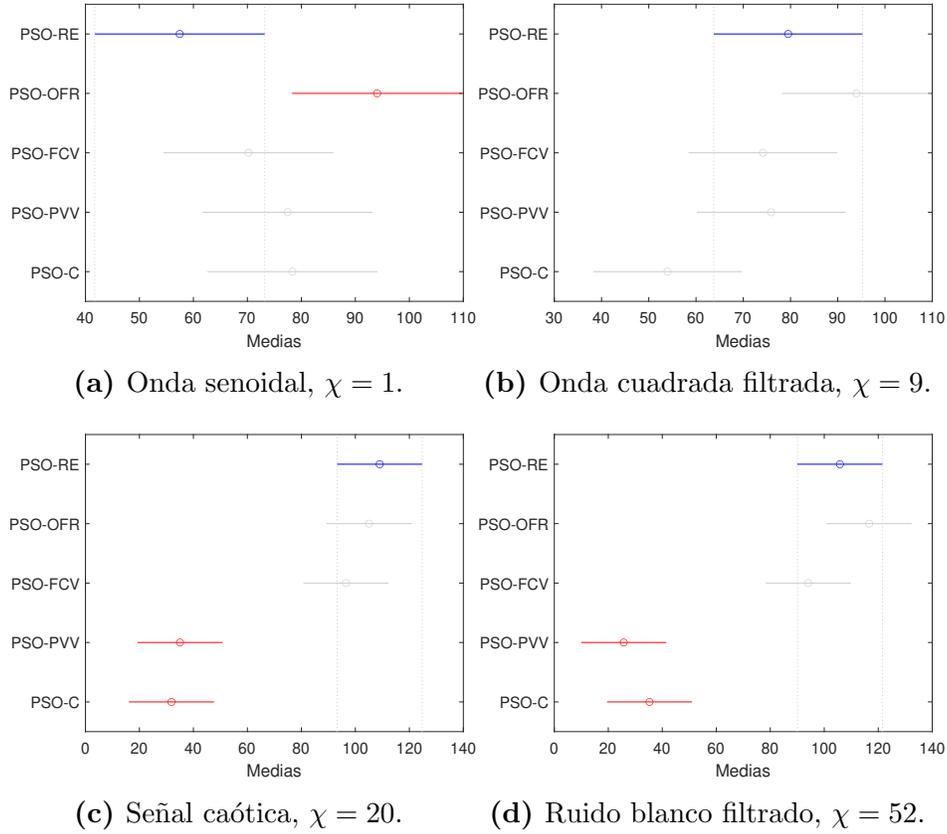


Figura 4.8: Resultados de la prueba de Bonferroni correspondientes a los algoritmos PSO para cada una de las señales de excitación aplicadas.

Cuando se emplea como excitación la onda cuadrada filtrada no existen diferencias significativas entre todos los algoritmos. En los casos donde se utilizan las señales que poseen un valor alto de χ se observa que los algoritmos más simples PSO-C y PSO-PVV tienen comportamientos similares pero generan resultados diferentes a los obtenidos mediante algoritmos PSO-FCV, PSO-OFR y PSO-RE.

4.2.2. Análisis estadístico de los parámetros

El desempeño de los algoritmos PSO respecto a los parámetros obtenidos se presenta en la Tabla 4.3 y con este fin se utiliza el parámetro \tilde{b}_* . La elección se debe a que este término se relaciona con la ganancia del servomecanismo por lo que si no es identificado correctamente produce una mayor pérdida de desempeño en una ley de control que emplea este parámetro.

En la Tabla 4.3 se puede notar un incremento de la desviación estándar, denotada por σ , para el parámetro \tilde{b}_* cuando se utilizan señales con una baja riqueza espectral. Este fenómeno se puede observar en la Fig. 4.9 donde se presenta el gráfico de caja

para el parámetro \tilde{b}_* obtenido de acuerdo a la señal de excitación aplicada y al tipo de algoritmo PSO.

A partir de los resultados anteriores, se puede observar que independientemente del algoritmo utilizado, la variabilidad de los parámetros obtenidos disminuye conforme se incrementa la RE de la señal de excitación aplicada al servomecanismo.

También es importante mencionar que el algoritmo PSO-RE presenta la menor variabilidad comparado con el resto de los algoritmos cuando $\chi = 1$. Por otro lado, Cuando se consideran las señales con valores de Riqueza Espectral $\chi = 9$, $\chi = 20$ y $\chi = 52$, el algoritmo PSO-RE proporciona una variabilidad menor que la correspondiente a los algoritmos PSO-C, PSO-PVV y PSO-FCV y similar a la obtenida por medio del algoritmo PSO-OFR.

4.3. Implementación de los parámetros identificados en una ley de control en lazo cerrado

Una de las principales motivaciones para llevar a cabo el proceso de identificación de parámetros es el diseño de leyes de control por realimentación. El objetivo es que la ley de control compense la dinámica original de un sistema para poder imponerle una dinámica deseada. En el caso del servomecanismo, la estimación de parámetros permite desarrollar leyes de control que generen el seguimiento de una referencia variante en el tiempo.

Suponiendo que los parámetros θ (4.5) del modelo (4.1) son conocidos se puede proponer una ley de control que permite el seguimiento de una referencia q_d variante en tiempo:

$$u = \frac{1}{b}[-d + a\dot{q} + c\text{sign}(\dot{q}) + \ddot{q}_d + 2\xi\omega_n\dot{e}_t + \omega_n^2 e_t] \quad (4.14)$$

donde el término $e_t = q_d - q$ es el error de seguimiento, y ω_n y ξ son constantes positivas.

Substituyendo la ley de control (4.14) en el modelo (4.1) se obtiene la siguiente dinámica del error de seguimiento:

$$\ddot{e}_t + 2\xi\omega_n\dot{e}_t + \omega_n^2 e_t = 0 \quad (4.15)$$

El punto de equilibrio en lazo cerrado de este sistema es $[e_t, \dot{e}_t]^\top = [0, 0]^\top$, el cual es exponencialmente estable. Por lo tanto, el error de seguimiento e_t converge exponencialmente a cero.

En la práctica el vector de parámetros θ es desconocido y sólo se tienen disponibles

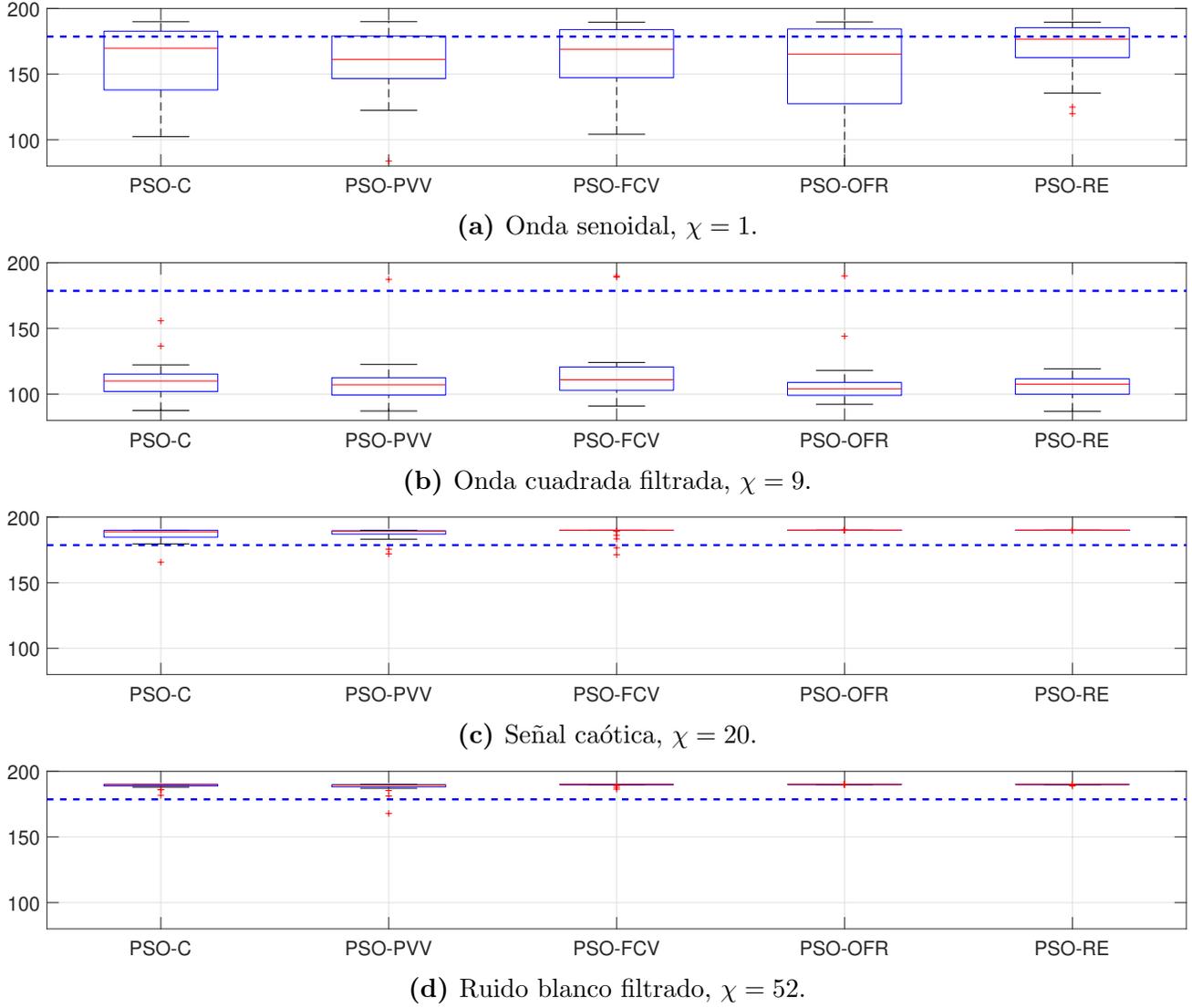


Figura 4.9: Variabilidad del estimado \tilde{b}_* dependiendo del algoritmo PSO aplicado y el tipo de señal de excitación utilizada para la adquisición de datos. La línea puntuada azul indica el valor de los parámetros estimados obtenidos con el algoritmo CLIE.

valores estimados de ellos. Entonces, se emplea el vector de parámetros estimados $\tilde{\theta}_*$ (4.10) para calcular la ley de control (4.14):

$$u = \frac{1}{\tilde{b}_*} [-\tilde{d}_* + \tilde{a}_* \dot{q} + \tilde{c}_* \text{sign}(\dot{q}) + \ddot{q}_d + 2\xi\omega_n \dot{e}_t + \omega_n^2 e_t] \quad (4.16)$$

Al substituir (4.16) en (4.1) se obtiene:

$$\ddot{e}_t + 2\xi\omega_n \dot{e}_t + \omega_n^2 e_t = -\phi_s^\top (\tilde{\theta}_* - \theta) \quad (4.17)$$

donde $\phi_s = [\dot{q}, -u, \text{sign}(\dot{q}), -1]^\top$ y $\tilde{\theta}_* - \theta$ es el error paramétrico.

Si $\tilde{\theta}_* = \theta$, entonces el error de seguimiento e_t converge a cero. En cualquier otro caso el error de seguimiento converge a una vecindad cercana al punto de equilibrio $[e_t, \dot{e}_t]^\top = [0, 0]^\top$ cuyo tamaño depende de la magnitud de $\phi_s^\top (\tilde{\theta}_* - \theta)$. Debido a esto y exceptuando el caso donde ϕ_s es ortogonal a $\tilde{\theta}_* - \theta$, el error de seguimiento e_t puede utilizarse para tener una indicación sobre el valor del error paramétrico $\tilde{\theta}_* - \theta$. Entonces, un valor pequeño del error de seguimiento e_t indicaría un error paramétrico $\tilde{\theta}_* - \theta$ pequeño.

Para evaluar el seguimiento de trayectoria cuando se utilizan los parámetros estimados mediante los algoritmos PSO se propone el siguiente índice de desempeño compuesto:

$$I_t = \int_0^t 100e_t^2 d\tau + \int_0^t \left| \frac{du(\tau)}{d\tau} \right| dt \quad (4.18)$$

El término $\int_0^t 100e_t^2 d\tau$ corresponde a la Integral del Error Cuadrático (IEC) que evalúa la tarea de seguimiento. El segundo término $\int_0^t \left| \frac{du(\tau)}{d\tau} \right| dt$ es la Integral del Valor Absoluto de la Variación del Control (IAVC) la cual evalúa el nivel de castaño en la señal producida por la ley de control.

La implementación experimental de (4.16) considera los parámetros $\omega_n = 40\pi$ y $\xi = 1$ para definir la dinámica que se desea imponer al servomecanismo. Se implementa utilizando un tiempo de muestro de 0.001s y el método de integración numérica de Euler. A su vez, la señal de referencia $q_d = r$ es generada mediante el Oscilador de Duffing (4.13).

En la Tabla 4.3 se presenta el valor obtenido del índice de desempeño I_t de acuerdo a los parámetros estimados que se implementaron en la ley de control. Los resultados experimentales muestran que los estimados obtenidos con señales de excitación con una baja riqueza espectral $\chi = 1$ y $\chi = 9$ presentan mayores valores del índice I_t que aquellos obtenidos cuando se emplean señales de alta riqueza espectral $\chi = 20$ y $\chi = 52$ en la etapa de identificación de parámetros. En el caso en que se emplean parámetros estimados mediante una señal de excitación con $\chi = 1$, el índice $I_t = 359.6115$ obtenido con los parámetros producidos por el PSO-RE es menor que el valor $I_t = 372.7866$ obtenido por el algoritmo PSO-OFR. Lo anterior muestra que el algoritmo PSO-RE produce resultados adecuados aún si la señal de excitación es de baja riqueza espectral. Debe mencionarse que para el resto de las señales, estos dos algoritmos sobrepasan el desempeño de los algoritmos PSO-C, PSO-PVV y PSO-FCV. Por otra parte, para $\chi = 52$ el PSO-RE y PSO-OFR producen respectivamente $I_t = 308.9118$ e $I_t = 307.0858$, lo que demuestra que el PSO-RE proporciona resultados competitivos respecto a un algoritmo de desarrollo reciente.

La Fig. 4.10 muestra la respuesta experimental del servomecanismo en lazo cerrado con la ley de control (4.16) al utilizar los parámetros estimados mediante los algoritmos PSO cuando se tiene una señal de excitación con $\chi = 1$. Inicialmente se podría suponer que los resultados obtenidos del seguimiento de trayectoria son similares basados en lo observado en la Fig.4.10a. En la Fig.4.10b se presenta una inspección mas detallada que permite notar que al utilizar los estimados obtenidos mediante el PSO-C y el PSO-PVV se observan oscilaciones amplias durante el seguimiento de la trayectoria. Por otra parte, en la Fig. 4.10c se presenta el seguimiento de trayectoria obtenido al implementar los parámetros estimados mediante el algoritmo PSO-OFR y PSO-RE. El algoritmo PSO-OFR muestra menor cantidad de oscilaciones que las obtenidas por el PSO-C y el PSO-PVV, pero los estimados que generan la menor cantidad de oscilaciones son los obtenidos mediante el algoritmo PSO-RE.

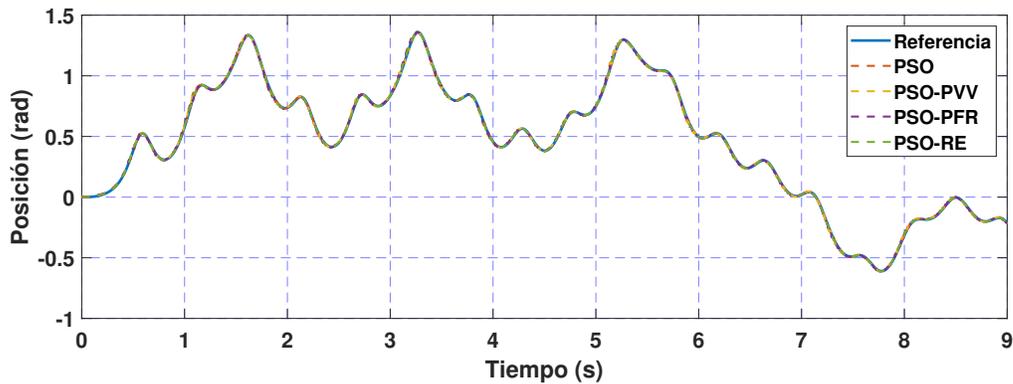
Por otra parte, en la Fig. 4.11a se observa el comportamiento del seguimiento de trayectoria cuando se utilizan los parámetros estimados obtenidos mediante una señal con riqueza espectral de $\chi = 52$. En este caso se incluye el seguimiento de trayectoria obtenido con la utilización de de los parámetros estimados mediante el algoritmo CLIE, el seguimiento presenta oscilaciones y errores notorios como puede observarse en la Fig.4.11c. En el caso de los estimados obtenidos con el resto de los algoritmos PSO se tienen comportamientos similares con un buen seguimiento de trayectoria y pocas oscilaciones. En particular el comportamiento con los estimados de PSO-OFR y PSO-RE coincide cuando se implementa ésta señal de excitación.

4.4. Conclusiones

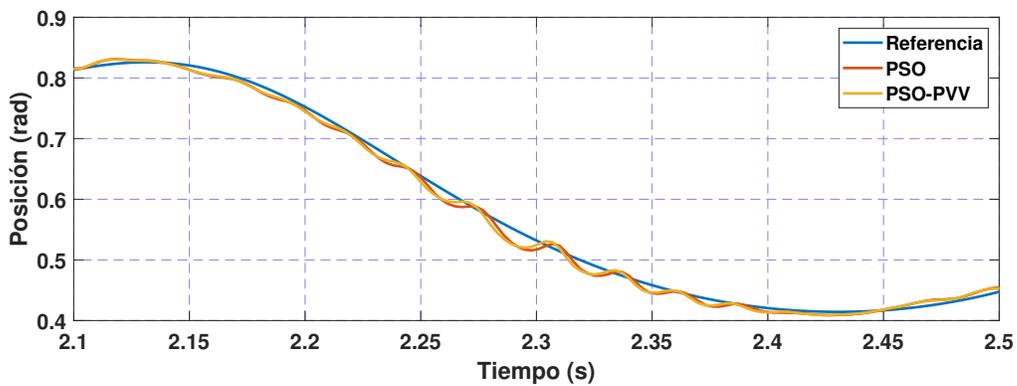
Basándose en los resultados obtenidos en este capítulo se puede concluir lo siguiente:

- La Riqueza Espectral (RE) χ de una señal puede medirse mediante el uso de la FFT.
- La variabilidad de los parámetros identificados por los algoritmos PSO disminuye cuando se emplean señales de excitación con un valor elevado de riqueza espectral χ .
- Los parámetros estimados mediante los algoritmos PSO son cercanos a los obtenidos mediante el algoritmo CLIE cuando se tiene una señal de excitación con un valor alto de χ .

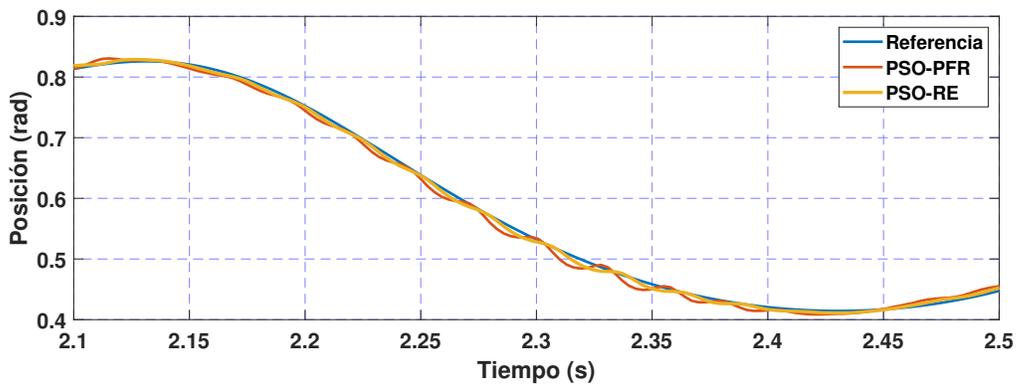
- El algoritmo PSO-RE disminuye la variabilidad de los parámetros estimados comparado con otros algoritmos PSO cuando se emplean señales de excitación con una Riqueza Espectral baja, i.e $\chi = 1$.
- Se evalúa la pertinencia de los parámetros estimado obtenidos mediante los algoritmos PSO empleándolos en el diseño de una ley de control. Midiendo el desempeño del sistema en lazo cerrado permite evaluar indirectamente la pertinencia de los parámetros estimados.
- Los parámetros estimados mediante el algoritmo PSO-RE son los que proporcionan mejor desempeño del sistema en lazo cerrado cuando la riqueza espectral es igual a $\chi = 1$.



(a) Respuesta experimental en el intervalo de tiempo 0 s – 5 s.

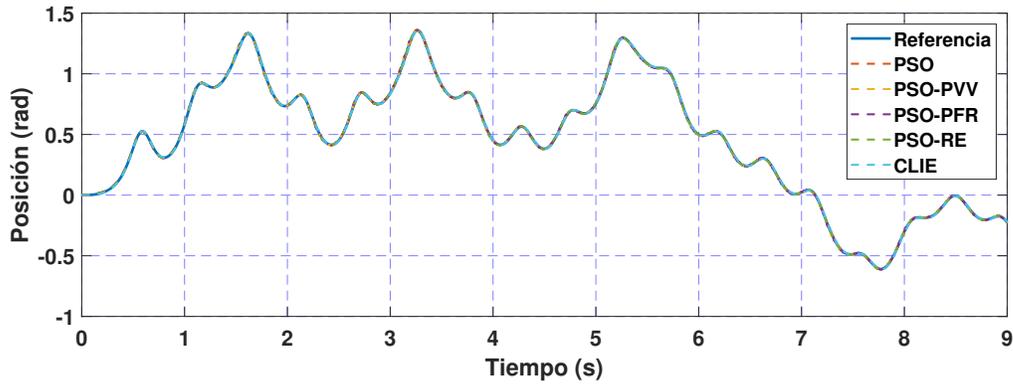


(b) Respuesta experimental en el intervalo de tiempo 2.1 s – 2.5 s correspondiente a los estimados obtenidos mediante los algoritmos PSO y PSO-PVV.

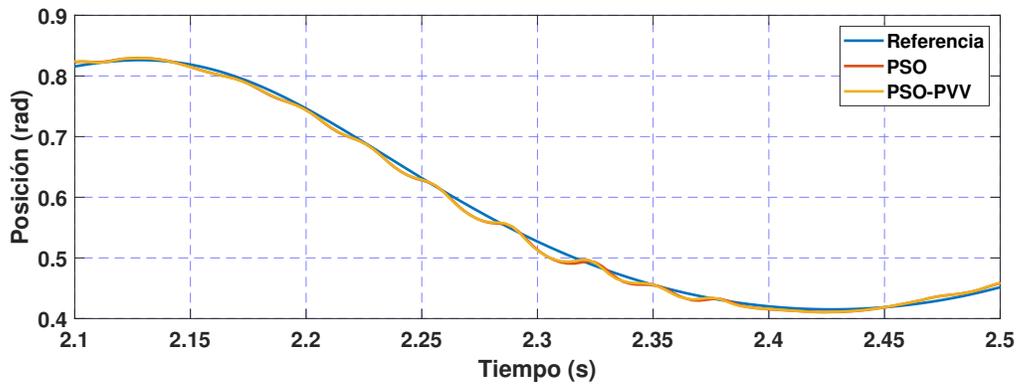


(c) Respuesta experimental en el intervalo de tiempo 2.1 s – 2.5 s correspondiente a los estimados obtenidos mediante los algoritmos PSO-PFR y PSO-RE.

