

# Reconstrucción de señales de falla en manipuladores de robots de servicio

Mitzi Anaid Ramírez Estrada

2020-2021



# Contents

<b>1</b>	<b>Introducción</b>	<b>5</b>
1.1	Motivación . . . . .	5
1.2	Planteamiento del problema . . . . .	5
1.3	Hipótesis . . . . .	5
1.4	Objetivos . . . . .	5
1.5	Descripción del documento . . . . .	6
<b>2</b>	<b>Antecedentes</b>	<b>7</b>
2.1	Robots de servicio doméstico . . . . .	7
2.2	Modelado y control de robots manipuladores . . . . .	9
2.2.1	Manipulator . . . . .	9
2.2.2	Modelo Dinámico . . . . .	10
2.3	Reconstrucción de fallas . . . . .	10
2.4	La plataforma ROS . . . . .	10
<b>3</b>	<b>Reconstrucción de fallas con un Observador por Modos Deslizantes</b>	<b>11</b>
3.1	Modelo del manipulador . . . . .	12
3.2	Diseño del observador . . . . .	12
3.3	Reconstrucción de la señal de falla . . . . .	12
3.4	Implementación en Simulink . . . . .	12
<b>4</b>	<b>Estimación de Parámetros</b>	<b>13</b>
4.1	Estimación de parámetros por mínimos cuadrados no lineales . . . . .	13
4.2	Mínimos cuadrados no lineales . . . . .	14
4.2.1	Algoritmo de Levenberg–Marquardt . . . . .	14
<b>5</b>	<b>Redes Neuronales</b>	<b>15</b>
5.1	Reconstrucción de fallas usando Redes Neuronales . . . . .	15
<b>6</b>	<b>Resultados</b>	<b>17</b>
6.1	Reconstrucción de la señal de falla . . . . .	17
6.2	Desempeño del SMO . . . . .	17
6.3	Desempeño del EKF . . . . .	17
6.4	Estimación de masa con ambos métodos . . . . .	17

6.5	Comparación de la efectividad entre métodos . . . . .	17
<b>7</b>	<b>Discusión</b>	<b>19</b>
7.1	Conclusiones . . . . .	19
7.2	Trabajo Futuro . . . . .	19

# Chapter 1

## Introducción

### 1.1 Motivación

### 1.2 Planteamiento del problema

En el desempeño de los robots de servicio, el cálculo de la masa del objeto que busca ser transportado por el robot representa una mejora en la eficiencia del cumplimiento de una tarea. Puede realizarse una mejor selección de objetos, filtrando aquellos que no cumplan con la masa estimada del objeto que se busca. Por ejemplo, en caso de que se requiera que el robot mueva un objeto que se encuentra entre varios elementos con la misma apariencia pero masas distintas, el éxito de la tarea podría lograrse en mayor tiempo, o incluso verse comprometido, si ésta es desempeñada por un robot que utilice sólo procesamiento de imágenes y no cuente con la posición exacta del objeto en específico. De esta forma, al contar con la estimación de masa, se podría seleccionar de mejor forma y en menor tiempo el elemento indicado y, acompañado de los algoritmos de navegación y visión, llevar el objeto seleccionado a la posición solicitada.

### 1.3 Hipótesis

### 1.4 Objetivos

Implementar un algoritmo de control mediante el cual sea posible estimar el comportamiento interno de un sistema si se conocen las entradas y las salidas obtenidas. Así, con la reconstrucción de dicho comportamiento, llevado a cabo cuando el robot se encuentra recibiendo una perturbación externa, se puede comparar con los datos nominales del sistema, en este caso los torques del brazo del robot sin carga, y determinar la diferencia entre el comportamiento nominal y el obtenido cuando el robot sostiene un objeto. Lo anterior descrito permite utilizar la señal de falla para estimar la masa del objeto y, a partir de ello, hacer

uso de un controlador que permita llegar a la posición esperada del efector final a pesar de la masa añadida, en este caso, el control implementado realizaría un cálculo de la corriente necesaria para aumentar el torque de los servomotores que conforman el brazo lograr el movimiento hasta la posición pedida.

## 1.5 Descripción del documento

## Chapter 2

# Antecedentes

### 2.1 Robots de servicio doméstico

Los robots de servicio doméstico se caracterizan por llevar a cabo tareas repetitivas, sucias, peligrosas o bien que consumen mucho tiempo, siendo su principal función evitar o disminuir la intervención del humano en dichas actividades. Deben ser autónomos en cierta medida, es decir, que puedan realizar las tareas mencionadas sin necesitar la ayuda de un operador, para esto deben contar con un sistema de sensores y motores que les permitan interactuar y moverse en el ambiente en que se encuentra.

Para el caso de los robots que participan en competencias como RoboCup@Home, es necesario que el robot cuente con las siguientes *Propiedades el Sistema* [9]:

- Facilidad de uso.
- Rápida calibración y preparación.
- Interacción natural y multi-modal.
- Apariencia y ergonomía.
- Adaptatividad/Inteligencia General.
- Robustez.
- Aplicabilidad general.

Este tipo de robots tienen la intención de apoyar principalmente a personas de edad avanzada y/o con algún tipo de discapacidad, reemplazando la intervención humana en tareas que así lo permitan.

Para realizar las actividades solicitadas se requieren *Habilidades funcionales*, listadas a continuación:

- Navegación: planeación de los movimientos necesarios para llegar a una posición conveniente evitando colisiones.

- Mapeo: conocimiento del espacio en que se encuentra.
- Reconocimiento de personas.
- Rastreo de personas.
- Reconocimiento de objetos
- Manipulación de objetos.
- Reconocimiento de voz.
- Reconocimiento de gestos humanos.

En las competencias, cada equipo decide cuál de