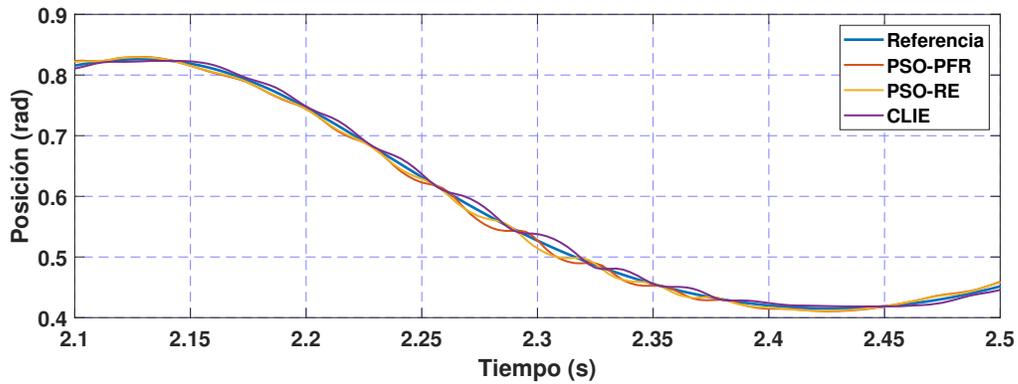
Figura 4.10: Respuesta experimental del servomecanismo cuando se aplica una ley de control dependiente de los parámetros estimados obtenidos con una señal de excitación con riqueza espectral $\chi = 1$.



(a) Respuesta experimental en el intervalo de tiempo 0 s – 5 s.



(b) Respuesta experimental en el intervalo de tiempo 2.1 s – 2.5 s correspondiente a los estimados obtenidos mediante los algoritmos PSO y PSO-PVV.



(c) Respuesta experimental en el intervalo de tiempo 2.1 s – 2.5 s correspondiente a los estimados obtenidos mediante los algoritmos PSO-OFR, PSO-RE y CLIE.

Figura 4.11: Respuesta experimental del servomecanismo cuando se aplica una ley de control dependiente de los parámetros estimados obtenidos con una señal de excitación con riqueza espectral $\chi = 52$.

Capítulo 5

Identificación paramétrica en condiciones de ruido utilizando el algoritmo PSO

El problema de identificación de parámetros cuando existe ruido de medición ha sido estudiado previamente en diversos algoritmos clásicos como el de Mínimos Cuadrados. Por otro lado, como se mencionó en la Subsección 1.2.2, en el caso de los algoritmos metaheurísticos, se han realizado estudios de identificación de parámetros pero estos no toman en cuenta el caso específico donde el ruido de medición está presente en las mediciones de la salida. Tomando como motivación esta observación, en este capítulo se presentará el análisis del efecto del ruido de medición en los parámetros identificados empleando el algoritmo PSO usando un caso numérico como ejemplo. Adicionalmente, se propondrá una metodología para la selección de frecuencias de corte de los filtros empleados para atenuar el ruido de medición que se abordará como un problema de optimización Multi-Objetivo (PMO). Los resultados se ilustrarán mediante experimentos realizados en un sistema electromecánico.

5.1. Identificación de parámetros con ruido de medición como problema de optimización numérica

Un sistema dinámico con una entrada y una salida se expresa de la siguiente manera:

$$\begin{aligned}\dot{x} &= f(t, x_s, u_s, \theta) \\ y_s &= g(t, x_s, u_s) \\ y_{me} &= y_s + \eta(t)\end{aligned}\tag{5.1}$$

donde $x_s \in \mathbb{R}^n$ son los estados del sistema, y_s es la salida, $y_{me} \in \mathbb{R}$ es la salida medida, $u_s \in \mathbb{R}$ es la señal de entrada, $\theta \in \mathbb{R}^m$ denota un vector de parámetros constantes

desconocidos y $\eta(t) \in \mathbb{R}$ representa al ruido de medición. Se supone que el sistema (5.1) es estable en lazo abierto. En caso en que no lo sea se le aplica una ley de control que lo estabilice tal y como se describe en el Capítulo 2.

Para aplicar el algoritmo PSO con propósitos de identificación paramétrica se sigue la metodología descrita en la Subsección 2.4. Para excitar al sistema se aplica una señal r , y con alta riqueza espectral. La simulación numérica del sistema (5.1) se efectúa mediante el siguiente modelo virtual:

$$\begin{aligned}\dot{x}_v &= f(t, x_v, u_v, \hat{\theta}) \\ y_v &= g(t, x_v, u_v)\end{aligned}\tag{5.2}$$

donde $x_v \in \mathbb{R}^n$ son los estados del modelo, y_v su salida, $u_v \in \mathbb{R}$ es su señal de entrada y $\hat{\theta} \in \mathbb{R}^m$ corresponde a un estimado del vector de parámetros θ . Debe mencionarse que el término $\eta(t)$ presente en el sistema no existe en su modelo.

Sea la versión en tiempo continuo de la función de desempeño J (2.3) definida como:

$$\begin{aligned}J(\hat{\theta}) &= \int_0^T (y_{me}(t) - y_v(t))^2 dt \\ &= \int_0^T (g(t, x, u) + \eta(t) - g(t, x_v, u_v))^2 dt \\ &= \int_0^T e_{me}^2 dt\end{aligned}\tag{5.3}$$

donde $T > 0$ es el tiempo durante el cual se aplica la señal de excitación al sistema (5.1) y se realiza la adquisición de datos, y $e_{me} = y_{me} - y_v$ es el error de modelo.

El problema de identificación de parámetros consistirá en minimizar la función de desempeño (5.3) respecto al vector de parámetros estimados $\hat{\theta}$ sujeto a un conjunto de soluciones factibles Ω .

Se puede notar que (5.3) no corresponde al caso ideal donde la función de desempeño es definida como:

$$J(\hat{\theta}) = \int_0^T e_i^2 dt\tag{5.4}$$

y el error $e_i = y_s - y_v$ no contiene ruido de medición, algo que en la práctica casi nunca sucede. En consecuencia, es de esperarse que el proceso de minimización no produzca los mismos parámetros estimados cuando se emplea el error e_{me} que cuando se utiliza e_i en la evaluación de la función de desempeño.

Una manera de enfrentar este problema en la práctica es mediante la aplicación de filtros a las mediciones y_{me} [80] que atenúen los efectos del ruido de medición $\eta(t)$.

Supóngase que el ruido de medición es de alta frecuencia y defínase el proceso de

filtrado:

$$\mathcal{L}\{y_f\} = \mathcal{L}\{y_{me}\} F(s) \quad (5.5)$$

donde $\mathcal{L}\{\bullet\}$ corresponde al operador de la transformada de Laplace y $F(s)$ es la función de transferencia que corresponde a un filtro pasa-bajas de tipo Butterworth definido como:

$$F(s) = \frac{1}{\prod_{k=1}^n \frac{(s+s_k)}{\omega_c}} \quad (5.6)$$

donde s_k son los polos los cuales deben seleccionarse con parte real negativa para que el filtro sea estable.

La función de desempeño a utilizar en este caso sería:

$$J(\hat{\theta}) = \int_0^T (y_f(t) - y_v(t))^2 dt \quad (5.7)$$

Notar que al aplicar el filtro se atenúa parte del espectro de la señal original y_{me} y la atenuación dependerá de la elección de los polos, o equivalentemente, de la frecuencia de corte del filtro. En consecuencia, la elección de esta última debe ser tal que se atenúe el ruido de medición sin modificar significativamente el contenido frecuencial correspondiente al comportamiento del sistema cuyos parámetros se desean identificar.

5.1.1. Ejemplo numérico

Para ilustrar las diferencias que se presentan en el mínimo de la función de desempeño y en los parámetros estimados cuando existe el ruido de medición $\eta(t)$ y cuando se filtra la señal de salida para reducir los niveles de ruido, se propone el siguiente ejemplo numérico de un sistema lineal de segundo orden que se define como:

$$\begin{aligned} \ddot{x} &= -a\dot{x} + bu \\ y &= x \\ y_{me} &= y + \eta(t) \end{aligned} \quad (5.8)$$

donde $a = 5$ y $b = 50$. Este sistema no es estable en lazo-abierto por lo que se propone la siguiente ley de control para estabilizarlo:

$$u = k_p e_t - k_d \dot{x} \quad (5.9)$$

donde el error de seguimiento se define como $e_t = r - y_{me}$. La referencia r aplicada al sistema corresponderá a un oscilador caótico de Duffing.

El ruido de medición η se supone Gaussiano de media cero y con una amplitud acotada en el intervalo de $[-0.3, 0.3]$. La simulación numérica se llevará a cabo utilizando el método de integración Runge-Kutta con un tamaño de paso de $0.001s$.

Para realizar la comparación se emplea el siguiente modelo virtual para evaluar los parámetros estimados:

$$\begin{aligned}\ddot{x}_v &= -\hat{a}\dot{x}_v + \hat{b}u_v \\ y_v &= x_v\end{aligned}\tag{5.10}$$

donde el vector de parámetros estimados se define como $\hat{\theta} = [\hat{a}, \hat{b}]^T$ y la función de desempeño a utilizar se encuentra dada por:

$$J = \int_0^T (s - y_v)^2\tag{5.11}$$

donde s es la señal aplicada para el proceso de optimización. Con la finalidad de evaluar el valor de la función de desempeño y su mínimo para cada una de las señales se propone la realización de un mapeo de la función de desempeño respecto a los parámetros estimados desde $\hat{a}_{\min} = 1$, $\hat{b}_{\min} = 1$ hasta $\hat{a}_{\max} = 100$, $\hat{b}_{\max} = 100$ con un tamaño de paso de 1. Una vez realizado el mapeo de la función se identificará el valor mínimo de J y los parámetros correspondientes que lo producen.

Para obtener los casos donde s corresponde a señales filtradas se implementa el siguiente filtro [81]:

$$F(s) = \frac{1}{\left(\frac{s}{\omega_n} + 1\right)\left(\frac{s^2}{\omega_n^2} + \frac{s}{\omega_n} + 1\right)}\tag{5.12}$$

donde ω_n son sus frecuencias de corte normalizadas. Se propone la implementación de tres frecuencias de corte diferentes para poder estudiar los efectos de la calidad del filtrado en el valor mínimo de la función de desempeño.

Tanto las frecuencias de corte normalizadas ω_n , como el valor mínimo de la función de desempeño J y sus correspondientes parámetros estimados $\hat{\theta}$ son presentados en la Tabla 5.1. De esta última se puede constatar que cuando existe ruido de medición y la señal de salida no se filtra los parámetros estimados correspondientes al mínimo de la función de desempeño son lejanos a los parámetros del sistema (5.8), mientras que en el caso donde se utiliza la señal original sin presencia de ruido los parámetros identificados son iguales a los parámetros del sistema (5.8) y la función de desempeño tiene un valor de cero. Ya que este último caso es difícil de encontrar en la práctica es necesario evaluar los efectos del filtrado en el mínimo de la función de desempeño.

Para esto se presentan en Fig. 5.1 y Fig. 5.2 las gráficas de la señal y_{me} y de cada una de las señales resultantes del proceso de filtrado y su correspondiente FFT

Señal s	ω_n	J	\hat{a}	\hat{b}
y	-	0	5	50
y_{me}	-	155.9	4	71
y_{f_1}	0.08	17.5213	2	43
y_{f_2}	0.20	24.8452	4	52
y_{f_3}	0.50	97.5362	3	59

Tabla 5.1: Parámetros estimados donde la función de desempeño presenta el valor mínimo para cada una de las señales.

la cual permita analizar las componentes frecuenciales de las señales filtradas y sin filtrar. En la Fig. 5.1a se puede notar que la señal y_{me} posee amplios niveles de ruido respecto a la señal original lo que se ve reflejado en un incremento de la amplitud de las componentes frecuenciales superiores a $8Hz$ mostradas en la Fig. 5.1b. Se puede notar que esta diferencia en las frecuencias superiores a $8Hz$ da como resultado que el mínimo de la función de desempeño se encuentre en los valores estimados $\hat{a} = 4$ y $\hat{b} = 71$, los cuales son claramente diferentes a los parámetros reales del sistema (5.8), mostrando que dicha señal es inadecuada para ser utilizada en un proceso de identificación de parámetros.

Cuando se utilizan señales filtradas es importante considerar que el proceso de filtrado provoca una modificación de la señal original por lo que se estudiarán tres casos. El primer caso corresponde a la señal y_{f_1} la cual se genera al considerar la frecuencia normalizada 0.08, en la Fig. 5.1c es claramente observable que la selección de una frecuencia de corte baja produce una atenuación del ruido pero sufre una modificación importante respecto a la señal ideal y . El origen de esta distorsión se puede observar en la Fig. 5.1d donde las componentes de frecuencia menores a $4Hz$ son mayores que los de la señal ideal y implicando que la utilización de una frecuencia de corte inadecuada puede provocar modificaciones en las frecuencias bajas. Como resultado los parámetros estimados $\hat{a} = 2$ y $\hat{b} = 43$ presentan diferencias respecto de los parámetros del sistema (5.8).

El segundo caso corresponde a la señal y_{f_2} la cual se obtiene al considerar una frecuencia de corte normalizada de 0.20 para el filtro (5.12). En la Fig. 5.2a se muestra que el comportamiento de la señal filtrada tiene una mayor similitud que el obtenido por la señal y_{f_1} lo que también se puede observar en la Fig. 5.2b donde es claro que el proceso de filtrado afecta en mayor medida a las componentes superiores a $6Hz$. La atenuación del ruido combinada con una mayor similitud a la señal original produjeron estimados cercanos a los del sistema, i.e. $\hat{a} = 4$ y $\hat{b} = 52$ al realizar la

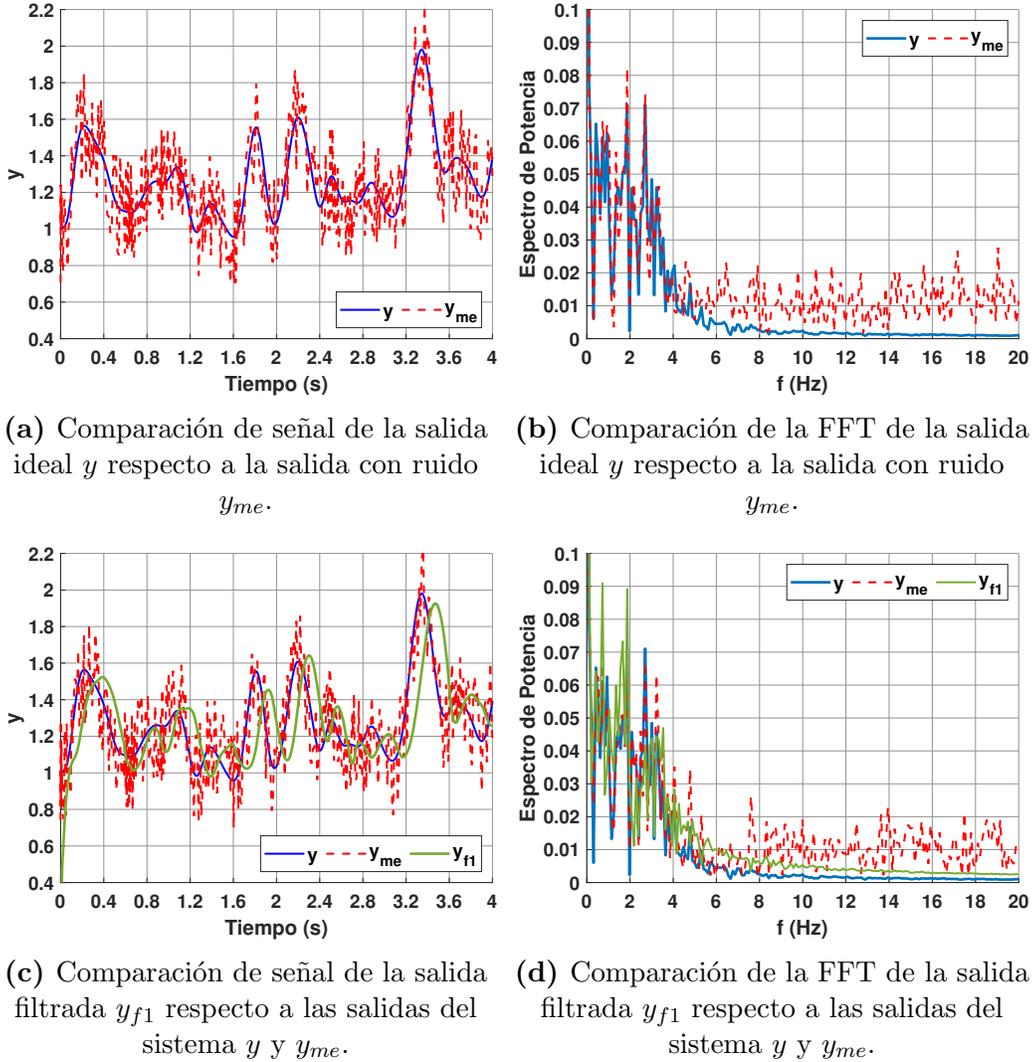
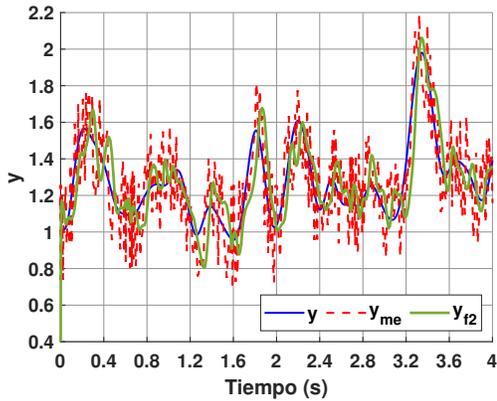


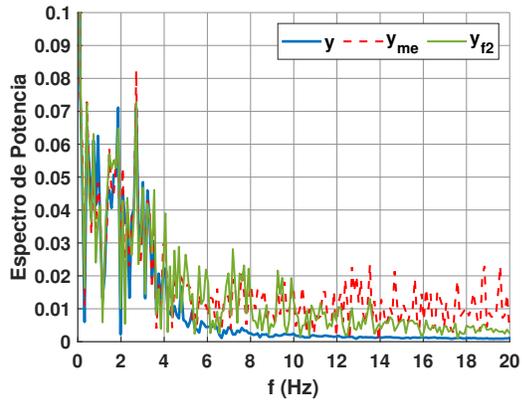
Figura 5.1: Comparación de las señales utilizadas para el cálculo de J y el análisis de componentes frecuenciales mediante la FFT.

búsqueda del mínimo de la función de desempeño J .

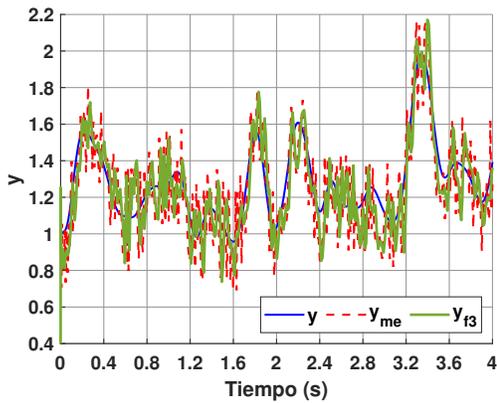
Finalmente el caso de la señal y_{f3} al utilizar la frecuencia de corte normalizada 0.50 permite estudiar el caso donde el ruido de medición no se ha atenuado adecuadamente. En la Fig. 5.2c se muestra que la señal utilizada aún contiene componentes significativos de ruido de medición aunque en menor medida respecto a la señal y_{me} . Sin embargo la presencia de las componentes de ruido en el ancho de banda superior a los $6Hz$ no presenta cambios significativos. Esto se constata en los parámetros estimados $\hat{a} = 3$ y $\hat{b} = 59$, los cuales difieren de los parámetros del sistema (5.8). Por lo que se puede concluir que un filtrado ineficiente donde las componentes de ruido no sean atenuadas adecuadamente puede provocar que el mínimo de la función de



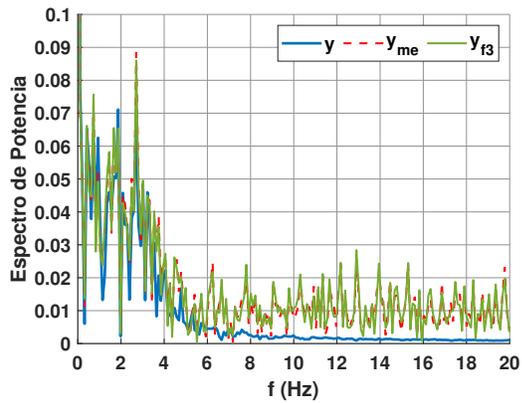
(a) Comparación de señal de la salida filtrada y_{f2} respecto a las salidas del sistema y y y_{me} .



(b) Comparación de la FFT de la salida filtrada y_{f2} respecto a las salidas del sistema y y y_{me} .



(c) Comparación de señal de la salida filtrada y_{f3} respecto a las salidas del sistema y y y_{me} .



(d) Comparativa de la FFT de la salida filtrada y_{f3} respecto a las salidas del sistema y y y_{me} .

Figura 5.2: Comparación de las señales utilizadas para el cálculo de J y el análisis de componentes frecuenciales mediante la FFT.

desempeño difiera del mínimo que se obtendría con la señal ideal y .

Se puede notar que la selección de la frecuencia de corte del filtro utilizado es de suma importancia para que los parámetros que generan el mínimo de la función de desempeño no difieran demasiado respecto a los parámetros que se obtendrían con la salida sin ruido. Si la frecuencia de corte es demasiado baja entonces se producirá una modificación importante de la señal, dando como resultado que el mínimo de la función de desempeño corresponda a parámetros estimados diferentes de los del sistema. Por otra parte, si la frecuencia de corte es demasiado alta entonces las componentes de ruido no son eliminadas adecuadamente, lo que provoca que los parámetros que corresponde al mínimo de la función de desempeño sean lejanos a los del sistema. Debido a esto es necesario realizar una selección cuidadosa de la frecuencia de corte del filtro para evitar que se produzca estos problemas.

5.2. Sintonización de filtros continuos como un problema de optimización Multi-Objetivo

Tal como se mostró en el ejemplo numérico anterior, la selección de la frecuencia de corte para el proceso de filtrado influye en la cercanía de los parámetros estimados que producen el mínimo de la función de desempeño respecto a los parámetros reales del sistema simulado. Para llevar a cabo la selección se propone definir la sintonización de un filtro continuo de tipo Butterworth [82] como un problema de optimización Multi-Objetivo (POMO). El filtro se define como:

$$H(s) = \frac{1}{\prod_{k=1}^o \frac{(s + s_k)}{\omega_c}} \quad (5.13)$$

donde ω_c es la frecuencia de corte y s_k es un coeficiente definido como:

$$s_k = \omega_c e^{\frac{j(2k+o-1)\pi}{2n}} \quad k = 1, \dots, o \quad (5.14)$$

de manera que los parámetros a ajustar del filtro sean la frecuencia de corte ω_c y el orden del filtro o . Tomar en cuenta el orden o del filtro Butterworth permite incrementar la atenuación del ruido cuando el orden se incrementa. Consecuentemente, un valor elevado del orden o atenúa en mayor medida las componentes espectrales de alta frecuencia [83]. Ya que el orden del filtro sólo modifica los niveles de atenuación sin afectar las frecuencias que se desean conservar se propondrá como una variable a optimizar con la finalidad de verificar si la aplicación de un filtro de orden mayor

puede generar una mejora en el acondicionamiento de la señal.

En la definición del problema de optimización MO se tendrán dos funciones de desempeño diferentes y cada una de ellas toma en cuenta uno de los efectos negativos que tiene una selección incorrecta de la frecuencia del filtro tal y como se observó en la sección anterior.

Para proponer la primera función de desempeño para ajustar la frecuencia de corte se propone considerar la similitud entre la señal original y la señal filtrada. Cuando se lleva a cabo un filtrado con una frecuencia de corte demasiado baja, la señal resultante presenta un desfase producto del cambio en las componentes de alta frecuencia de la señal.

Para evaluar la similitud entre las señales se propone el uso de los coeficientes de correlación de Pearson [84] que se definen de la siguiente manera. Dadas dos señales y y y_f los coeficientes de correlación se estiman usando la siguiente fórmula:

$$\rho(y, y_f) = \frac{\sum_{j=1}^N (y(j) - \bar{y})(y_f(j) - \bar{y}_f)}{\left(\sum_{j=1}^N (y(j) - \bar{y})^2 \sum_{j=1}^N (y_f(j) - \bar{y}_f)^2 \right)^{1/2}} \quad (5.15)$$

donde \bar{y} y \bar{y}_f son las medias de las señales y N es el número de muestras en las señales y y y_f . El coeficiente ρ se encuentra acotado entre $[-1, 1]$ y sus valores implican que:

- Si $\rho = 1$ se tiene una correlación positiva perfecta.
- Si $\rho = -1$ se tiene una correlación negativa perfecta, lo que implica que existe una relación inversa.
- Si $\rho = 0$ las señales no están correlacionadas.

Basándose en este coeficiente se puede proponer la siguiente función de desempeño:

$$\mathcal{J}_1 = e^{-|\rho|} \quad (5.16)$$

El comportamiento de la función de desempeño \mathcal{J}_1 se muestra en la Fig. (5.3), se puede notar que cuando $\rho = 0$ la función de desempeño poseerá su valor máximo lo que significa que las señales no están correlacionadas. Conforme los valores de ρ tiendan a 1 o a -1 la función de desempeño disminuirá su valor.

El segundo factor a considerar es la presencia de ruido en la señal de salida. Si existe una gran cantidad de componentes de ruido, entonces el mínimo de la función de desempeño puede verse afectado y los parámetros identificados pueden tener discrepancias significativas respecto a los parámetros reales del sistema a identificar. Se supone que la salida del sistema y posee un ancho de banda $[f_1, f_2]$, mientras que el

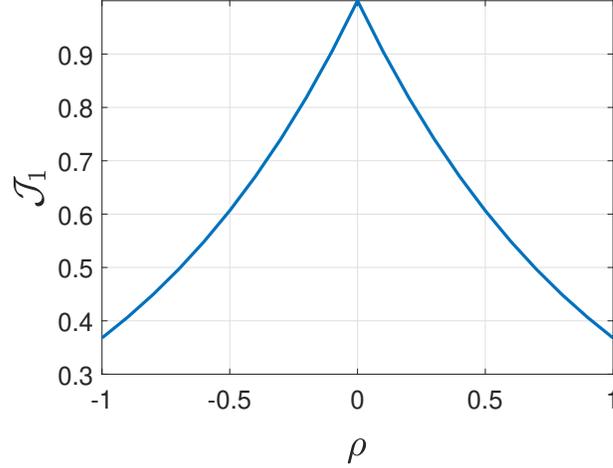


Figura 5.3: Función de desempeño \mathcal{J}_1 .

ruido de medición se considerara de alta frecuencia tal que se encuentre contenido en el ancho de banda $[f_2, f_3]$ siendo $f_1 < f_2 < f_3$.

Con la finalidad de evaluar el efecto del ruido en la señal después del proceso de filtrado, la Relación Señal-Ruido (RSR) [85] se utilizará para evaluar la calidad de la señal obtenida. Esta relación se calcula como:

$$RSR = 10 \log_{10} \left(\frac{P_{se}}{P_{ru}} \right) \quad (5.17)$$

donde P_{se} y P_{ru} corresponden a las potencias de la señal y del ruido respectivamente. La potencia de la señal se calcula como:

$$P_{se} = \int_{f_1}^{f_2} E(\tau) d\tau \quad (5.18)$$

donde P_s corresponde a la potencia en el intervalo frecuencial, E es la energía de la señal para la frecuencia τ obtenida mediante la FFT, f_1 es la frecuencia inferior usualmente considerada como cero y f_2 es la frecuencia que corresponde a la componente espectral de mayor valor en la señal de salida y_{me} . Esta se puede calcular mediante la utilización de la FFT para descomponer la señal y_{me} en sus componentes frecuenciales y entonces identificar la frecuencia de mayor valor.

La potencia del ruido se define como:

$$P_{ru} = \int_{f_2}^{f_3} E(\tau) d\tau \quad (5.19)$$

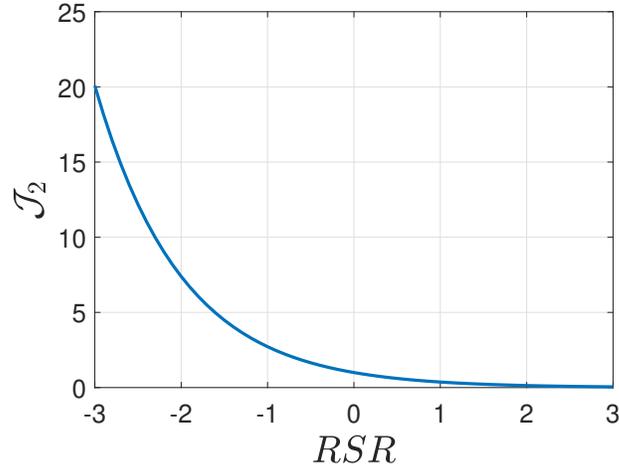


Figura 5.4: Función de desempeño \mathcal{J}_2 .

donde $f_3 = f_s/2$ es la componente espectral de frecuencia máxima de la señal y_{me} que puede ser calculada mediante el algoritmo FFT basándose en el teorema de Nyquist [77], siendo f_s la frecuencia de muestreo utilizada para la adquisición de los datos.

Tomando en cuenta lo anterior el segundo índice de desempeño se define como:

$$\mathcal{J}_2 = e^{-RSR} \quad (5.20)$$

En la Fig. 5.4 se puede observar el comportamiento de la función de desempeño \mathcal{J}_2 . Cuando el valor de RSR tiene valores negativos la función de desempeño \mathcal{J}_2 se incrementa, esto se debe a que un valor negativo de RSR corresponde a una señal donde existe una gran cantidad de ruido de medición. Cuando el valor de RSR es positivo el valor de la función de desempeño \mathcal{J}_2 disminuye hasta ser cercano a cero.

Por lo que el problema MO utilizado para escoger el orden y la frecuencia del filtro se define como:

$$\min (\mathcal{J}([f_c, o])) = [\mathcal{J}_1(f_c, o), \mathcal{J}_2(f_c, o)]^T \quad (5.21)$$

donde $f_c \in \mathbb{R}$ es la frecuencia de corte y $o \in \mathbb{N}$ es el orden del filtro.

5.3. Algoritmos de optimización para problemas Multi-Objetivo

Dado que se desea resolver un problema de optimización MO para la sintonización del filtro (5.13), es necesario utilizar un algoritmo especializado en la resolución de estos problema. Por esta razón se propone la utilización de los algoritmos Multi-Objective Particle Swarm Optimization (MOPSO) y Non-Dominated Sorting Genetic

Algorithm II (NSGA-II).

5.3.1. Multi-Objective Particle Swarm Optimization (MOPSO)

El algoritmo MOPSO [12] es una modificación del algoritmo PSO (2.1) que le permite solucionar problemas de optimización Multi-Objetivo. En este caso, la solución única x^* obtenida mediante el algoritmo PSO es sustituida en el algoritmo MOPSO por un único frente de Pareto \mathcal{PF}^* (1.9) que se actualiza durante cada iteración.

El algoritmo MOPSO continua considerando las soluciones x_i como partículas con su propia dinámica dada por:

$$\begin{aligned}\omega_i(k+1) &= \beta\omega_i(k) + \phi_1(k)p(x_{i,*}(k) - x_i(k)) + \phi_2(k)l(\mathcal{X}(h) - x_i(k)) \\ x_i(k+1) &= \omega_i(k+1)\end{aligned}\quad (5.22)$$

donde $\mathcal{X} \in \mathbb{R}^v$ es el conjunto de soluciones no-dominadas que son óptimas de Pareto \mathcal{P} (1.8) y siendo $h \in \mathbb{N}$ un término generado aleatoriamente en el intervalo $h \in [1, v]$. Este conjunto de soluciones se actualiza en cada iteración al incluir las posiciones actuales de las partículas del enjambre y al eliminar todas aquellas soluciones que no sean óptimas de Pareto \mathcal{F} . En consecuencia, la dimensión v del conjunto varía una vez que el algoritmo se ejecuta. El cálculo del término $x_{i,*}(k)$ es realizado mediante:

$$x_{i,*}(k) = \begin{cases} x_i(k) & \text{si } x_i(k) \prec x_{i,*}(k-1) \\ x_{i,*}(k-1) & \text{si } x_{i,*}(k-1) \prec x_i(k) \end{cases}\quad (5.23)$$

donde $\vec{x} \prec \vec{y}$ denota la dominancia de un vector \vec{x} respecto a un vector \vec{y} descrita en (1.6). La implementación del MOPSO se presenta en el **Algoritmo 2**.

5.3.2. Non-Dominated Sorting Genetic Algorithm II (NSGA-II)

El segundo algoritmo considerado es el Non-Dominated Sorting Genetic Algorithm II (NSGA-II) [13], el cual ha sido ampliamente utilizado para la resolución de problema de optimización MO. Se basa en la idea de la evolución de r_p individuos de una población durante g generaciones mediante los procesos de mutación y cruza, donde se favorece la reproducción de los individuos con una mejor aptitud de supervivencia. Cada uno de estos individuos corresponde a una solución $x_i \in \mathbb{R}^n$ y su aptitud de supervivencia es evaluada mediante $\mathcal{J} = (\mathcal{J}_2(\cdot), \mathcal{J}_2(\cdot))$.

El algoritmo NSGA-II utiliza diversos mecanismos que le permiten evaluar la pertinencia de las soluciones con la finalidad de seleccionar aquellas que serán utilizadas para generar la siguiente generación. Estos mecanismos son descritos a continuación:

Algoritmo 2: Multi-Objective Particle Swarm Optimization (MOPSO)

Datos: Ω y $\mathcal{J} = (\mathcal{J}_1(\cdot), \mathcal{J}_2(\cdot))$

Resultado: \mathcal{PF}^*

- 1 Inicializar la población $x_i(1) \quad i = 1, \dots, r_p$ con soluciones aleatorias tales que $x_i \in \Omega$;
 - 2 Inicializar la velocidad de cada partícula tal que $\omega_i(1) = 0 \quad i = 1, 2, \dots, r_p$;
 - 3 Evaluar las soluciones $x_i(1)$ utilizando \mathcal{J} ;
 - 4 Identificar los vectores no-dominados \mathcal{F} y guardarlos en el conjunto \mathcal{X} ;
 - 5 Inicializar los términos $x_{i,*}(1) = x_i(1) \quad i = 1, \dots, r_p$;
 - 6 Colocar $k = 2$;
 - 7 **Mientras** $k < k_{max}$ **hacer**
 - 8 | Calcular la velocidad y posición de cada partícula i utilizando (5.22) ;
 - 9 | Si $x_i(k) \notin \Omega$ entonces $x_i(k) = x_i(k - 1)$;
 - 10 | Evaluar cada partícula $x_i(k)$ utilizando \mathcal{J} .;
 - 11 | Se calculan $x_{i,*}(k)$ utilizando (5.23). ;
 - 12 | Incluir las partículas $x_i(k) \quad i = 1, \dots, r_p$ en el conjunto \mathcal{X} ;
 - 13 | Determinar la Optimalidad de Pareto \mathcal{F} para las soluciones del conjunto \mathcal{X} utilizando (1.7).;
 - 14 | Eliminar del conjunto \mathcal{X} las soluciones que no corresponda a un óptimo de Pareto \mathcal{F} . ;
 - 15 | $k = k + 1$;
 - 16 **Fin**
 - 17 $\mathcal{PF}^* = \mathcal{X}$
-

- El **Acomodo rápido no dominante** es un algoritmo utilizado para acomodar las soluciones basándose en el concepto de dominancia de Pareto (1.6). Este genera varios frentes \mathcal{F}_i tales que la condición $\mathcal{F}_1 \prec \mathcal{F}_2 \prec \dots$ implica que las soluciones contenidas en el frente \mathcal{F}_1 no son dominadas por ninguna otra, mientras que las soluciones en el frente \mathcal{F}_2 son dominadas únicamente por las soluciones del primer frente y así sucesivamente. El rango R_n asignado a cada solución corresponde al frente \mathcal{F}_j en el que se encuentre tal que:

$$R_n(x_i) = j \quad x_i \in \mathcal{F}_j \quad (5.24)$$

- El **Torneo de selección binario** corresponde a la selección de dos soluciones aleatorias y la comparación de los rangos asignados, tal que se selecciona la solución cuyo rango sea menor. Una vez que estas soluciones han sido utilizadas no se vuelven a emplear para realizar una comparación. Cuando todas las soluciones han sido utilizadas se tendrán $\frac{T_p}{2}$ soluciones y con ellas se llevará a cabo un proceso de **Recombinación y mutación** [86] para generar una nueva población \mathcal{V} .
- La **Asignación de distancia de apiñamiento** corresponde a un algoritmo que evalúa la distancia que existe entre las soluciones de un frente \mathcal{F}_i con la finalidad de determinar la diversidad de sus soluciones. Mientras mayor sea la distancia D_c entre las soluciones, se dice que existirá una mayor diversidad entre ellas ya que no estarán apiñadas en un solo punto del frente definido con el **Algoritmo 3**.
- El **operador de comparación de apiñamiento** \prec_n utiliza el rango de las soluciones R_n y su distancia de apiñamiento D_c permite comparar entre dos soluciones x_i y x_j , y definir si una está dominada respecto a la otra. El proceso que utiliza este operador se define en el **Algoritmo 4**

El **Algoritmo 5** presenta el pseudo-código requerido para la implementación del NSGA-II.

5.4. Optimización Multi-Objetivo de la frecuencia de corte de un filtro continuo aplicado a señales con ruido

En esta sección se realiza la optimización de la frecuencia de corte y del orden de un filtro cuya función de transferencia está descrita en (5.13), empleando señales

Algoritmo 3: Asignación de distancia de apiñamiento

Datos: $\mathcal{F}_i, \mathcal{J}_1, \mathcal{J}_2$
Resultado: $D_c(x_i) \quad x_i \in \mathcal{F}_i$

- 1 Identificar la cantidad n_{fi} de soluciones en \mathcal{F}_i ;
- 2 Inicializar $D_c(x_i) = 0 \quad x_i \in \mathcal{F}_i$. ;
- 3 **para** $j_1 \leftarrow 1$ **a** 2 **hacer**
- 4 | Acomodar las soluciones de \mathcal{F}_i utilizando \mathcal{J}_{j_1} ;
- 5 | $D_c(\mathcal{F}_i(1)) = \infty$;
- 6 | $D_c(\mathcal{F}_i(n_{fi})) = \infty$;
- 7 | **para** $j_2 \leftarrow 2$ **a** $(n_{fi} - 1)$ **hacer**
- 8 | | $D_c(\mathcal{F}_i(j_2)) = \frac{D_c(\mathcal{F}_i(j_2)) + (\mathcal{F}_i(j_2+1) - \mathcal{F}_i(j_2-1))}{\mathcal{J}_{j_1}(\max) - \mathcal{J}_{j_1}(\min)}$;
- 9 | | Donde $\mathcal{J}_{j_1}(\max) = \operatorname{argmax}_{x_i \in \mathcal{F}_i} \{\mathcal{J}_{j_1}(x_i)\}$;
- 10 | | Donde $\mathcal{J}_{j_1}(\min) = \operatorname{argmin}_{x_i \in \mathcal{F}_i} \{\mathcal{J}_{j_1}(x_i)\}$;
- 11 | **fin**
- 12 **fin**

Algoritmo 4: Operador de comparación de apiñamiento

Datos: $x_i, x_j, D_c(x_i), D_c(x_j), R_n(x_i), R_n(x_j)$
Resultado: \prec_n

- 1 **Si** $R_n(x_i) < R_n(x_j)$ **entonces**
- 2 | $x_i \prec_n x_j$;
- 3 **En otro caso**
- 4 | **Si** $R_n(x_i) = R_n(x_j)$ **y** $R_n(x_i) > R_n(x_j)$ **entonces**
- 5 | | $x_i \prec_n x_j$;
- 6 | **En otro caso**
- 7 | | $x_j \prec_n x_i$;
- 8 | **Fin**
- 9 **Fin**

Algoritmo 5: Non-Dominated Sorting Genetic Algorithm II (NSGA-II)

Datos: Ω, g, N, C, M y $\mathcal{J} = (\mathcal{J}_2(\cdot), \mathcal{J}_2(\cdot))$

Resultado: \mathcal{PF}^*

- 1 Inicializar la población \mathcal{X} que contiene $x_i(1) \quad i = 1, \dots, r_p$ con soluciones aleatorias tales que $x_i \in \Omega$;
 - 2 Evaluar las soluciones $x_i(1)$ utilizando \mathcal{J} ;
 - 3 Asignar un rango $R_n(x_i)$ basándose en la **Acomodo rápido no dominante.** ;
 - 4 Generar una población de vástagos \mathcal{V} mediante **Torneo de selección binario** y **Recombinación y mutación** utilizando \mathcal{X} .;
 - 5 Colocar $k = 2$;
 - 6 **Mientras** $k < k_{max}$ **hacer**
 - 7 | Generar una población combinada $\mathcal{R} = \mathcal{X} \cup \mathcal{V}$. ;
 - 8 | Implementar el **Acomodo rápido no dominante** para generar $\mathcal{F} = [\mathcal{F}_1, \mathcal{F}_2, \dots]$ el cual contiene los frentes no-dominados de \mathcal{R} . ;
 - 9 | Inicializar $j = 1$ y $\mathcal{O} = \emptyset$;
 - 10 | **Mientras** $S_z(\mathcal{O} \cup \mathcal{F}_j) \leq r_p$ **hacer**
 - 11 | | Implementación de **Asignación de distancia de apiñamiento** al frente \mathcal{F}_j ;
 - 12 | | $\mathcal{O} = \mathcal{O} \cup \mathcal{F}_j$;
 - 13 | | $j = j + 1$;
 - 14 | **Fin**
 - 15 | Acomodar en orden descendente \mathcal{F}_j utilizando el operador \prec_n ;
 - 16 | Asignar $\mathcal{X} = \mathcal{O} \cup \mathcal{F}_j[1 : (r_p - S_z(\mathcal{O}))]$;
 - 17 | Generar una población de vástagos \mathcal{V} mediante **Torneo de selección binario** y **Recombinación y mutación.** utilizando \mathcal{X} ;
 - 18 | $k = k + 1$;
 - 19 **Fin**
 - 20 $\mathcal{PF}^* = \mathcal{X}$
-

obtenidas experimentalmente mediante un prototipo de laboratorio consistente de un servomecanismo, mediante el uso de la optimización Multi-Objetivo descrita en la sección precedente. La salida en posición del servomecanismo contiene ruido de medición. El modelo del servomecanismo es no lineal pero se espera que de manera similar a lo presentado en la Sección 5.1.1 exista una sintonización del filtro que permita atenuar las componentes de ruido sin producir alteraciones notorias en las componentes espectrales de baja frecuencia.

5.4.1. Modelo del servomecanismo

Se considera un modelo no lineal del servomecanismo de mayor complejidad que el presentado en la Sección 5.1.1 y cuya estructura está dada por [20]:

$$\begin{aligned}\ddot{q} &= -a\dot{q} + bu - c\text{sign}(\dot{q}) + d \\ q_{me} &= q + \eta(t)\end{aligned}\tag{5.25}$$

donde q , \dot{q} y \ddot{q} son la posición, velocidad y aceleración angular respectivamente, mientras que u es la señal de control aplicada. La variable q_{me} es la posición medida del servomecanismo, mientras que $\eta(t)$ corresponde a un término de ruido de medición cuyas componentes espectrales se encuentran en un ancho de banda superior al ancho de banda de la dinámica del servomecanismo. El término a se relaciona a la fricción viscosa, b es la ganancia de entrada, el término c esta relacionado a la fricción de Couloumb, $\text{sign}(\cdot)$ es la función signo y d corresponde a una perturbación constante generada por los voltajes parásitos en el amplificador de potencia.

Debido a que el sistema electromecánico no es estable en lazo abierto, es necesario estabilizarlo por medio de un controlador de realimentación Proporcional Derivativo (PD):

$$u = k_p e - k_d \dot{q}_{fme}\tag{5.26}$$

las constantes $k_p, k_d \in \mathbb{R}^+$ son las ganancias proporcional y derivativa respectivamente. El error de posición se define como $e = r - q_{me}$ y el término \dot{q}_{me} corresponde a la estimación de la velocidad basada en la posición medida. La estimación de esta velocidad \dot{q}_{fm} primero requiere de obtener una medición filtrada de la posición q_{fm} :

$$\mathcal{Q}_{fme} = \mathcal{Q}_{me} W(s)\tag{5.27}$$

siendo $\mathcal{Q}_{fme} = \mathcal{L}\{q_{fme}\}$ y $\mathcal{Q}_{me} = \mathcal{L}\{q_{me}\}$, donde la notación $\mathcal{L}\{\bullet\}$ representa al

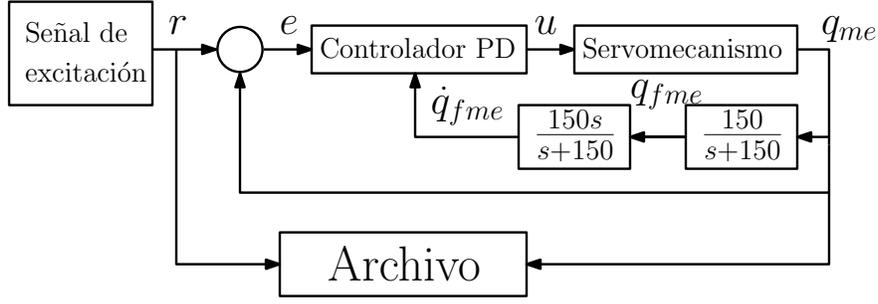


Figura 5.5: Esquema de adquisición de datos del servomecanismo para su utilización en la identificación de la frecuencia de corte y orden del filtro mediante algoritmos metaheurísticos Multi-Objetivo.

operador de Laplace. La función de transferencia que corresponde al filtro es:

$$W(s) = \frac{150}{s + 150} \quad (5.28)$$

Se aplica una segunda función de transferencia que permite la estimación de la velocidad filtrada como:

$$sQ_{fme} = Q_{fme}V(s) \quad (5.29)$$

donde $sQ_{fme} = \dot{q}_{fme}$ y la función de transferencia $V(s)$ se define como:

$$V(s) = \frac{150s}{s + 150} \quad (5.30)$$

La salida filtrada del servomecanismo q_f que se empleará en los esquemas de optimización Multi-Objetivo se obtiene mediante la implementación del filtro pasabajas (5.13) donde se modifica la frecuencia de corte ω_c y el orden del filtro n .

5.4.2. Identificación de la frecuencia de corte y el orden de un filtro Butterworth mediante algoritmos de optimización Multi-Objetivo

La identificación de la frecuencia de corte ω_c y del orden del filtro o es realizada mediante los algoritmos MOPSO [86] y NSGA-II [13]. Para realizar este proceso es necesario implementar la adquisición de datos tal como se muestra en la Figura 5.5 donde se puede notar que los datos son obtenidos cuando se ha implementado el controlador PD. Mediante el archivo de datos se resuelve el problema de optimización Multi-Objetivo (5.21). Además, se aplica el filtro (5.13) a la señal q_{me} para generar la señal filtrada q_f como se muestra en la Figura 5.6.

El coeficiente de correlación de Pearson ρ que aparece en la función de desempeño

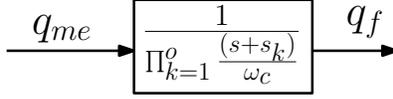


Figura 5.6: Proceso de filtrado utilizado para generar la señal q_f basándose en los parámetros ω_c y o generados por el algoritmo metaheurístico.

\mathcal{J}_1 es calculado como:

$$\rho(q_{me}, q_f) = \frac{\sum_{j=1}^N (q_{me}(j) - \bar{q}_{me})(q_f(j) - \bar{q}_f)}{\left(\sum_{j=1}^N (q_{me}(j) - \bar{q}_{me})^2 \sum_{j=1}^N (q_f(j) - \bar{q}_f)^2 \right)^{1/2}} \quad (5.31)$$

Las cotas de las variables a optimizar son las siguientes:

$$\begin{aligned} 0 < \omega_c < \frac{f_s}{2} \in \mathbb{R} \\ 1 < o < 6 \in \mathbb{N} \end{aligned} \quad (5.32)$$

donde f_s es la frecuencia de muestreo de la señal.

5.4.3. Planteamiento experimental

Los datos experimentales se obtienen a partir de una plataforma experimental construida con un motor HITEC HS-485HB RC cuya electrónica interna ha sido substituida con un amplificador de potencia Texas Instruments LM675. La posición angular del servomotor es medida mediante un potenciómetro conectado como divisor de voltaje. Esta plataforma se presenta en la Fig. 5.7.

La adquisición de los datos experimentales se realiza mediante el uso del programa Matlab-Simulink R2011b en combinación con el programa Quarc de Quanser Consulting utilizando el método de integración Runge-Kutta con un paso de integración de 0.001 s. El programa se ejecuta en una computadora personal de 32-bits PC Intel Core 2 ejecutando el sistema operativo Windows 7. Una tarjeta Q8 de Quanser Consulting permite la adquisición de los datos. Los parámetros del controlador PD son los siguientes $K_p = 3$ y $K_d = 0.075$.

El cálculo de las frecuencias requeridas para la implementación del índice RSR se realiza utilizando los datos experimentales mostrados en la Fig. 5.8. La frecuencia dominante es de 14.8551 Hz por lo que ésta será considerada la $f_2 = 14.8515$ Hz, por otra parte la frecuencia mínima que se puede obtener es $f_1 = 0$ Hz y la frecuencia máxima será $f_3 = 500$ Hz. Las componentes superiores a 200 Hz tienen una magnitud despreciable por lo que para apreciar los resultados la Fig. 5.8 solo mostrará hasta 200 Hz.

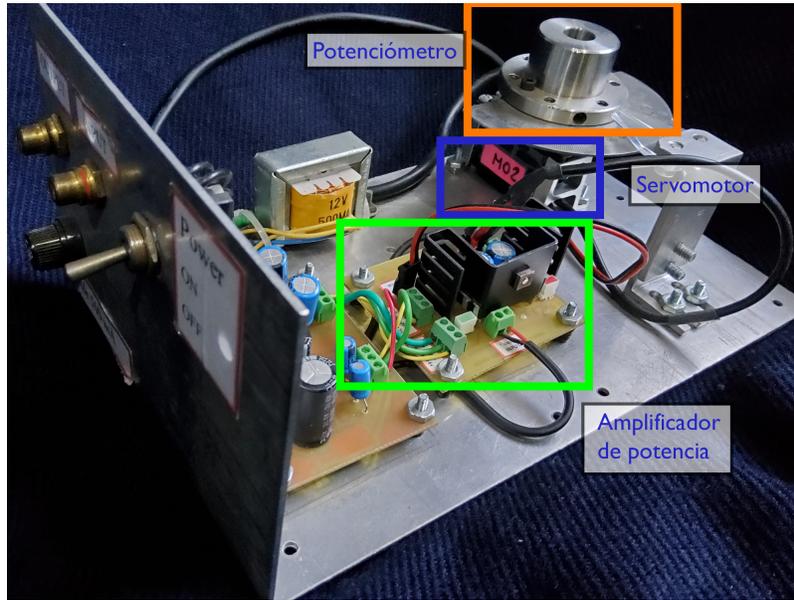


Figura 5.7: Plataforma experimental de un servomecanismo con medición de la posición utilizando un potenciómetro.

Los algoritmos MOPSO y NSGA-II son codificados y ejecutados utilizando Matlab R2020a ejecutándose en un procesador 64-bits AMD Ryzen 5 1600X con el sistema operativo Windows 10. La sintonización de los algoritmos se ha realizando mediante la herramienta IRACE [79] ejecutada en la paquetería Rstudio. Los parámetros sintonizados se presentan en la Tabla 5.2.

Algorithm	Parameters
MOPSO	$p = 1.102, l = 1.014$ $\beta = 1.03$
NSGA-II	$M = 0.2345, C = 0.8542$

Tabla 5.2: Parámetros de los algoritmos para resolución de los problema de optimización MO utilizando la heramienta IRACE.

Estos algoritmos consideran una población de 50 individuos y un total de 400 iteraciones y se ejecutan 50 veces para generar los análisis estadísticos correspondientes.

5.4.4. Frentes de Pareto resultantes de la optimización del filtro

Los frentes de Pareto presentados para los algoritmos MOPSO y NSGA-II se componen de la combinación de los frentes de Pareto obtenidos durante 50 ejecuciones de cada uno de ellos. Para cada algoritmo, al realizar la combinación de sus correspon-

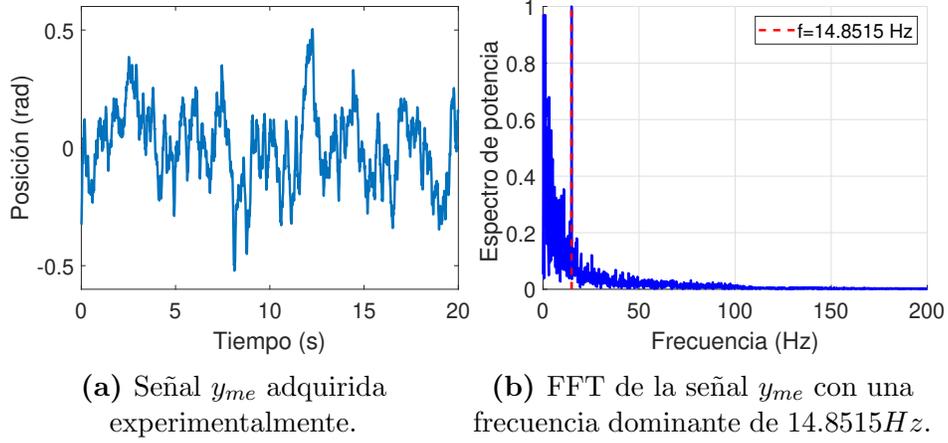


Figura 5.8: Datos experimentales obtenidos para el proceso de sintonización de un filtro pasa-bajas.

dientes 50 frentes, se eliminan las soluciones dominadas en el conjunto con la finalidad de generar un único frente de Pareto para cada uno de los algoritmos. La Fig. 5.9 presenta ambos frentes de Pareto generados para los algoritmos MOPSO y NSGA-II donde se puede notar existe una relación de compromiso donde si se incrementa el índice de desempeño \mathcal{J}_1 entonces el valor del índice \mathcal{J}_2 disminuye y viceversa. Este comportamiento justifica la implementación de una técnica de optimización Multi-Objetivo.

Con la finalidad de seleccionar la solución del frente de Pareto que se utilizará para sintonizar el filtro se propone utilizar la solución que presente una menor distancia Euclidiana respecto al punto $\mathcal{J}_1 = 0$ y $\mathcal{J}_2 = 0$, tal que:

$$\Phi(\mathcal{J}_1, \mathcal{J}_2) = \sqrt{\mathcal{J}_1^2 + \mathcal{J}_2^2} \quad (5.33)$$

de manera que la solución $[\omega_c, o] \in \mathcal{PF}$ que minimice el valor de Φ será designada como $[\omega_c^*, o^*]$. Las soluciones que minimiza el término Φ para el algoritmo MOPSO y para el algoritmo NSGA-II respectivamente se presentan en la Tabla 5.3. Se puede constatar que ambos algoritmos coinciden en que para el caso de un filtro de tipo Butterworth el orden del filtro es 1. También se puede notar que las frecuencias de corte calculadas por ambos algoritmos son cercanas entre si mostrando que alrededor de estos valores se tiene un compromiso entre la eliminación de ruido y la similitud de la señal filtrada respecto a la señal con ruido.

La evaluación de los resultados mediante el uso de una prueba no-parámetrica requiere la propuesta de un índice que evalúe los resultados obtenidos. Dado que para el problema Multi-Objetivo el resultado no es un único conjunto de valores sino

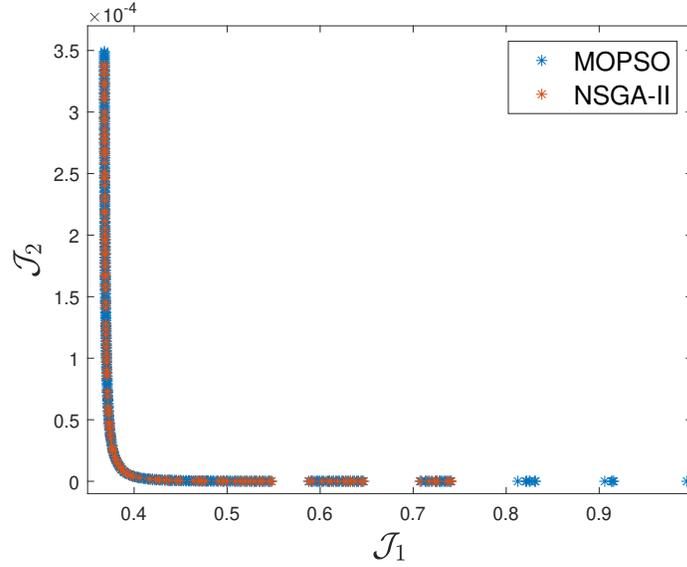


Figura 5.9: Frentes de Pareto generados por los algoritmos MOPSO y NSGA-II.

Algoritmo	ω_c^*	o^*	\mathcal{J}_1	\mathcal{J}_2	$\Phi(\mathcal{J}_1, \mathcal{J}_2)$
MOPSO	164.43	1	0.3752	1.1351×10^{-5}	0.3752
NSGA-II	157.5	1	0.3621	1.6847×10^{-5}	0.3621

Tabla 5.3: Solución del problema de optimización Multi-Objetivo generada por los algoritmos MOPSO y NSGA-II.

todo el frente de Pareto, es necesario proponer un índice que evalúe la diversidad y convergencia de las soluciones. El índice propuesto es el **hiper-volumen** \mathcal{H} [87], usando como referencia el punto $[0, 0]$. El hiper-volumen de un conjunto de soluciones corresponde a la porción del espacio objetivo que está dominada por las soluciones en el frente de Pareto \mathcal{PF} , por lo que mientras menor sea el valor calculado, esto significa que las soluciones son más cercanas al punto $[0, 0]$. Esto implica que el algoritmo metaheurístico para solucionar el problema de optimización MO genera un frente de Pareto que contiene soluciones que producen un valor más pequeño de las funciones de desempeño \mathcal{J}_1 y \mathcal{J}_2 . La media de los hiper-volumenes calculados para cada uno de los algoritmos se presenta en la Tabla 5.4.

El diagrama de caja de los hiper-volumenes calculados para los frentes de Pareto de los resultados obtenidos por el algoritmo NSGA-II y MOPSO se presentan en la Fig. 5.10. Se puede notar que el algoritmo NSGA-II produce una mayor variabilidad del hiper-volumen con una media más elevada, mientras que el algoritmo MOPSO presenta una menor variabilidad y una media de menor valor como se menciona en

Algoritmo	\mathcal{H}	$\sigma(\mathcal{H})$
MOPSO	1.6266×10^{-9}	2.4454×10^{-10}
NSGA-II	1.3688×10^{-7}	2.2794×10^{-8}

Tabla 5.4: Media y varianza de los hiper-volumenes calculados para los frentes de Pareto generados por las 50 ejecuciones del algoritmo MOPSO y NSGA-II respectivamente.

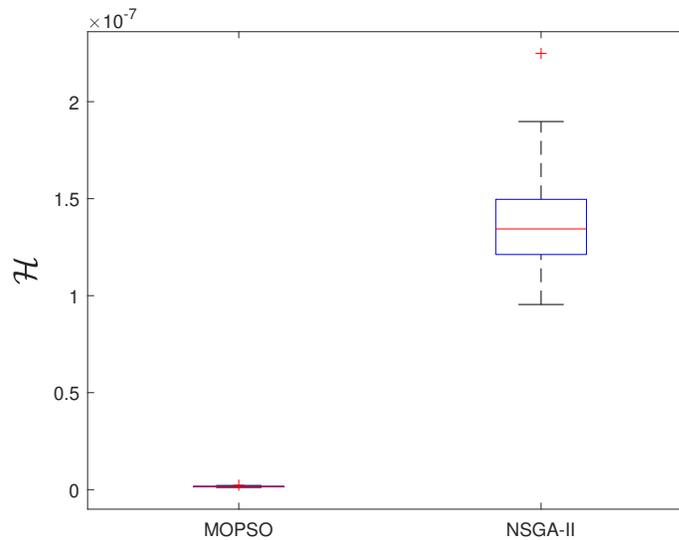


Figura 5.10: Diagrama de caja de los hiper-volumenes calculados para los frentes de Pareto.

la Tabla 5.4. Esto implica que el algoritmo MOPSO genera soluciones que logran minimizar de una mejor manera las funciones de desempeño \mathcal{J}_1 y \mathcal{J}_2 .

Dado que sólo se cotejan dos algoritmos, el análisis no-parámétrico se realizará mediante una prueba de Wilcoxon utilizando los hiper-volumenes. El p-valor obtenido es de 7.556×10^{-10} , lo cual implica que existen diferencias significativas entre el desempeño de ambos algoritmos, consecuentemente, las soluciones al problema de filtrado presentarán una variación dependiendo de que algoritmo sea utilizado para soluciones el PMO. Ya que el algoritmo MOPSO presenta un menor valor del hiper-volumen se considerará que las soluciones generadas por este último reducirán los valores de \mathcal{J}_1 y \mathcal{J}_2 , teniendo como consecuencia que el filtro sintonizado mediante este algoritmo atenuará de una mejor manera el ruido de medición sin disminuir la similitud entre la señal filtrada y la señal adquirida originalmente respecto al filtro que es sintonizado utilizando el algoritmo NSGA-II.

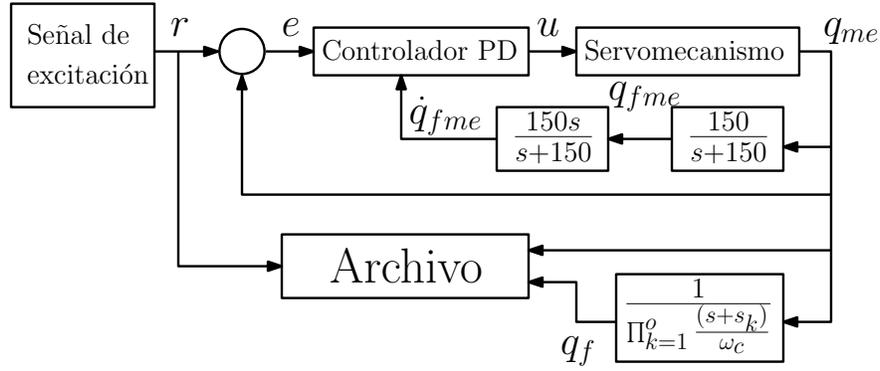


Figura 5.11: Esquema de adquisición de datos del servomecanismo para su utilización en la identificación de parámetros mediante algoritmos PSO.

5.5. Caso de estudio: Identificación de parámetros de un servomecanismo empleando señales con ruido de medición

En esta sección se realiza el proceso de identificación paramétrica del modelo del servomecanismo descrito en (5.25) donde la salida q_{me} presenta altos niveles de ruido de medición. La salida en posición del servomecanismo se procesa mediante el filtro (5.13) cuyo orden y frecuencia de corte se determinaron en la sección anterior mediante la solución del Problema de Optimización Multi-Objetivo y mediante diversas frecuencias de corte seleccionadas manualmente con fines de comparación.

5.5.1. Identificación de parámetros empleando algoritmos PSO

La identificación de los parámetros a , b , c y d del modelo (5.1) se lleva a cabo mediante los siguientes algoritmos PSO: PSO clásico (PSO-C) [10], PSO con Ponderación de Velocidad Variable (PSO-PVV) [66] y PSO con Factor de Constricción Variable (PSO-FCV) [67].

El esquema implementado para la adquisición de datos utilizados en los algoritmos metaheurísticos para la identificación de parámetros se muestra en la Fig. 5.11 considerando una señal de excitación generada por el oscilador de Duffing descrito en (4.13). Es importante mencionar que se incluye dentro de esta etapa la adquisición de la medición de posición procesada mediante el filtro (5.13).

Ya que el proceso de identificación de parámetros mediante un algoritmo metaheurístico requiere la implementación de una simulación dinámica del sistema, es necesario proponer el modelo del sistema simulado siguiente:

$$\ddot{q}_m = -\hat{a}\dot{q}_m + \hat{b}u_m - \hat{c}\text{sign}(\dot{q}_m) + \hat{d} \quad (5.34)$$

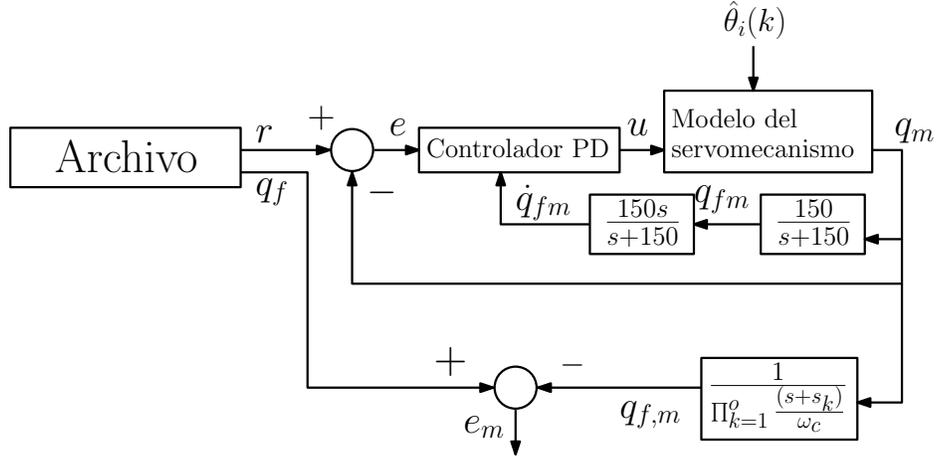


Figura 5.12: Simulación dinámica del servomecanismo utilizando datos experimentales filtrados durante el proceso de identificación de parámetros mediante algoritmos PSO.

donde la señal de control u_m se define como:

$$u_m = k_p(r - q_m) - k_d \dot{q}_{f,m} \quad (5.35)$$

Los valores de las ganancias k_p y k_d son los mismos que los que se utilizan en el controlador (5.26). El modelo virtual utilizado considera tanto la función de transferencia (5.30) utilizado para estimar la velocidad que se utiliza en el controlador (5.35) como el filtro (5.13) que se aplica a la salida del servomecanismo. El único término que no es incluido es $\eta(t)$ ya que es desconocido. El sistema virtual utilizado para la prueba de soluciones calculadas por los algoritmos metaheurísticos se presenta en la Fig. 5.12 .

5.5.2. Planteamiento del problema de optimización

El planteamiento del problema de optimización es el similar al que se presenta en la Sección 4.1.3. La función de optimización que se utiliza es (4.8) y la forma en que los parámetros son obtenidos como resultados del proceso de optimización del PSO se presenta en (4.10). La diferencia que existe respecto a lo planteado en la Sección 4.1.3 se encuentra en (4.9) ya que el rango de la perturbación \hat{d} es modificado al definir las cotas $d_{\min} = -5$ y $d_{\max} = 5$.

5.5.3. Planteamiento experimental

El servomecanismo y el equipo experimental utilizado se presenta en la Sección 5.4.3. Se realizará la identificación de parámetros mediante los algoritmos PSO previamente descritos. Para verificar la factibilidad de los parámetros identificados, estos se emplearán en el diseño de una ley de control la cual se aplica experimentalmente al servomecanismo. El desempeño del sistema en lazo cerrado se evaluará mediante varios índices de desempeño.

Con la finalidad de comparar los resultados obtenidos con un método de identificación paramétrica no metaheurístico, se propone la implementación del algoritmo CLIE [20] expresado en (4.6), cuya ley de adaptación de parámetros utiliza una matriz de ganancia $\Gamma = \text{diag}[3, 10, 1, 2]$ y cuyo regresor es definido en (4.7).

El algoritmo CLIE en su expresión (4.6) requiere utilizar el término $\epsilon_u = u_e - u$ donde las señales de control para el caso donde existe ruido de medición se definen como:

$$\begin{aligned} u &= k_p(r - q_f) - k_d\dot{q}_f \\ u_e &= k_p(r - q_{e,f}) - k_d\dot{q}_{e,f} \end{aligned} \quad (5.36)$$

donde la señal q_f y $q_{e,f}$ son obtenidas mediante la aplicación del filtro (5.13) de orden $o = 1$ y frecuencia de corte $\omega_c = 150$, que se aplica a las señales q y q_e respectivamente. Las señales \dot{q}_f y $\dot{q}_{e,f}$ son obtenidas al aplicar los filtros (5.28) y (5.30) a las señales q y q_e respectivamente. La implementación del algoritmo CLIE se muestra en la Fig. (5.13). La señal de referencia r utilizada en (5.36) es generada mediante un oscilador de Duffing (4.13) sintonizado con los parámetros $\sigma_1 = 1$, $\sigma_2 = 2$, $A_1 = 0.9$, $A_2 = 0.25$ y considerando las condiciones iniciales $z_1(0) = 0$, $z_2(0) = 0$, $z_3(0) = 0$ y $z_4(0) = 0$.

Los algoritmos PSO son implementados considerando un enjambre de 30 partículas con un número máximo de iteraciones $k_{max} = 1000$ y un criterio de paro $\alpha = 0.01$. Para evaluar la repetibilidad de los resultados de cada uno de los algoritmos metaheurísticos estos se ejecuta 30 veces. En cada ejecución se utilizaron las señales adquiridas en 3 experimentos diferentes bajo las mismas condiciones de experimentación y con la misma señal de referencia aplicada. Los parámetros estimados por cada uno de los algoritmos PSO se encuentran en la Tabla 5.5.

Los algoritmos PSO son codificados utilizando Matlab R2020a y se ejecutan en un procesador 64-bits AMD Ryzen 5 1600X con el sistema operativo Windows 10. La sintonización de los algoritmos se realiza mediante la herramienta IRACE [79] ejecutada en la paquetería Rstudio. La simulación numérica del sistema dinámico se realiza utilizando el método numérico Runge-Kutta con un periodo de integración de 0.001s. Las señales utilizadas en el algoritmo PSO son generadas mediante la aplicación del filtro (5.13) a la salida q_{me} .

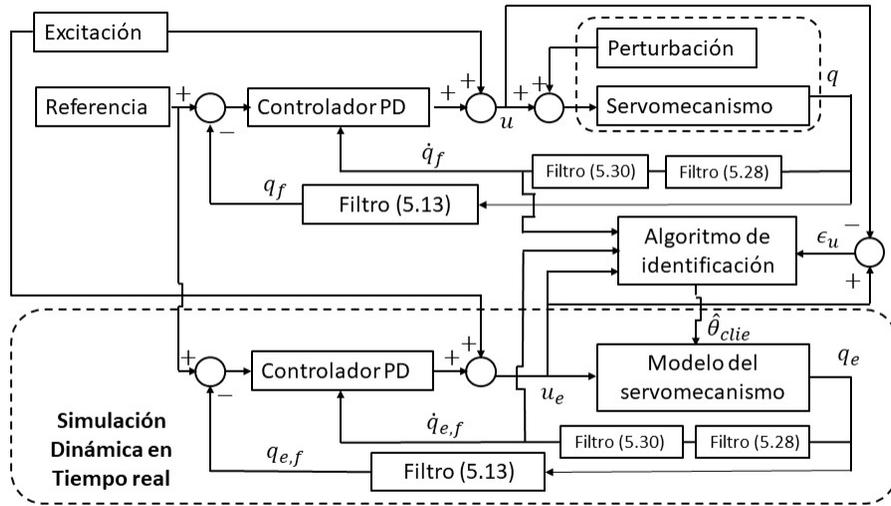


Figura 5.13: Esquema de identificación de parámetros mediante el algoritmo CLIE bajo condiciones de ruido de medición.

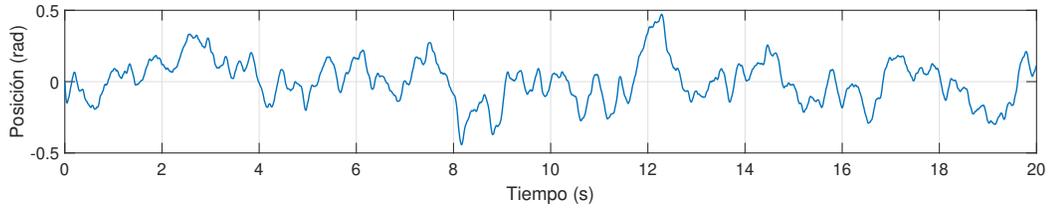
Con la finalidad de verificar los efectos del ruido y el desempeño del filtrado en la identificación paramétrica mediante algoritmos PSO se emplean varias frecuencias de corte adicionales a aquellas obtenidas como solución al problema Multi-Objetivo y que son mostradas en la Tabla 5.3. Las frecuencias son $\omega_c = 20$, $\omega_c = 150$, $\omega_c = 157.5$, $\omega_c = 164.43$, $\omega_c = 300$ y $\omega_c = 500$. Las señales resultantes son presentadas en la Fig. 5.14.

5.5.4. Análisis de los parámetros identificados

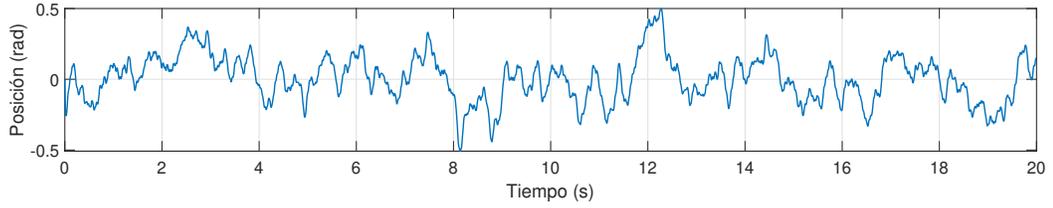
Los parámetros identificados del servomecanismo obtenidos por los algoritmos PSO son mostrados en la Tabla 5.6. En ésta se presenta la mediana de los valores obtenidos en los 30 experimentos realizados. También se presenta el índice de desempeño J y la desviación estándar de los parámetros denotada por σ .

Se puede notar que conforme se incrementa la frecuencia de corte ω_c , en los casos del algoritmo PSO-C y PSO-FCV la función de desempeño J (4.8) se incrementa excepto en el algoritmo PSO-PVV, donde el valor de J obtenido con $\omega_c = 157.5$ es mayor que el correspondiente a $\omega_c = 164.43$.

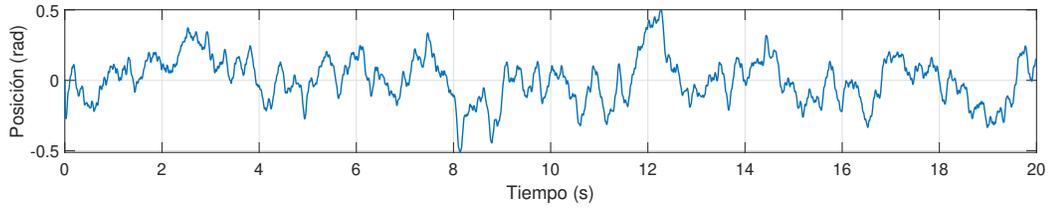
En los casos donde se empleó una frecuencia de corte $\omega_c = 20$ la función de desempeño (4.8) tiene un valor más pequeño que el obtenido con las frecuencias



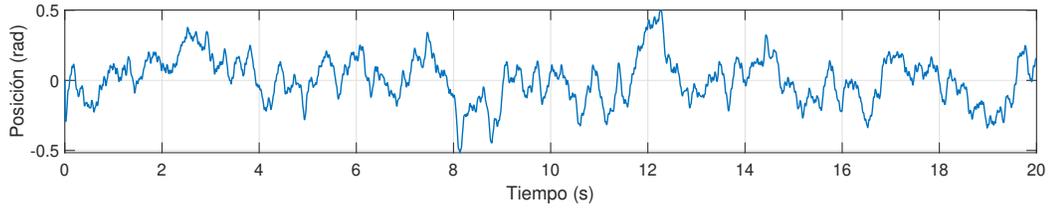
(a) Señal filtrada con (5.13) considerando $\omega_c = 20$.



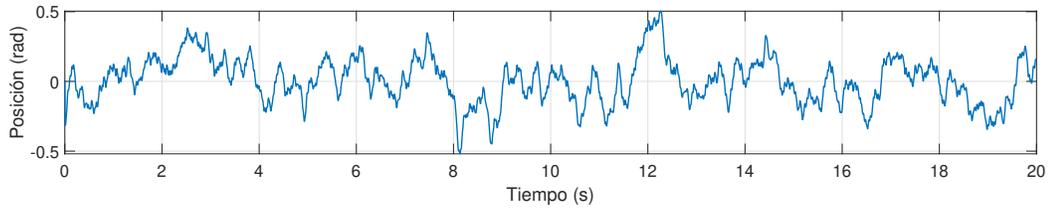
(b) Señal filtrada con (5.13) considerando $\omega_c = 150$.



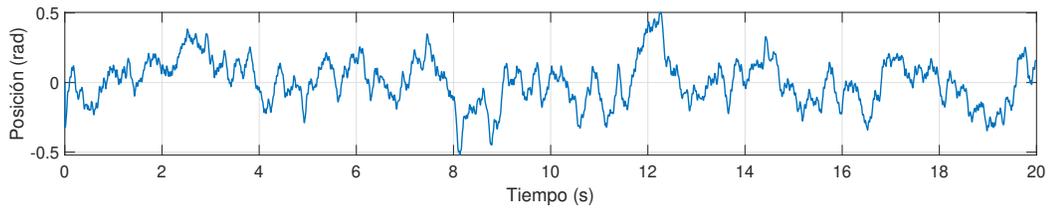
(c) Señal filtrada con (5.13) considerando $\omega_c = 157.5$.



(d) Señal filtrada con (5.13) considerando $\omega_c = 164.43$.



(e) Señal filtrada con (5.13) considerando $\omega_c = 300$.



(f) Señal filtrada con (5.13) considerando $\omega_c = 500$.

Figura 5.14: Señales filtradas de la posición del sercomecanismo mediante el filtro (5.13) para diferentes frecuencias de corte ω_c .

Algoritmo PSO	Parámetros sintonizados
PSO-C	$p = 1.0128, l = 1.6471$
PSO-PVV	$p = 1.3152, l = 1.7129,$ $\beta_{\max} = 0.8553, \beta_{\min} = 0.3541$
PSO-FCV	$p = 2.1187, l = 1.4578,$ $\gamma_{\max} = 0.9124, \gamma_{\min} = 0.7524$

Tabla 5.5: Parámetros del algoritmo PSO sintonizados mediante la herramienta IRA-CE.

de corte optimizadas por los algoritmos Multi-Objetivo. Sin embargo, los valores de los parámetros identificados en los casos donde se empleó $\omega_c = 20$ son claramente diferentes de los obtenidos mediante el algoritmo CLIE. En particular, el parámetro \hat{b} calculado por los algoritmos PSO para esta frecuencia de corte se encuentra en el rango $\hat{b} = [18.77, 22.69]$ lo cual es aproximadamente cinco veces menor que lo obtenido mediante el algoritmo CLIE, que es $\hat{b} = 107.62$.

La diferencia entre los valores de los parámetros estimados mencionada anteriormente indica que existen problemas en la identificación paramétrica aún cuando la función de desempeño J dada por (4.8) presenta el menor valor en todos los algoritmos empleando la frecuencia de corte $\omega_c = 20$. Una posible explicación de este comportamiento es que al emplear este valor las componentes de baja frecuencia de la señal filtrada se atenúan demasiado lo que produce una señal que no refleja adecuadamente la dinámica menos compleja comparada con las señales filtradas con frecuencias de corte ω_c más elevadas como se observa en la Fig. 5.14. En consecuencia, empleando $\omega_c = 20$ produce el valor más pequeño del índice de desempeño J (4.8), pero es posible que con este valor el filtrado sea demasiado agresivo y por lo tanto la señal filtrada ya no refleje adecuadamente la dinámica del servomecanismo, y que en consecuencia se produzcan parámetros estimados muy diferentes a los producidos por el algoritmo CLIE.

Por las razones expuestas anteriormente se propone la evaluación de los parámetros identificados mediante su empleo en el diseño de la ley de control por realimentación (4.14) aplicada al servomecanismo. La trayectoria del sistema en lazo cerrado se evalúa mediante los índices de desempeño IEC, IACV e IAC presentados en la Tabla 5.9. Un menor error de seguimiento de la trayectoria significaría un error paramétrico pequeño.

No.	PSO	ω_c	\hat{a}	\hat{b}	\hat{c}	\hat{d}	J	$\sigma(\hat{a})$	$\sigma(\hat{b})$	$\sigma(\hat{c})$	$\sigma(\hat{d})$
1	PSO-C	20	10.45	22.65	0.43	1.71	42.34	0.0049	0.0168	0.0188	0.0005
2	PSO-C	150	12.81	95.05	4.88	7.27	72.70	0.0955	1.0541	2.3213	0.0437
3	PSO-C	157.5	14.03	95.63	4.65	7.36	74.15	0.0864	1.0675	2.3157	0.0438
4	PSO-C	164.43	13.89	96.06	5.14	7.51	75.36	0.0874	1.1027	2.2976	0.0438
5	PSO-C	300	13.70	122.35	4.15	9.52	78.66	0.0891	1.1342	2.2785	0.0438
6	PSO-C	500	14.49	137.17	2.11	10.56	79.86	0.0273	0.1347	0.6878	0.0077
7	PSO-PVV	20	9.59	22.69	1.29	1.71	49.75	3.0441	1.2912	0.4437	0.7878
8	PSO-PVV	150	12.78	93.72	2.74	-3.69	138.1	3.4593	3.4572	1.1485	7.6421
9	PSO-PVV	157.5	13.98	97.15	3.87	-1.57	141.25	3.5123	3.5785	1.1563	7.5213
10	PSO-PVV	164.43	14.01	101.14	4.18	-1.35	139.21	3.5238	3.6875	1.1799	7.5189
11	PSO-PVV	300	22.75	126.61	4.52	-1.72	139.5	5.2711	7.6629	1.5467	4.7824
12	PSO-PVV	500	18.01	143.52	12.87	-3.02	152.6	0.5692	10.353	3.5655	4.4432
13	PSO-FCV	20	6.66	18.77	0.65	1.49	55.61	0.0001	0.1098	0.0042	0.0003
14	PSO-FCV	150	6.67	93.50	3.48	7.29	77.41	0.0001	2.6483	0.0307	0.0117
15	PSO-FCV	157.5	6.67	96.17	3.74	7.91	77.78	0.0001	2.6853	0.0312	0.0102
16	PSO-FCV	164.43	6.67	99.85	3.92	8.13	78.62	0.0001	2.7127	0.0321	0.0125
17	PSO-FCV	300	6.68	117.85	4.02	9.00	82.39	0.0001	5.5157	0.0695	0.0390
18	PSO-FCV	500	6.66	133.88	2.24	10.12	83.29	0.0035	1.4537	0.0227	0.0137
19	CLIE	150	15.01	107.62	1.59	3.14	—	—	—	—	—

Tabla 5.6: Parámetros estimados mediante los algoritmos PSO y el algoritmo CLIE empleando mediciones con ruido filtradas.

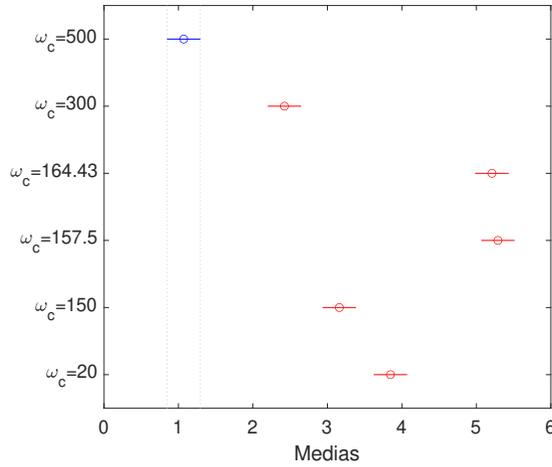


Figura 5.15: Prueba de Bonferroni basada en los índice de desempeño del PSO para las diferentes frecuencias de corte ω_c del filtro (5.13) aplicado a la señal q_{me} .

Estadística no paramétrica

Para verificar que los algoritmos PSO generan resultados diferentes cuando se modifican las frecuencias de corte de los filtros que procesan las señales de posición del servomecanismo y que estos cambios no son generados por la naturaleza aleatoria de los algoritmos PSO, es necesario realizar pruebas estadísticas no-paramétricas.

Primeramente se implementó la prueba de Friedman aplicado a los valores de la función de desempeño J obtenidos por los algoritmos PSO. Esta prueba evaluará si estos valores se modifican para valores diferentes de la frecuencia de corte ω_c . La prueba de Friedman arroja un p-valor de 13.5×10^{-219} el cual implica que existirán diferencias significativas cuando se modifica la frecuencia de corte del filtro (5.13) sin importar el tipo de algoritmo PSO empleado. Posteriormente se implementa la prueba de Kruskal-Wallis que arroja un p-valor de 5.6×10^{-218} seguido de una prueba de confianza de Bonferroni cuyos resultados se muestran en la Fig. 5.15. Se observa que las medias obtenidas mediante la prueba de Bonferroni son diferentes para casi todas las frecuencias de corte utilizadas lo que implica los resultados son diferentes basándose en el filtrado aplicado a la señal q_{me} . La única excepción corresponde a la frecuencias $\omega_c = 157.5$ y $\omega_c = 164.43$, las cuales producirán resultados similares. Este resultado no es sorprendente ya que ambas frecuencias corresponden a las soluciones al problema de optimización Multi-Objetivo de las frecuencias de corte de un filtro como se observa en la Tabla 5.3 .

Una segunda prueba de confianza de Bonferroni es implementada utilizando los parámetros estimados para determinar la existencia de diferencias entre ellos de manera individual. Los p-valores calculados mediante la prueba de Kruskal-Wallis para

cada uno de los parámetros se presentan en la Tabla 5.7, donde se observa que cada uno de ellos es menor a 0.01, por lo que el intervalo de confianza de la prueba es superior al 99%.

Parámetro estimado	P-valor
\hat{a}	3.70222×10^{-136}
\hat{b}	2.90473×10^{-179}
\hat{c}	1.02986×10^{-132}
\hat{d}	6.05284×10^{-128}

Tabla 5.7: P-valores de la prueba de Kruskal-Wallis para al análisis de cada uno de los parámetros identificados.

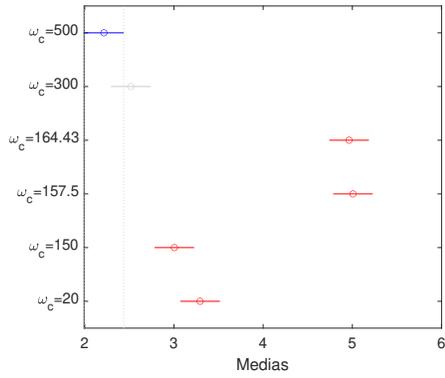
Los resultados de la pruebas de Bonferroni correspondientes se muestran en la Fig. 5.16. En el caso del parámetro \hat{a} la Fig. 5.16a muestra que los casos donde se utilizan señales filtradas empleando las frecuencias de corte optimizadas por los algoritmos Multi-Objetivo proporcionan estimados similares entre ellos y diferentes respecto a los obtenidos con el resto de las frecuencias de corte. En los casos donde $\omega_c = 500$ y $\omega_c = 300$ que corresponden a frecuencias de corte altas presentan resultados similares entre ellos pero claramente diferentes a los obtenidos con el resto de las frecuencias de corte.

Al estudiar el parámetro estimado \hat{b} se observa que todas las frecuencias inferiores a $\omega_c = 164.43$ producen resultados similares como se aprecia en la Fig. 5.16b mientras que los estimados correspondientes a las frecuencias $\omega_c = 300$ y $\omega_c = 500$ son diferentes.

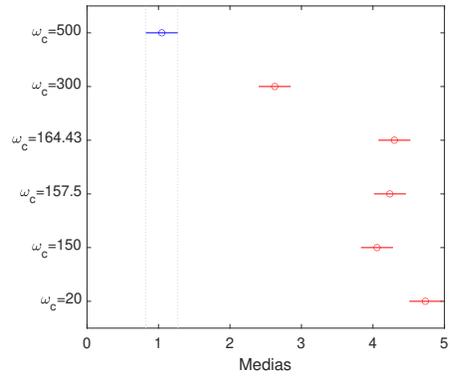
En el caso del parámetro estimado \hat{c} se observa nuevamente que cuando se utilizan las frecuencias $\omega_c = 157.5$ y $\omega_c = 164.43$ se obtienen resultados similares y claramente diferentes al resto. También se observa que no existen diferencias significativas entre los estimados correspondientes a $\omega_c = 20$, $\omega_c = 150$ y $\omega_c = 300$.

Finalmente, respecto al parámetro estimado \hat{d} se observa un comportamiento similar al observado en el parámetro estimado \hat{c} donde los estimados obtenidos con la frecuencias de corte optimizadas son similares y diferentes del resto. Se tiene que los estimados de \hat{d} obtenidos mediante las frecuencias de corte $\omega_c = 300$ y $\omega_c = 500$ presentan un alto grado de similitud por lo que se puede concluir que no existen diferencias significativas entre ellos.

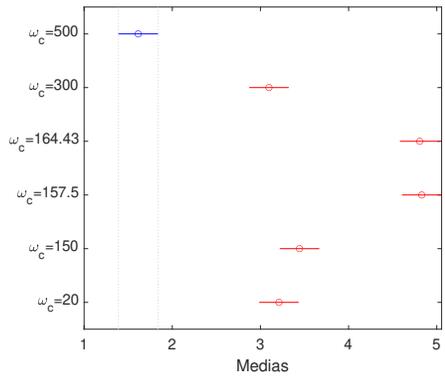
Un tercer grupo de pruebas de Bonferroni basadas en el valor de J son realizadas con la finalidad de determinar si los resultados obtenidos varían de acuerdo al algoritmo PSO utilizado para las diferentes frecuencias de corte ω_c . Los p-valores de



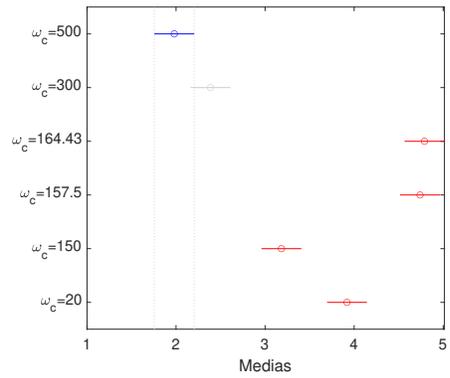
(a) Prueba de Bonferroni basada en los estimados del parámetro \hat{a} .



(b) Prueba de Bonferroni basada en los estimados del parámetro \hat{b} .



(c) Prueba de Bonferroni basada en los estimados del parámetro \hat{c} .



(d) Prueba de Bonferroni basada en los estimados del parámetro \hat{d} .

Figura 5.16: Pruebas de Bonferroni basada en los parámetros estimados con el algoritmo PSO respecto a las frecuencias de corte ω_c .

las pruebas de Kruskal-Wallis correspondientes se presentan en la Tabla 5.8 donde se puede constatar que el intervalo de confianza de dichas pruebas es superior al 99 %.

Frecuencia de corte (ω_c)	P-valor
20	3.64269×10^{-32}
150	2.49925×10^{-41}
157.5	6.82887×10^{-43}
164.43	2.49925×10^{-41}
300	1.73088×10^{-42}
500	6.82421×10^{-42}

Tabla 5.8: P-valores de las pruebas de Kruskal-Wallis basadas en el tipo de algoritmo PSO para las diferentes frecuencias de corte.

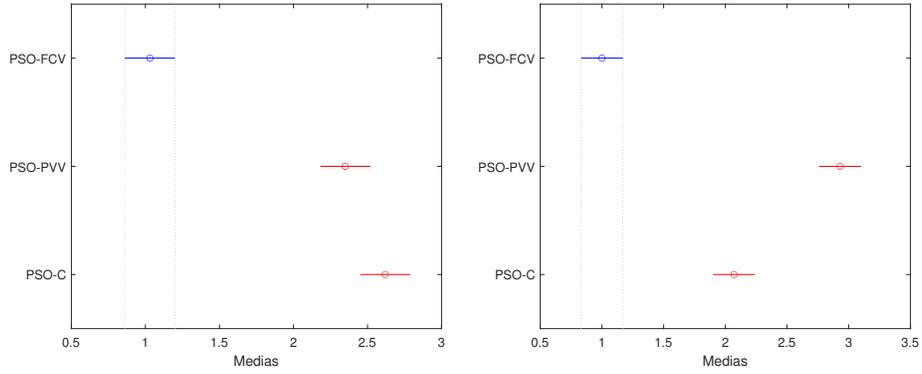
En la Fig. 5.17 se presentan las pruebas de Bonferroni para cada una de las frecuencias de corte utilizadas en el estudio. Se puede notar que para todos los casos excepto cuando $\omega_c = 20$ los resultados presentan variaciones diferentes de acuerdo al algoritmo utilizado para la identificación de parámetros. En el caso $\omega_c = 20$ los algoritmos PSO-C y PSO-PVV proporcionan resultados similares entre si y diferentes de los obtenidos mediante el algoritmo PSO-FCV.

Análisis estadístico de los parámetros

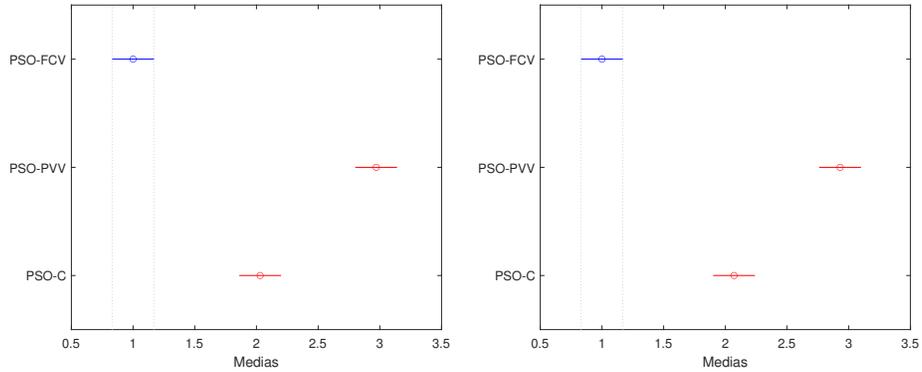
El desempeño de los algoritmos PSO respecto a los parámetros obtenidos se presenta en la Tabla 5.6 y se utilizará el parámetro \hat{b} . El término \hat{b} corresponde a la ganancia del servomecanismo por lo que un error de identificación produce un incremento del error en una ley de control donde se utilizó este estimado.

Se puede notar que el valor de \hat{b} disminuye conforme disminuye la frecuencia de corte ω_c . Por otro lado, se observa que los valores de \hat{b} son similares para $\omega_c = 150$, $\omega_c = 157.5$ y $\omega_c = 164.43$. Este comportamiento se puede observar independientemente del algoritmo aplicado para la identificación de los parámetros.

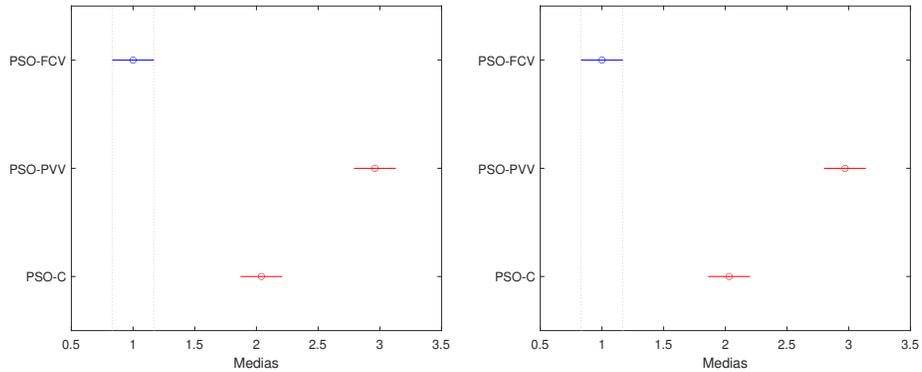
Es importante mencionar que el algoritmo PSO-C es el que presenta la menor variabilidad $\sigma(\hat{b})$ comparado con el resto de los algoritmos para todas las frecuencias ω_c mientras que el algoritmo PSO-PVV es el que presenta una mayor variabilidad $\sigma(\hat{b})$ independientemente de la frecuencia de corte ω_c .



(a) Prueba de Bonferroni basada en los resultados obtenidos con la frecuencia de corte $\omega_c = 20$. (b) Prueba de Bonferroni basada en los resultados obtenidos con la frecuencia de corte $\omega_c = 150$.



(c) Prueba de Bonferroni basada en los resultados obtenidos con la frecuencia de corte $\omega_c = 157.5$. (d) Prueba de Bonferroni basada en los resultados obtenidos con la frecuencia de corte $\omega_c = 164.43$.



(e) Prueba de Bonferroni basada en los resultados obtenidos con la frecuencia de corte $\omega_c = 300$. (f) Prueba de Bonferroni basada en los resultados obtenidos con la frecuencia de corte $\omega_c = 500$.

Figura 5.17: Pruebas de Bonferroni respecto a los algoritmos PSO basadas en la función de desempeño J , para cada una de las frecuencias de corte ω_c .

5.5.5. Implementación de los parámetros identificados en una ley de control por realimentación aplicada al servomecanismo

La evaluación de los parámetros estimados obtenidos mediante los algoritmos PSO se realizará mediante su empleo en el diseño de una ley de control. La ley de control aplicada al servomecanismo es (4.14), la cual al ser aplicada en el sistema ideal del servomecanismo (4.1) produce la dinámica (4.17). En la Sección 4.3 se muestra que el error de posición y su derivada convergen a una vecindad alrededor del origen, el tamaño de la vecindad disminuye si lo hace el error paramétrico $\hat{\theta} - \theta$.

La evaluación de los parámetros estimados producidos por los algoritmos PSO se realiza mediante los índices de desempeño IEC (2.28), IAC (2.29) e IACV (2.30). Estos resultados se presentan en la Tabla 5.9 en donde también se evalúan los parámetros producidos por el algoritmo CLIE.

La implementación experimental de (4.14) considera los parámetros $\omega_n = 30\pi$ y $\xi = 1$ para definir la dinámica que se desea imponer al servomecanismo. A su vez, la señal de referencia es creada al combinar una señal cuadrada con frecuencia de 0.3 Hz y una amplitud de 2.7 rad con una señal constante de -1.7 rad . La señal de referencia resultante es filtrada mediante:

$$F(s) = \frac{20}{s + 20} \quad (5.37)$$

La ley de control se implementa utilizando un tiempo de muestro de $0.001s$ y el método de integración numérica de Euler.

Los resultados se presentan en la Tabla 5.9 la cual contiene los índices calculados para cada conjunto de parámetros estimados mediante los algoritmos PSO y el algoritmo CLIE. Se puede notar que el valor mínimo de IEC es obtenido cuando se utilizan los parámetros calculados con el algoritmo PSO-C y una frecuencia de corte $\omega_c = 164.43$. Esto muestra que el seguimiento es mejor en el sentido del índice IEC cuando se emplean las frecuencias de corte ω_c optimizadas mediante los algoritmos Multi-Objetivo. Este comportamiento se repite en el caso del algoritmo PSO-PVV donde el valor mínimo de IEC es obtenido cuando se utiliza la frecuencia de corte optimizada $\omega_c = 157.5$ y para el caso del algoritmo PSO-FCV donde la frecuencia de corte optimizada es $\omega_c = 164.43$.

Al analizar los índices IAC e IACV se puede notar que conforme se utilizan frecuencias de corte ω_c de mayor valor estos índices disminuyen. Esto se encuentra directamente relacionado con el valor de \hat{b} , ya que al incrementarse éste la señal de control disminuirá de amplitud.

La Fig. 5.18 muestra el seguimiento de trayectoria en el servomecanismo al utili-

No.	PSO	ω_c	<i>IEC</i>	<i>IACV</i>	<i>IAC</i>
1	PSO-C	20	2273	6221.6	31623
2	PSO-C	150	2279.9	1224.4	7237.4
3	PSO-C	157.5	2228.1	1140.1	7052.2
4	PSO-C	164.43	2201.1	1123.1	6871.9
5	PSO-C	300	2281.8	919.7	4613.6
6	PSO-C	500	2336.1	809.6	5302.2
7	PSO-PVV	20	2506.5	6755.5	33993
8	PSO-PVV	150	2334	1324.5	7518.1
9	PSO-PVV	157.5	2293.6	1298.3	6842.3
10	PSO-PVV	164.43	2298.7	1284.2	6941.2
11	PSO-PVV	300	2366.9	609.1	4673.2
12	PSO-PVV	500	2398.8	902.9	5663.8
13	PSO-FCV	20	2447.1	8568.6	40294
14	PSO-FCV	150	2429.6	1591.9	7870.4
15	PSO-FCV	157.5	2385.2	1463.2	6125.3
16	PSO-FCV	164.43	2378.2	1472.9	6097.5
17	PSO-FCV	300	2407.4	1032.8	5129.5
18	PSO-FCV	500	2435.9	844.9	5787.3
19	CLIE	150	2323.8	998.7	6527.4

Tabla 5.9: Evaluación de los parámetros estimados mediante experimentos de control en lazo cerrado.

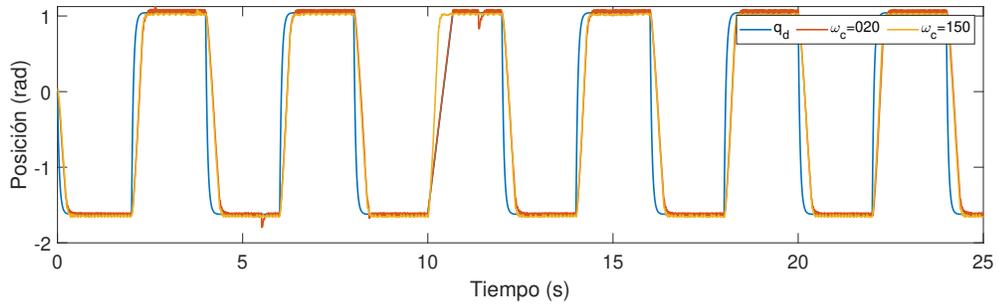
zar los parámetros estimados calculados con el algoritmo PSO-C el cual emplea datos procesados con filtros con diferentes frecuencias de corte ω_c . Tanto en la Fig. 5.18a como en la Fig. 5.18b se aprecia que se realiza la tarea de seguimiento de la referencia. Sin embargo, al revisar en detalle el seguimiento, se observan diferencias en las respuestas como se aprecia en las Fig. 5.18c y 5.18d. Cuando se utilizan parámetros estimados con los datos filtrados con una frecuencia de corte $\omega_c = 20$, el seguimiento de trayectoria presenta una gran cantidad de oscilaciones. Cuando se utilizan $\omega_c = 150$ y $\omega_c = 164.43$ se observa menos oscilaciones. La menor cantidad de oscilaciones se presenta en el caso $\omega_c = 500$.

Al observar los índices IEC, IAC e IACV mostrados en la Tabla 5.9 se puede notar que cuando se utilizan las frecuencias de corte ω_c del filtro optimizadas mediante el algoritmo Multi-Objetivo el índice IEC presenta los menores valores lo que indica un mejor seguimiento de la trayectoria, con un esfuerzo de control moderado medido mediante los índices IAC e IACV.

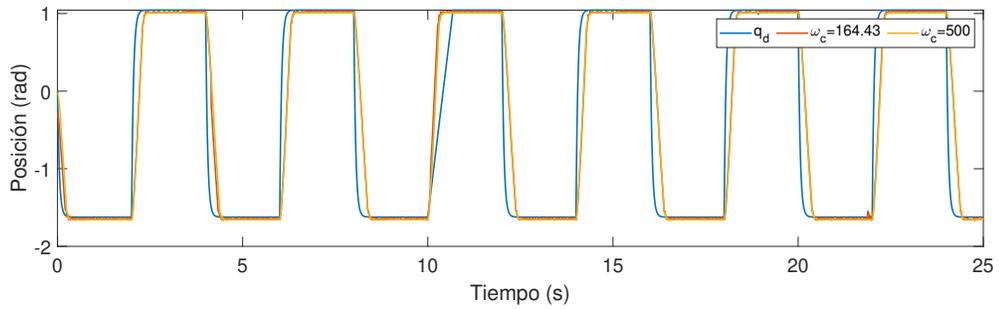
5.6. Conclusiones

Basándose en los resultados obtenidos en este capítulo se puede concluir lo siguiente:

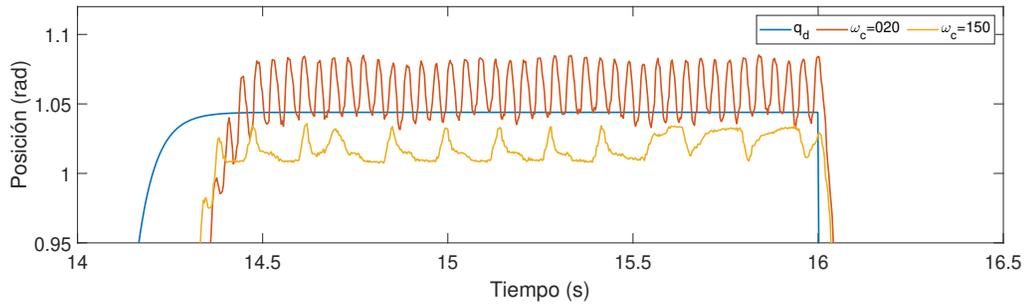
- La estimación de parámetros mediante algoritmos metaheurísticos se ve afectado por el ruido de medición.
- La sintonización de filtros empleados en el acondicionamiento de las señales utilizadas en la identificación de parámetros necesita ser realizada de forma tal que se atenúen las componentes de alta frecuencia del ruido sin modificar las componentes frecuenciales asociadas a la dinámica del sistema bajo identificación.
- El proceso de sintonización de los filtros puede considerarse como un problema de optimización Multi-Objetivo considerando los coeficientes de correlación de Pearson ρ y la relación señal ruido *RSR*.
- Los parámetros estimados mediante los algoritmos PSO son cercanos a los obtenidos con el algoritmo CLIE cuando se utiliza una frecuencia de corte obtenida mediante optimización Multi-Objetivo.
- Los parámetros estimados mediante los algoritmos PSO, y que utilizan señales acondicionadas con filtros que emplean frecuencias de corte optimizadas mediante algoritmos Multi-Objetivo, proporcionan los menores valores del índice



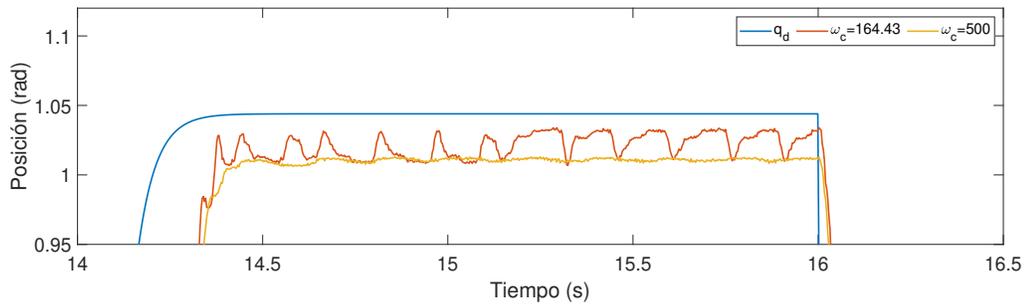
(a) Respuesta experimental en el intervalo de tiempo de $0s - 25s$ al utilizar parámetros estimados con señales filtradas con las frecuencias de corte $\omega_c = 20$ y $\omega_c = 150$.



(b) Respuesta experimental en el intervalo de tiempo de $0s - 25s$ al utilizar parámetros estimados con señales filtradas con las frecuencias de corte $\omega_c = 164.43$ y $\omega_c = 500$.



(c) Respuesta experimental en el intervalo de tiempo de $14s - 16.5s$ al utilizar parámetros estimados con señales filtradas con las frecuencias de corte $\omega_c = 20$ y $\omega_c = 150$.



(d) Respuesta experimental en el intervalo de tiempo de $14s - 16.5s$ al utilizar parámetros estimados con señales filtradas con las frecuencias de corte $\omega_c = 164.43$ y $\omega_c = 500$.

Figura 5.18: Seguimiento de trayectoria utilizando parámetros estimados mediante el algoritmo PSO-C.

IEC el cual mide la calidad de seguimiento, y además proporcionan un compromiso adecuado entre el error de seguimiento y el nivel y la variación de la señal de control, estos últimos expresados mediante los índices IAC e IACV.

Capítulo 6

Aplicación de funciones de penalización en la sintonización de observadores de estado extendido utilizando PSO

Los estados de un sistema dinámico no siempre se encuentran disponibles, ya sea por falta de instrumentación o por la naturaleza misma del sistema. Esto implica que los controladores basados en realimentación de estados no siempre pueden ser implementados. Una manera de estimar los estados es a través de la implementación de un observador que utiliza el error entre la salida medida y la salida generada por el propio observador. Por ejemplo, el observador de Luenberger [88] requiere para su implementación del conocimiento completo de la estructura del sistema dinámico que se desea observar y de sus parámetros. Esta información no siempre está disponible en la práctica y además la planta puede estar sujeta a perturbaciones desconocidas. Para hacer frente a este problema se han propuesto los Observadores de Estado Extendido (OEE), donde las perturbaciones y los términos desconocidos del sistema son agrupados como una perturbación, esta última es estimada por el observador junto con las variables de estado del sistema.

Los métodos de sintonización de las ganancias de los OEE mediante técnicas de optimización basadas en algoritmos metaheurísticos reportados en la literatura no toman en cuenta la estabilidad del observador como parte del problema de optimización como se menciona en la Sección 1.3. En [62] este es resuelto mediante una parametrización de las ganancias del observador respecto a la frecuencia natural no amortiguada en el caso de un polinomio característico de segundo orden. Este enfoque produce únicamente polos repetidos en el polinomio característico del observador, lo cual puede limitar su flexibilidad. Por esta razón es necesario examinar el desempeño de las técnicas metaheurísticas en la sintonización de los OEE cuando se modifica el problema de optimización en el caso de observadores con un polinomio característico

con polos estables no necesariamente repetidos.

En este capítulo se presentarán tres planteamientos del problema de sintonización de las ganancias de un OEE que emplean la optimización basada en el algoritmo PSO, y que toman en cuenta las condiciones de estabilidad del OEE. Para mostrar el desempeño de los planteamientos se emplea un caso de estudio numérico que consiste en un péndulo actuado sujeto a perturbaciones externas.

6.1. Observador de Estado Extendido (OEE)

Considérese un sistema dinámico de segundo orden de una entrada y una salida perturbado [57]:

$$\begin{aligned}\dot{y}_1 &= y_2 \\ \dot{y}_2 &= u + \xi(t)\end{aligned}\tag{6.1}$$

donde $y = [y_1, y_2]^\top$ y u son el estado y la señal de control respectivamente y y_1 es la salida medible. El término $\xi(t)$ es una perturbación continua y uniformemente acotada que cumple con:

$$|\xi(t)| \leq \alpha_0, \quad |\dot{\xi}(t)| \leq \alpha_1\tag{6.2}$$

para $\alpha_1, \alpha_2 \in \mathbf{R}^+$.

Un OEE para el sistema (6.1) posee la siguiente dinámica:

$$\begin{aligned}\dot{\hat{y}}_1 &= \hat{y}_2 + \lambda_0 e \\ \dot{\hat{y}}_2 &= u + \hat{z} + \lambda_1 e \\ \dot{\hat{z}} &= \lambda_2 e\end{aligned}\tag{6.3}$$

donde el término \hat{z} es un estimado de $\xi(t)$, y $\lambda_0, \lambda_1, \lambda_2 \in \mathbf{R}^+$ son las ganancias del observador que deben ser sintonizadas. El error de estimación se define como $e = y_1 - \hat{y}_1$ cuya dinámica corresponderá a:

$$\ddot{e} + \lambda_0 \dot{e} + \lambda_1 e + \lambda_2 e = \dot{\xi}(t)\tag{6.4}$$

El polinomio característico de la dinámica del error es:

$$s^3 + \lambda_0 s^2 + \lambda_1 s + \lambda_2 = 0\tag{6.5}$$

La dinámica del error de estimación (6.4) será estable si los polos de (6.5) poseen

parte real negativa y si se cumple la siguiente relación:

$$\lambda_0 \lambda_1 > \lambda_2 \quad (6.6)$$

Una de las maneras de sintonizar las ganancias de este observador es la siguiente:

$$\lambda_0 = \frac{2\zeta\omega_n + p}{\epsilon}, \quad \lambda_1 = \frac{2\zeta\omega_n p + \omega_n^2}{\epsilon^2}, \quad \lambda_2 = \frac{\omega_n^2 p}{\epsilon^3} \quad (6.7)$$

donde $\omega_n \in \mathbf{R}^+$ es la frecuencia natural deseada del observador, $\zeta \in \mathbf{R}^+$ es el término de amortiguamiento y $p \in \mathbf{R}^+$ es un término constante. El ancho de banda de este observador se propone utilizando $\epsilon \in \mathbf{R}^+$. El ancho de banda se incrementa disminuyendo el valor de ϵ . Sin embargo, valores pequeños de ϵ combinados con errores de estimación inicial del estado elevados producen el fenómeno de **peaking**, [89] el cual implica la presencia de valores transitorios elevados en la estimación de los estados. Sintonizaciones similares para sistemas de orden mayor empeoran el problema de **peaking** al incrementarse la potencia a la cual es elevado ϵ . Otro problema bien conocido respecto a este tipo de observadores es la presencia de ruido de medición. Un incremento en el orden del OEE amplifica el ruido de medición presente en la salida y_1 , que se traduce en una degradación de los estados estimados por el observador [90].

6.2. Sintonización de las ganancias del OEE como un problema de optimización

La sintonización del OEE mediante el uso de algoritmos metaheurísticos requiere el cumplimiento de las condiciones siguiente:

- El polinomio (6.5) debe satisfacer los criterios de estabilidad de Routh-Hurwitz [91].
- El error de estimación e debe tender a cero o a una vecindad cercana a cero.

Con la finalidad de aplicar el algoritmo PSO para este tipo de problemas se definirá a cada partícula x_i como:

$$x_i = [\lambda_0, \lambda_1, \lambda_2]^T \quad (6.8)$$

Por otra parte, es necesario definir el conjunto Ω de soluciones factibles x_i tal que se cumplan las siguientes restricciones:

$$\Omega := \{x_i \in \mathbb{R}^3 \mid g_1(x) \leq 0, \dots, g_{r_i}(x) \leq 0, \\ h_1(x) = 0, \dots, h_{r_e}(x) = 0\} \quad (6.9)$$

donde las funciones $g_i(x)$ corresponden a r_i restricciones de desigualdad, mientras que $h_i(x)$ corresponden a r_e restricciones de igualdad. Mas aún, la definición de Ω para este problema puede incluir restricciones para cada componente λ_i de la partícula x_i tal que se tengan cotas superiores e inferiores de ellas, por ejemplo $\underline{\lambda}_i \leq \lambda_i \leq \bar{\lambda}_i, i = 1, 2, 3$. Además, la evaluación de un índice de desempeño en este caso requiere la ejecución de una simulación dinámica del OEE dada por:

$$\begin{bmatrix} \dot{\hat{y}}_1 \\ \dot{\hat{y}}_2 \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \hat{y}_2 \\ u + z \\ 0 \end{bmatrix} + x_i e \quad (6.10)$$

Se analizarán tres métodos para plantear el problema de optimización numérica para la sintonización de ganancias con la finalidad de determinar su pertinencia:

- Sintonización empleando cotas inferiores de las ganancias del observador [58, 92, 59].
- Sintonización empleando cotas inferiores y superiores de las ganancias del observador [62].
- Empleo de funciones de penalización que son propuestas en este trabajo.

6.2.1. Sintonización considerando cotas inferiores (CI)

El problema de optimización numérica cuando se consideran únicamente cotas inferiores (CI) se puede describir mediante el uso de la siguiente función de desempeño:

$$J_1(x_i) = \int_0^t |y_1(\tau) - \hat{y}_1(\tau)| d\tau \quad (6.11)$$

donde $y(t)$ y $\hat{y}(t)$ son la salida medida y la estimación obtenida por la simulación dinámica respectivamente. El conjunto de soluciones factibles Ω_1 que determina las ganancias del observador se define como:

$$\Omega_1 = \{x_i | 0 < \underline{\lambda}_i \leq \lambda_i, i = 1, 2, 3\} \quad (6.12)$$

donde $\underline{\lambda}_i$ es una cota inferior de la ganancia λ_i , enfoque que se utiliza en [58, 92, 59].

Esta manera de definir el problema de optimización numérica puede producir ganancias de observador que generen un OEE inestable. Consecuentemente, una simulación numérica del sistema (6.10) que presente un comportamiento inestable generará

un problema numérico en la implementación del algoritmo PSO impidiendo el correcto funcionamiento del mismo. Esto sucede si la combinación de ganancias λ en la solución x_i produce raíces con parte real positiva en el polinomio (6.5). En estos casos el valor estimado de $\hat{y}_1(t)$ converge teóricamente hacia valores infinitos por lo que los métodos numéricos utilizados para simular al sistema dinámico interrumpen su funcionamiento haciendo imposible completar la simulación.

6.2.2. Sintonización considerando cotas inferiores y superiores (CIS)

Una manera de evitar los problemas de estabilidad relacionados a la sintonización del OEE es mediante el empleo simultáneo de cotas inferiores y superiores (CIS) de las ganancias del observador [62] dentro de la definición del conjunto de soluciones factibles, esto es:

$$\Omega_2 = \{x_i | 0 < \underline{\lambda}_i \leq \lambda_i \leq \bar{\lambda}_i, i = 1, 2, 3\} \quad (6.13)$$

Los términos $\underline{\lambda}_i$ y $\bar{\lambda}_i$ corresponden respectivamente a las cota superior e inferior de la ganancia λ_i . Para calcular las cotas, los parámetros ξ , p y ϵ en (6.7) se mantienen fijos y se proponen cotas para ω_n en el intervalo $\underline{\omega}_n \leq \omega_n \leq \bar{\omega}_n$. Empleando (6.7) se pueden calcular los valores de $\underline{\lambda}_0$, $\underline{\lambda}_1$, y $\underline{\lambda}_2$ mediante el uso de $\underline{\omega}_n$, ξ , p , y ϵ . De forma similar, utilizando $\bar{\omega}_n$, ξ , p , y ϵ es posible calcular las cotas superiores $\bar{\lambda}_0$, $\bar{\lambda}_1$ y $\bar{\lambda}_2$. Conviene mencionar que la función de desempeño para este problema es (6.11).

Esta forma de definir el problema de optimización incrementa la capacidad de obtener soluciones estables. Pero a su vez, se presenta la posibilidad de que Ω_2 no contenga el mínimo global debido a las cotas impuestas, provocando que las soluciones en Ω_2 sean sub-óptimas. Además, es posible que los valores x_i generados en la búsqueda produzcan polos inestables al no cumplirse las condiciones de estabilidad de Routh-Hurwitz.

6.2.3. Sintonización utilizando funciones de penalización (FP)

Considerando los dos métodos anteriores se puede notar la necesidad de plantear el problema de optimización tal que las soluciones continúen generando simulaciones dinámicas del OEE estables pero no que se encuentren limitadas a un conjunto específico. Debido a lo anterior, se propone en este trabajo el uso de funciones de penalización (FP) en la función de desempeño (6.11). La expresión para la nueva función de desempeño es:

$$J_2(x_i) = \int_0^t |y_1(\tau) - \hat{y}_1(\tau)| d\tau + \kappa \sum_{j=1}^3 e^{\eta_j \mathbb{R}[s_j]} \quad (6.14)$$

Las funciones de penalización corresponden a los términos $e^{\eta_j \mathbb{R}[s_j]}$ los cuales son positivos. El parámetro $\kappa \in \mathbb{R}^+$ pondera a las funciones de penalización dentro de la función de costo. La constante $\eta_j \in \mathbb{R}^+$ pondera al término $\mathbb{R}[\cdot]$ que es una función que extrae la parte real de un número complejo, mientras que $s_j, j = 1, 2, 3$ son las raíces del polinomio (6.5) las cuales permiten reescribirlo como:

$$s^3 + \lambda_0 s^2 + \lambda_1 s + \lambda_2 = (s + s_1)(s + s_2)(s + s_3) = 0 \quad (6.15)$$

Algoritmo 6: Optimización por Enjambre de Partículas (PSO) para la sintonización de ganancias del OEE utilizando el método de Funciones de Penalización

Datos: $J(\cdot)$, Ω_3 y Ω_4
Resultado: x_*

- 1 Generar soluciones aleatorios $x_i(1) \in \Omega_3 \quad i = 1, \dots, r$;
- 2 Evaluar las soluciones $x_i(1)$ usando $J(\cdot)$ mediante (6.10) y (6.15) ;
- 3 Obtener los términos $x_{i,*}(1)$ y $x_*(1)$;
- 4 Calcular las velocidades $\omega_i(2)$ y las posiciones $x_i(2)$ tal que estas últimas pertenezcan a Ω_4 ;
- 5 Colocar $k = 2$;
- 6 **Mientras** $k < k_{max}$ **hacer**
- 7 Evaluar las soluciones $x_i(k)$ usando $J(\cdot)$ mediante (6.10) y (6.15);
- 8 Obtener los términos $x_{i,*}(k)$ and $x_*(k)$;
- 9 **Si** $J(x_*(k)) \geq \alpha$ **entonces**
- 10 Calcular las velocidad $\omega_i(k)$ y las posiciones $x_i(k)$;
- 11 **Mientras** $x_i(k) \notin \Omega_4$ **hacer**
- 12 Decrementar la velocidad tal que $\omega_i(k) = \frac{\omega_i(k)}{2}$;
- 13 Calcular una nueva posición $x_i(k)$ using $\omega_i(k)$;
- 14 **Fin**
- 15 **En otro caso**
- 16 **Detenerse**;
- 17 **Fin**
- 18 $k = k + 1$;
- 19 **Fin**

El **Algoritmo 6** describe la optimización empleando el algoritmo PSO para la sintonización de ganancias del OEE.

La idea de la utilización de las funciones de penalización es la siguiente. Considérese la solución $x_i = [\lambda_0, \lambda_1, \lambda_2]^T$. Al calcular los polos s_j del polinomio (6.15) empleando x_i y obtener sus partes reales, si $\mathbb{R}[s_j] < 0, j = 0, 1, 2$, esto implica que el polinomio es Hurwitz estable y la dinámica del OEE es exponencialmente esta-

ble. Además, se cumple $0 < e^{\eta_j \mathbb{R}[s_j]} < 1$ y entonces el índice de desempeño $J_2(x_i)$ decrementa su valor. Por otra parte, si $\mathbb{R}[s_j] > 0$ para al menos uno de los polos s_j , entonces $1 < e^{\eta_j \mathbb{R}[s_j]}$. Lo anterior tiene como consecuencia que el polinomio (6.15) deje ser Hurwitz estable y por lo tanto la dinámica del OEE es inestable. Además, la función de desempeño $J_2(x_i)$ se incrementa. En consecuencia, el uso de las funciones de penalización en el índice de desempeño $J_2(x_i)$ permite descartar las soluciones x_i que generan dinámicas inestables del OEE durante la ejecución del algoritmo PSO.

Durante la primera iteración del algoritmo PSO es necesario imponer las condiciones (6.6) obtenidas por el criterio de estabilidad de Routh-Hurwitz mediante la consideración del siguiente conjunto de soluciones factibles:

$$\Omega_3 = \{x_i | 0 < \lambda_i \leq \bar{\lambda}_i, i = 1, 2, 3 | \lambda_0 \lambda_1 > \lambda_2\} \quad (6.16)$$

Esto con la finalidad de asegurar la estabilidad de los polos del OEE al inicio del proceso de optimización. Una vez que se inicia el cálculo de las nuevas soluciones, a partir de la segunda iteración del algoritmo la estabilidad de las soluciones se mantiene mediante la implementación de las funciones de penalización, ya que éstas provocan que las nuevas ganancias al ser aplicadas en (6.5) no den lugar a polos con parte real positiva.

Se puede proponer un nuevo conjunto de restricciones para que las ganancias del observador no tomen valores elevados y así poder presentar comparaciones justas con los otros métodos de sintonización. Estas restricciones se definen como:

$$\Omega_4 = \{x_i | \lambda_i \leq \bar{\lambda}_i, i = 1, 2, 3 | \lambda_0 \lambda_1 > \lambda_2\} \quad (6.17)$$

El conjunto Ω_4 puede se propone para limitar las altas ganancias que se podrían obtener mediante la metodología aplicada, ya que al no existir cotas superiores las ganancias pueden incrementarse hasta que se termine de ejecutar el **Algoritmo 6**.

El conjunto Ω_4 sólo considera las cotas superiores $\bar{\lambda}_i$ para limitar las ganancias del observador, ya que la estabilidad de las soluciones se mantiene mediante las funciones de penalización empleadas en (6.14).

El empleo de funciones de penalización para sintonizar las ganancias de un OEE puede ser extendido al caso de orden n orden. La función de desempeño correspondiente es:

$$J_3(x_i) = \int_0^t |y_1(\tau) - \hat{y}_1(\tau)| d\tau + \kappa \sum_{j=1}^n e^{\eta \mathbb{R}[s_j]} \quad (6.18)$$

6.3. Caso de estudio: Péndulo actuado sujeto a perturbaciones exógenas

Para evaluar la sintonización del OEE se considerará el caso de un péndulo actuado que está sujeto a una perturbación exógena cuya dinámica está descrita por:

$$\begin{aligned}\ddot{y} &= -\frac{g}{L}\sin(y) + \zeta(t) + u \\ &= u + \xi(t)\end{aligned}\tag{6.19}$$

donde y es la posición angular del péndulo, u es la señal de control, g es la aceleración de gravedad, L la longitud del péndulo, $\frac{g}{L}\sin(y)$ el par gravitacional el cual se supone desconocido y $\zeta(t) = 0.5(1 + e^{-\sin(3t)\sin(t)})\cos(0.5t)$ una perturbación exógena acotada. El término $\xi(t) = -\frac{g}{L}\sin(y) + \zeta(t)$ contiene la perturbación exógena y el par gravitacional.

La entrada u se define como:

$$u = \begin{cases} 0 & t < 2 \\ 1 & t \geq 2 \end{cases}\tag{6.20}$$

El péndulo actuado se representa en el espacio de estados como (6.1), por lo que el OEE para este sistema es:

$$\begin{aligned}\dot{\hat{y}}_1 &= \hat{y}_2 + \lambda_0(y - \hat{y}_1) \\ \dot{\hat{y}}_2 &= u + \hat{z} + \lambda_1(y - \hat{y}_1) \\ \dot{\hat{z}} &= \lambda_2(y - \hat{y}_1)\end{aligned}\tag{6.21}$$

donde \hat{z} es el estimado de $\xi(t)$.

6.3.1. Planteamiento experimental

La implementación de los algoritmos se realiza utilizando el programa Matlab 2020b ejecutado en una computadora con un procesador de 64 bits AMD Ryzen 5 1600X. La simulación del péndulo y el OEE es ejecutada en Simulink utilizando un tiempo de muestreo de 0.001s con el método numérico de integración Runge-Kutta. La sintonización de los parámetros del algoritmo PSO clásico se realiza fuera de línea mediante la herramienta IRACE [79] ejecutada en el programa RStudio. Los valores obtenidos son $p = 1.0128$ y $l = 1.6471$.

Las cotas para el método CI son propuestas de forma manual por el usuario, en

este caso se propusieron las siguientes:

$$\Omega_1 = \begin{cases} 40 \leq \lambda_0 \\ 500 \leq \lambda_1 \\ 3200 \leq \lambda_2 \end{cases} \quad (6.22)$$

El conjunto de cotas para el método CIS se obtiene mediante los siguientes valores: $\underline{\omega}_n = 1$, $\bar{\omega}_n = 4$, $\epsilon = 0.1$, $\xi = 1$ y $p = 2$. Estas constantes producen:

$$\Omega_2 = \begin{cases} 40 \leq \lambda_0 \leq 100 \\ 500 \leq \lambda_1 \leq 3200 \\ 3200 \leq \lambda_2 \leq 32000 \end{cases} \quad (6.23)$$

Basándose en las desigualdades anteriores también se proponen las cotas para generar la soluciones iniciales para el método FP, tal que se seleccionen valores aleatorios dentro de los siguientes intervalos:

$$\Omega_3 = \begin{cases} 40 \leq \lambda_0 \leq 50 \\ 500 \leq \lambda_1 \leq 1600 \\ 3200 \leq \lambda_2 \leq 16000 \end{cases} \quad (6.24)$$

Más aún con la finalidad de que exista una comparación justa entre el método CIS y el método FP, este último se restringirá a las soluciones factibles contenidas en:

$$\Omega_4 = \{x_i | \lambda_i \leq 32000, i = 1, 2, 3\} \quad (6.25)$$

para las iteraciones $k > 2$. Debe notarse que la estabilidad del OEE dentro de este conjunto no se puede asegurar por lo que ésta dependerá de la capacidad de las funciones de penalización en (6.14) para mantener las ganancias del observador tal que éste tenga una dinámica estable. Los parámetros seleccionados de dicha función de desempeño son: $\kappa = 1$ y $\eta_j = 100, j = 1, 2, 3$. Cada uno de los algoritmos se ejecuta en 30 ocasiones con $k_{max} = 100$ y $\alpha = 0.01$ correspondientes a los criterios de paro presentados en el **Algoritmo 6**.

Debe mencionarse que la primera forma de evaluar el desempeño será mediante el número de ejecuciones exitosas del algoritmo en el sentido de no presentar problemas numéricos. Esto se muestra en la Tabla 6.1.

Método de sintonización	Ejecuciones exitosas
CI	10
CIS	30
FP	30

Tabla 6.1: Número de ejecuciones exitosas del algoritmo PSO en la sintonización del OEE.

Se puede notar que la implementación del problema de optimización mediante el método CI sólo produce 10 ejecuciones exitosas, mientras que los métodos CIS y FP generan 30 ejecuciones exitosas. Esto implica que 20 ejecuciones del método CI generaron soluciones que corresponde a observadores inestables. Por lo que este método se demuestra impráctico para su implementación y no será analizado en las secciones subsecuentes.

6.3.2. Análisis estadístico

Considerando la naturaleza estadística de las señales ϕ_1 y ϕ_2 en (2.1) requiere que los resultados obtenidos sean analizados mediante técnicas estadísticas para determinar la repetibilidad de las ganancias calculadas y verificar que existan diferencias significativas entre los resultados obtenidos mediante los métodos CIS y FP. La media de las ganancias calculadas se presenta en la Tabla 6.2, donde a su vez se muestra el valor medio de la función de desempeño aplicado para cada una de ellas.

Método de sintonización	J	λ_0	λ_1	λ_2
CIS (J_1)	1.1494	42.06	1866.5	29867
FP (J_2)	1.1071	43.28	1796.3	31916

Tabla 6.2: Ganancias calculadas mediante el algoritmo PSO para el método de Cotas Inferiores y Superiores (CIS) y el método Funciones de Penalización (FP).

Debe mencionarse que las ganancias obtenidas mediante el método CIS son similares a las obtenidas con el método FP, y satisfacen el criterio de estabilidad de Routh-Hurwitz para el sistema de tercer orden (6.5). Esto incluye que sean positivas y satisfagan la condición de estabilidad $\lambda_0\lambda_1 > \lambda_2$. Mas aún, el valor de la función de desempeño J_2 que evalúa el método PF es menor que el obtenido por J_1 del método CIS.

Con la finalidad de evaluar la repetibilidad de los resultados obtenidos se presenta en la Tabla 6.3 el porcentaje de desviación estándar. Debe mencionarse que para

las ganancias λ_0 y λ_2 , ambos planteamientos presentan desviaciones similares, pero cuando se trata de la ganancia λ_1 se puede notar claramente que es el planteamiento FP que presenta una menor variabilidad en sus resultados.

Método de sintonización	J	λ_0	λ_1	λ_2
CIS (J_1)	5.37 %	3.21 %	20.55 %	0.58 %
FP (J_2)	0.90 %	3.28 %	4.57 %	0.33 %

Tabla 6.3: Porcentaje de desviación estándar de las ganancias del Observador de Estado Extendido calculadas mediante el algoritmo PSO para el método de Cotas Inferiores y Superiores (CIS) y el método Funciones de Penalización (FP).

Para verificar las diferencias entre ambos resultados y que estas no sean producidas por la naturaleza estadística de los resultados, se propone el uso de la prueba de Bonferroni usando las medias calculadas mediante Kruskal-Wallis. El p-valor obtenido con este método es de 2.353×10^{-58} y los resultados de la prueba se muestra en la Fig. 6.1. En esta última se puede notar claramente que existen diferencias altamente significativas entre los resultados obtenidos únicamente basados en la forma en que se plantea el problema de optimización a pesar de que sus ganancias se encuentran restringidas con la misma cota superior.

6.3.3. Desempeño del Observador de Estado Extendido

Para evaluar el desempeño del OEE obtenido con ambos planteamientos del problema de optimización se propone una simulación numérica del modelo (6.19) considerando $g = 9.81$ y $L = 0.7$, siendo el OEE sintonizado con las ganancias presentadas en la Tabla 6.2. La señal de entrada corresponde a (6.20), mientras que la ejecución se lleva a cabo con el método de integración Runge-Kutta con un tamaño de paso de 0.001s. La calidad de la estimación generada por el OEE es cuantificada utilizando la Integral del Error Cuadrático (IEC) definido como:

$$IEC = \int_0^t (y_1(\tau) - \hat{y}_1(\tau))^2 d\tau \quad (6.26)$$

siendo $y_1(t) - \hat{y}_1(t)$ el error de observación, estos resultados se presentan en la Tabla 6.4. De esto debe mencionarse que es el método de sintonización FP donde se produce un menor valor de ICE, implicando un mejor desempeño del observador.

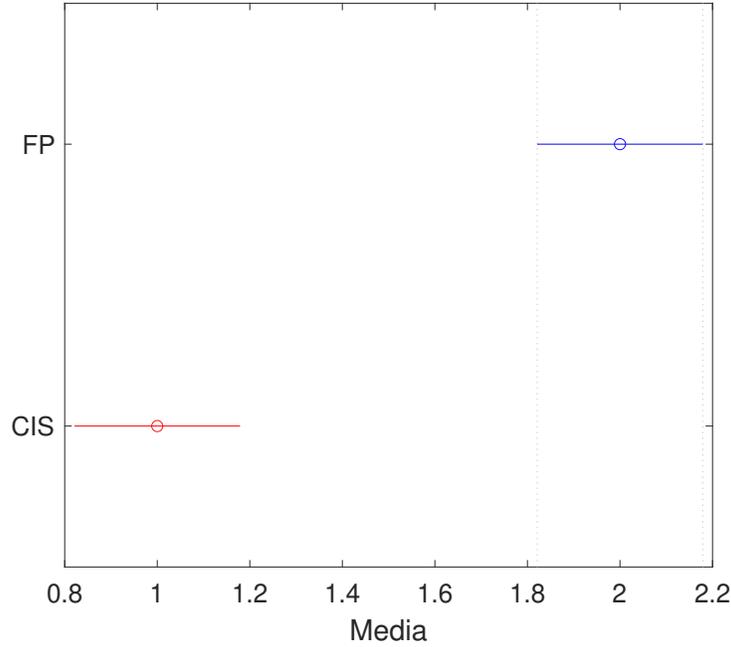


Figura 6.1: Prueba de Bonferroni para la sintonización del observador OEE mediante el planteamiento CIS y FP.

Método de sintonización	IEC
CIS	1.4858×10^{-4}
FP	1.32×10^{-4}

Tabla 6.4: Integral del Error Cuadrático IEC al implementar las ganancias calculadas mediante los métodos CIS y FP con el algoritmo PSO.

El comportamiento del seguimiento se presenta en la Fig. 6.2 el cual muestra la evolución en el tiempo de la posición angular del péndulo y_1 respecto a la estimación de la posición \hat{y}_1 . En la misma Fig. se presenta la perturbación definida como $\xi(t)$ respecto a su estimación $\hat{\xi}$.

Se puede notar que la estimación tanto de la posición angular del péndulo como de la perturbación son similares para ambos métodos de sintonización, sin embargo, la ganancia λ_3 obtenida por el método de sintonización FP se encuentra cercana a la cota propuesta en (6.25). En caso de no emplearse esta cota es esperable que el algoritmo generará ganancias del observador con valores mayores pero sin perder las condición de estabilidad de Routh-Hurwitz, situación que en la práctica podría aumentar el efecto del ruido de medición. Cabe mencionar que el caso de estudio en

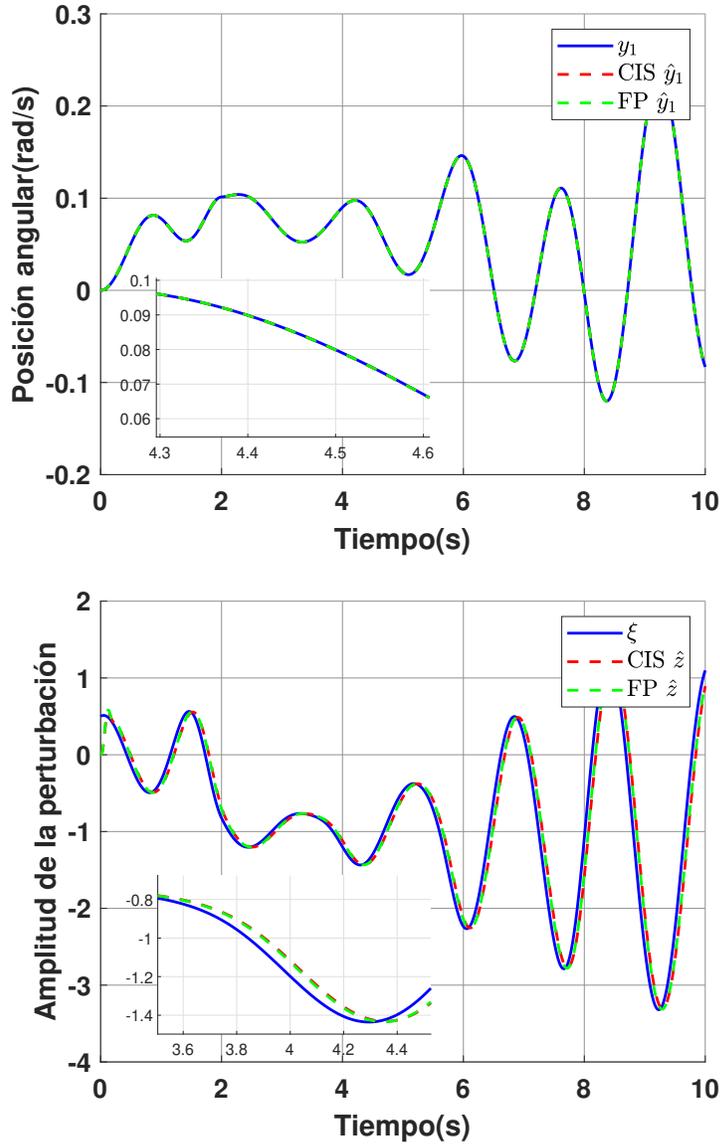


Figura 6.2: Estimación de \hat{y}_1 y \hat{z} generadas por el OEE sintonizado mediante el algoritmo PSO.

el cual se considera un observador de tercer orden permite usar (6.7) para proponer las cotas utilizadas en el método de sintonización CIS.

La selección de cotas para los métodos de sintonización CI y CIS puede ser llevada a cabo de forma arbitraria, pero se sugiere la utilización de expresiones auxiliares como en (6.7) con la finalidad de que el espacio de soluciones factibles Ω_2 tenga una mayor posibilidad de contener ganancias que generen una dinámica estable en el observador. Sin embargo, no existe certeza que todas las soluciones en Ω_2 al ser utilizadas como ganancias permitan que el observador sea estable. Más aún, estas expresiones auxilia-

res sólo se encuentran definidas para polinomios de orden bajo. La sintonización de observadores cuya dinámica es mayor a tres dificulta la implementación del método de sintonización CIS.

6.4. Conclusiones

En este capítulo se presentan tres métodos de sintonización de Observadores de Estado Extendido. El primer método de sintonización se basa en el uso de Cotas Inferiores (CI), el cual únicamente asegura que las ganancias del observador calculadas tengan valores positivos. El segundo método se basa en Cotas Inferiores y Superiores (CIS), donde se plantean un par de cotas para cada una de las ganancias del observador, éstas pueden ser seleccionadas arbitrariamente o mediante el uso de expresiones matemáticas basadas en el polinomio característico del observador pero no aseguran que todas las ganancias calculadas generen una dinámica estable en el observador. El último método propuesto en este trabajo utiliza Funciones de Penalización (FP) para asegurar que las ganancias calculadas para el observador den como resultado una dinámica estable del observador sin necesidad de definir cotas máximas o mínimas durante la mayor parte de la ejecución del algoritmo.

Basándose en los resultados obtenidos en este capítulo se puede concluir lo siguiente:

- El método de sintonización CI para obtener ganancias del Observador de Estado Extendido ha demostrado no ser una opción factible para esta tarea. Esto debido a que las ganancias calculadas pueden producir una dinámica inestable en el observador al generarse errores numéricos en la simulación dinámica, haciendo imposible continuar la ejecución del algoritmo.
- En el caso del método de sintonización CIS, a pesar de que en los experimentos realizados generó ganancias que al ser aplicadas en el Observador de Estado Extendido dieron lugar a dinámicas estables, no tiene una forma de asegurar que todas las ganancias contenidas en Ω_2 permitan cumplir con el criterio de estabilidad de Routh-Hurwitz. A pesar de que pueden existir criterios auxiliares para proponer las cotas, estos suelen ser para dinámicas donde el polinomio sea de orden bajo implicando que para sistemas de orden elevado no se tiene una guía de cómo calcularlas, haciendo que estas se propongan de manera totalmente arbitraria.
- El método de sintonización FP permite asegurar que todas las ganancias del observador calculadas den como resultado una dinámica estable mediante la

utilización de las funciones de penalización, que provocan que se incremente de forma drástica la ponderación de toda solución, que de como resultado polos con parte real positiva en el polinomio característico del observador. Este método de sintonización a su vez permite calcular soluciones con una menor variabilidad que las obtenidas con el método sintonización de CIS. Además, las ganancias calculadas mediante el método FP generan un menor error de estimación respecto al generado empleando las ganancias obtenidas mediante el método CIS.

Capítulo 7

Conclusiones

El presente trabajo ha permitido evaluar la factibilidad y los requerimientos para la implementación de algoritmos meta-heurísticos para solucionar problemas dentro del área de Control Automático. Como resultado de las problemáticas planteadas en este trabajo se han podido obtener las siguientes conclusiones:

1. Identificación de parámetros empleando el algoritmo PSO-RE.
 - Los algoritmos meta-heurísticos, en particular el algoritmo PSO es una opción factible para solucionar problemas de identificación de parámetros de sistemas dinámicos.
 - La implementación del algoritmo PSO para la identificación de parámetros puede ser realizado utilizando únicamente las señales de referencia y de salida. Esto implica que se evade la necesidad de implementar sensores adicionales u observadores que permitan obtener información acerca de los estados a los que no se puede acceder directamente.
 - Para implementar la identificación de parámetros mediante el algoritmo PSO es necesario que el sistema a identificar sea excitado adecuadamente al aplicar una señal de referencia que posea una alta riqueza espectral.
 - La medición de la riqueza espectral de una señal puede ser calculada mediante el uso de la transformada rápida de Fourier y una cota que depende de la frecuencia de muestreo y el número de muestras en la señal.
 - El algoritmo propuesto PSO-RE disminuye la variabilidad de las soluciones obtenidas cuando se utilizan señales de excitación con una baja riqueza espectral para la identificación de parámetros.
2. Sintonización de filtros empleando algoritmos de optimización Multi-Objetivo.

- El mínimo de la función de desempeño basada en el error de salida para el proceso de identificación de parámetros se ve afectado por la presencia del ruido de medición.
- Un proceso de filtrado inadecuado provoca que el mínimo de la función de desempeño empleada en la identificación de parámetros corresponda a valores de los parámetros diferentes de aquellos en los que se encuentra el mínimo de la función de desempeño cuando se considera el caso de la identificación de paramétrica de un sistema sin la presencia de ruido de medición.
- La sintonización de filtros continuos utilizados para el acondicionamiento de señales en un proceso de identificación de parámetros puede ser planteada como un problema Multi-Objetivo con la finalidad de encontrar un compromiso entre la eliminación del ruido de medición y la pérdida de correlación de la señal filtrada respecto a la señal original.

3. Sintonización de observadores empleando funciones de penalización

- En el caso de la sintonización de Observadores de Estado Extendido mediante algoritmos meta-heurísticos es necesario considerar la necesidad de que todas las soluciones empleadas en una simulación dinámica del observador produzcan un sistema exponencialmente estable.
- La utilización de funciones de penalización evita el uso de restricciones explícitas sin comprometer la estabilidad de un observador de estado extendido. Esto se logra al incluir la parte real de los polos del observador obtenidos de su polinomio característico como parámetros de las funciones de penalización. En el caso de que las ganancias calculadas tengan valores de su parte real cercanos a cero, las funciones de penalización incrementarían drásticamente el valor de la función de desempeño provocando que estas soluciones sean descartadas por el algoritmo PSO para el cálculo de las nuevas ganancias en la siguiente iteración del algoritmo.

7.1. Trabajo Futuro

Como resultado del trabajo realizado en la presente tesis se pueden abordar los siguientes temas:

- Evaluación de diferentes técnicas meta-heurísticas para la identificación de parámetros en sistemas dinámicos bajo diferentes condiciones de excitación espectral.

- Desarrollo de algoritmos especializados para la identificación de parámetros en sistemas dinámicos para casos donde se tiene una pobre riqueza espectral en las señales de excitación.
- Optimización de algoritmos meta-heurísticos para su implementación en la sintonización de observadores para aplicaciones en tiempo real.
- Desarrollo de métodos de sintonización de leyes de control mediante algoritmos meta-heurísticos considerando las condiciones de estabilidad para definir el espacio de soluciones factibles y/o la función de desempeño.

Bibliografía

- [1] Rainer Storn and Kenneth Price. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.
- [2] Marco Dorigo and Luca Maria Gambardella. Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on evolutionary computation*, 1(1):53–66, 1997.
- [3] Yuhui Shi et al. Particle swarm optimization: developments, applications and resources. In *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*, volume 1, pages 81–86. IEEE, 2001.
- [4] Dipankar Dasgupta. *Artificial immune systems and their applications*. Springer Science & Business Media, 2012.
- [5] Anne Auger and Benjamin Doerr. *Theory of randomized search heuristics: Foundations and recent developments*, volume 1. World Scientific, 2011.
- [6] Leonard David Berkovitz. *Convexity and optimization in R^n* . Wiley Online Library, 2002.
- [7] Kenneth V Price. Differential evolution. In *Handbook of optimization*, pages 187–214. Springer, 2013.
- [8] Marco Dorigo, Mauro Birattari, and Thomas Stutzle. Ant colony optimization. *IEEE computational intelligence magazine*, 1(4):28–39, 2006.
- [9] Dervis Karaboga and Bahriye Akay. A comparative study of artificial bee colony algorithm. *Applied mathematics and computation*, 214(1):108–132, 2009.
- [10] Riccardo Poli, James Kennedy, and Tim Blackwell. Particle swarm optimization. *Swarm intelligence*, 1(1):33–57, 2007.
- [11] Alejandro Rodríguez-Molina, Miguel G Villarreal-Cervantes, Efrén Mezura-Montes, and Mario Aldape-Pérez. Adaptive controller tuning method based on online multiobjective optimization: A case study of the four-bar mechanism. *IEEE transactions on cybernetics*, 51(3):1272–1285, 2019.

- [12] CA Coello Coello and Maximino Salazar Lechuga. Mopso: A proposal for multiple objective particle swarm optimization. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, volume 2, pages 1051–1056. IEEE, 2002.
- [13] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.
- [14] Rolf Isermann and Marco Münchhof. *Identification of dynamic systems: an introduction with applications*. Springer Science & Business Media, 2010.
- [15] L Ljung. *System Identification-Theory for the User 2nd edition PTR Prentice-Hall*. Prentice Hall, 1999.
- [16] Xingsheng Deng, Liangbo Yin, Sichun Peng, and Meiqing Ding. An iterative algorithm for solving ill-conditioned linear least squares problems. *Geodesy and Geodynamics*, 6(6):453–459, 2015.
- [17] Shankar Sastry and Marc Bodson. *Adaptive control: stability, convergence and robustness*. Courier Corporation, 2011.
- [18] Simon Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.
- [19] Simon Haykin. *Adaptive filter theory*. Prentice-Hall, Inc., 1996.
- [20] Ruben Garrido and Roger Miranda. Dc servomechanism parameter identification: A closed loop input error approach. *ISA transactions*, 51(1):42–49, 2012.
- [21] Mohamed Louzazni, Ahmed Khouya, Khalid Amechnoue, Alessandro Gandelli, Marco Mussetta, and Aurelian Crăciunescu. Metaheuristic algorithm for photovoltaic parameters: comparative study and prediction with a firefly algorithm. *Applied Sciences*, 8(3):339, 2018.
- [22] M Ali, MA El-Hameed, and MA Farahat. Effective parameters' identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer. *Renewable energy*, 111:455–462, 2017.
- [23] Jean-Paul Pelteret, Bastian Walter, and Paul Steinmann. Application of metaheuristic algorithms to the identification of nonlinear magneto-viscoelastic constitutive parameters. *Journal of Magnetism and Magnetic Materials*, 464:116–131, 2018.
- [24] Omar Sandre-Hernandez, Roberto Morales-Caporal, Jose Rangel-Magdaleno, Hayde Peregrina-Barreto, and Julio Noel Hernandez-Perez. Parameter identification of pmsms

- using experimental measurements and a pso algorithm. *IEEE Transactions on Instrumentation and Measurement*, 64(8):2146–2154, 2015.
- [25] V Sankardoss and P Geethanjali. Pmdc motor parameter estimation using bio-inspired optimization algorithms. *IEEE Access*, 5:11244–11254, 2017.
- [26] Zhao-Hua Liu, Hua-Liang Wei, Xiao-Hua Li, Kan Liu, and Qing-Chang Zhong. Global identification of electrical and mechanical parameters in pmsm drive based on dynamic self-learning pso. *IEEE Transactions on Power Electronics*, 33(12):10858–10871, 2018.
- [27] Clemens C Maier, Simon Schröders, Wolfgang Ebner, Marius Köster, Alexander Fidlín, and Christoph Hametner. Modeling and nonlinear parameter identification for hydraulic servo-systems with switching properties. *Mechatronics*, 61:83–95, 2019.
- [28] Zhongqiang Wu and Chunqi Du. The parameter identification of pmsm based on improved cuckoo algorithm. *Neural Processing Letters*, pages 1–15, 2019.
- [29] Dalia Yousri, Dalia Allam, and Magdy Eteiba. Parameters identification of fractional order permanent magnet synchronous motor models using chaotic meta-heuristic algorithms. In *Mathematical Techniques of Fractional Order Systems*, pages 529–558. Elsevier, 2018.
- [30] Dalia Yousri, Dalia Allam, and MB Eteiba. Chaotic whale optimizer variants for parameters estimation of the chaotic behavior in permanent magnet synchronous motor. *Applied Soft Computing*, 74:479–503, 2019.
- [31] Hassan Yousefi, Heikki Handroos, and Azita Soleymani. Application of differential evolution in system identification of a servo-hydraulic system with a flexible load. *Mechatronics*, 18(9):513–528, 2008.
- [32] Alejandro Rodríguez-Molina, Miguel G Villarreal-Cervantes, Efrén Mezura-Montes, and Mario Aldape-Pérez. Adaptive controller tuning method based on online multiobjective optimization: A case study of the four-bar mechanism. *IEEE transactions on cybernetics*, 2019.
- [33] Alejandro Rodríguez-Molina, Efrén Mezura-Montes, Miguel G Villarreal-Cervantes, and Mario Aldape-Pérez. Multi-objective meta-heuristic optimization in intelligent control: A survey on the controller tuning problem. *Applied Soft Computing*, page 106342, 2020.
- [34] Kumpati S Narendra and Anuradha M Annaswamy. Persistent excitation in adaptive systems. *International Journal of Control*, 45(1):127–160, 1987.
- [35] Er-Wei Bai and Sosale Shankara Sastry. Persistency of excitation, sufficient richness and parameter convergence in discrete time adaptive control. *Systems & control letters*, 6(3):153–163, 1985.

- [36] Ti-Chung Lee, Ying Tan, and Dragan Nešić. Stability and persistent excitation in signal sets. *IEEE Transactions on Automatic Control*, 60(5):1188–1203, 2014.
- [37] Yongduan Song, Kai Zhao, and Miroslav Krstic. Adaptive control with exponential regulation in the absence of persistent excitation. *IEEE Transactions on Automatic Control*, 62(5):2589–2596, 2016.
- [38] Erkan Kayacan, Shinkyu Park, Carlo Ratti, and Daniela Rus. Online system identification algorithm without persistent excitation for robotic systems: Application to reconfigurable autonomous vessels. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1840–1847. IEEE, 2019.
- [39] Jian Wang, Denis Efimov, Stanislav Aranovskiy, and Alexey A Bobtsov. Fixed-time estimation of parameters for non-persistent excitation. *European Journal of Control*, 55:24–32, 2020.
- [40] Girish Chowdhary, Tansel Yucelen, Maximillian Mühlegg, and Eric N Johnson. Concurrent learning adaptive control of linear systems with exponentially convergent bounds. *International Journal of Adaptive Control and Signal Processing*, 27(4):280–301, 2013.
- [41] Eli Fogel and Yih-Fang Huang. On the value of information in system identification—bounded noise case. *Automatica*, 18(2):229–238, 1982.
- [42] Chyun-Chau Fuh and Hsun-Heng Tsai. Adaptive parameter identification of servo control systems with noise and high-frequency uncertainties. *Mechanical systems and signal Processing*, 21(3):1437–1451, 2007.
- [43] Zhongbao Wei, Jiyun Zhao, Rui Xiong, Guangzhong Dong, Josep Pou, and King Jet Tseng. Online estimation of power capacity with noise effect attenuation for lithium-ion battery. *IEEE Transactions on Industrial Electronics*, 66(7):5724–5735, 2018.
- [44] Sandor Beregi, David AW Barton, Djamel Rezgoui, and Simon A Neild. Robustness of nonlinear parameter identification in the presence of process noise using control-based continuation. *Nonlinear Dynamics*, pages 1–16, 2021.
- [45] Simon Baev, I Shkolnikov, Y Shtessel, and A Poznyak. Sliding mode parameter identification of systems with measurement noise. *International Journal of Systems Science*, 38(11):871–878, 2007.
- [46] Giovanni Iacca, Ferrante Neri, and Ernesto Mininno. Noise analysis compact differential evolution. *International Journal of Systems Science*, 43(7):1248–1267, 2012.
- [47] Pratyusha Rakshit and Amit Konar. Non-dominated sorting bee colony optimization in the presence of noise. *Soft Computing*, 20(3):1139–1159, 2016.

- [48] Junqi Zhang, Xixun Zhu, Yuheng Wang, and MengChu Zhou. Dual-environmental particle swarm optimizer in noisy and noise-free environments. *IEEE transactions on cybernetics*, 49(6):2011–2021, 2018.
- [49] Wei Du, Le Tong, and Yang Tang. Metaheuristic optimization-based identification of fractional-order systems under stable distribution noises. *Physics Letters A*, 382(34):2313–2320, 2018.
- [50] DM Fébba, EC Bortoni, AF Oliveira, and RM Rubinger. The effects of noises on metaheuristic algorithms applied to the pv parameter extraction problem. *Solar Energy*, 201:420–436, 2020.
- [51] Liguó Yuan and Qigui Yang. Parameter identification of fractional-order chaotic systems without or with noise: Reply to comments. *Communications in Nonlinear Science and Numerical Simulation*, 67:506–516, 2019.
- [52] R Cortez-Vega, J Maldonado, and Rubén Garrido. Parameter identification using pso under measurement noise conditions. In *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 103–108. IEEE, 2019.
- [53] Omar Sandre-Hernandez, Roberto Morales-Caporal, Jose Rangel-Magdaleno, Hayde Peregrina-Barreto, and Julio Noel Hernandez-Perez. Parameter identification of pmsms using experimental measurements and a pso algorithm. *IEEE Transactions on Instrumentation and Measurement*, 64(8):2146–2154, 2015.
- [54] David Luenberger. Observers for multivariable systems. *IEEE Transactions on Automatic Control*, 11(2):190–197, 1966.
- [55] Jorge Davila, Leonid Fridman, and Arie Levant. Second-order sliding-mode observer for mechanical systems. *IEEE transactions on automatic control*, 50(11):1785–1789, 2005.
- [56] Kouhei Ohnishi, Masaaki Shibata, and Toshiyuki Murakami. Motion control for advanced mechatronics. *IEEE/ASME transactions on mechatronics*, 1(1):56–67, 1996.
- [57] Hebertt Sira-Ramírez, Alberto Luviano-Juárez, Mario Ramírez-Neria, and Eric William Zurita-Bustamante. *Active disturbance rejection control of dynamic systems: a flatness based approach*. Butterworth-Heinemann, 2018.
- [58] Zhonggang Yin, Chao Du, Jing Liu, Xiangdong Sun, and Yanru Zhong. Research on autodisturbance-rejection control of induction motors based on an ant colony optimization algorithm. *IEEE Transactions on Industrial Electronics*, 65(4):3077–3094, 2017.

- [59] Xingshuo Hai, Zili Wang, Qiang Feng, Yi Ren, Binghui Xu, Jingjing Cui, and Haibin Duan. Mobile robot adrc with an automatic parameter tuning mechanism via modified pigeon-inspired optimization. *IEEE/ASME Transactions on Mechatronics*, 24(6):2616–2626, 2019.
- [60] Yingjie Zhang, Chaodong Fan, Fangfang Zhao, Zhaoyang Ai, and Zhonghan Gong. Parameter tuning of adrc and its application based on cccsa. *Nonlinear dynamics*, 76(2):1185–1194, 2014.
- [61] Cheng Peng, Yantao Tian, Yue Bai, Xun Gong, Changjun Zhao, Qingjia Gao, and Dongfu Xu. Adrc trajectory tracking control based on pso algorithm for a quad-rotor. In *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, pages 800–805. IEEE, 2013.
- [62] Amjad J Humaidi, Hussein M Badr, and Akram Hashim Hameed. Pso-based active disturbance rejection control for position control of magnetic levitation system. In *2018 5th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 922–928. IEEE, 2018.
- [63] Efrén Mezura-Montes and Jorge Isacc Flores-Mendoza. Improved particle swarm optimization in constrained numerical search spaces. In *Nature-inspired algorithms for optimisation*, pages 299–332. Springer, 2009.
- [64] Mohammad Reza Bonyadi and Zbigniew Michalewicz. Particle swarm optimization for single objective continuous space problems: a review. *Evolutionary computation*, 25(1):1–54, 2017.
- [65] Daniel Bratton and James Kennedy. Defining a standard for particle swarm optimization. In *2007 IEEE swarm intelligence symposium*, pages 120–127. IEEE, 2007.
- [66] Junhong Li, Tiancheng Zong, and Guoping Lu. Parameter identification of hammerstein-wiener nonlinear systems with unknown time delay based on the linear variable weight particle swarm optimization. *ISA transactions*, 2021.
- [67] Shan-Kun Nie, Yu-Jia Wang, Shanli Xiao, and Zhifeng Liu. An adaptive chaos particle swarm optimization for tuning parameters of pid controller. *Optimal Control Applications and Methods*, 38(6):1091–1102, 2017.
- [68] Baoye Song, Zidong Wang, and Lei Zou. An improved pso algorithm for smooth path planning of mobile robots using continuous high-degree bezier curve. *Applied Soft Computing*, 100:106960, 2021.
- [69] Oliver Nelles. *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*. Springer Science & Business Media, 2013.

- [70] Petros A Ioannou and Jing Sun. *Robust adaptive control*. Courier Corporation, 2012.
- [71] Joaquín Derrac, Salvador García, Daniel Molina, and Francisco Herrera. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation*, 1(1):3–18, 2011.
- [72] Jacinto Carrasco, Salvador García, MM Rueda, Swagatam Das, and Francisco Herrera. Recent trends in the use of statistical tests for comparing swarm and evolutionary computing algorithms: Practical guidelines and a critical review. *Swarm and Evolutionary Computation*, 54:100665, 2020.
- [73] Donald W Zimmerman and Bruno D Zumbo. Relative power of the wilcoxon test, the friedman test, and repeated-measures anova on ranks. *The Journal of Experimental Education*, 62(1):75–86, 1993.
- [74] James Jaccard, Choi K Wan, and Jim Jaccard. *LISREL approaches to interaction effects in multiple regression*. Number 114. sage, 1996.
- [75] Stephen Boyd and Sosale Shankara Sastry. Necessary and sufficient conditions for parameter convergence in adaptive control. *Automatica*, 22(6):629–639, 1986.
- [76] Steven W Shaw and Christophe Pierre. Normal modes of vibration for non-linear continuous systems. *Journal of sound and vibration*, 169(3):319–347, 1994.
- [77] John G Proakis. *Digital signal processing: principles algorithms and applications*. Pearson Education India, 2008.
- [78] Ronald I Becker and Norman Morrison. The errors in fft estimation of the fourier transform. *IEEE transactions on signal processing*, 44(8):2073–2077, 1996.
- [79] Manuel López-Ibáñez, Jérémie Dubois-Lacoste, Leslie Pérez Cáceres, Mauro Birattari, and Thomas Stützle. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58, 2016.
- [80] Theodore Deliyannis, Yichuang Sun, and J Kel Fidler. *Continuous-time active filter design*. CRC press, 2019.
- [81] Ashu Soni, Nivedita Sreejeth, Varun Saxena, and Maneesha Gupta. Series optimized fractional order low pass butterworth filter. *Arabian Journal for Science and Engineering*, 45(3):1733–1747, 2020.
- [82] Shibendu Mahata, Shilpak Banerjee, Rajib Kar, and Durbadal Mandal. Revisiting the use of squared magnitude function for the optimal approximation of $(1 + \alpha)$ -order butterworth filter. *AEU-International Journal of Electronics and Communications*, 110:152826, 2019.

- [83] Shibendu Mahata, Suman Kumar Saha, Rajib Kar, and Durbadal Mandal. Optimal design of fractional order low pass butterworth filter with accurate magnitude response. *Digital Signal Processing*, 72:96–114, 2018.
- [84] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1–4. Springer, 2009.
- [85] Cyril Plapous, Claude Marro, and Pascal Scalart. Improved signal-to-noise ratio estimation for speech enhancement. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(6):2098–2108, 2006.
- [86] Carlos A Coello Coello, Gary B Lamont, David A Van Veldhuizen, et al. *Evolutionary algorithms for solving multi-objective problems*, volume 5. Springer, 2007.
- [87] Lyndon While, Philip Hingston, Luigi Barone, and Simon Huband. A faster algorithm for calculating hypervolume. *IEEE transactions on evolutionary computation*, 10(1):29–38, 2006.
- [88] Michael Zeitz. The extended luenberger observer for nonlinear systems. *Systems & Control Letters*, 9(2):149–156, 1987.
- [89] Hassan K Khalil. Nonlinear systems third edition. *Patience Hall*, 2002.
- [90] Rafal Madoński and Przemyslaw Herman. Survey on methods of increasing the efficiency of extended state disturbance observers. *ISA transactions*, 56:18–27, 2015.
- [91] Edmund X DeJesus and Charles Kaufman. Routh-hurwitz criterion in the examination of eigenvalues of a system of nonlinear ordinary differential equations. *Physical Review A*, 35(12):5288, 1987.
- [92] Bartłomiej Wicher and Stefan Brock. Tuning optimization of extended state observer for two mass system with elastic joint and backlash. In *2018 18th International Conference on Mechatronics-Mechatronika (ME)*, pages 1–6. IEEE, 2018.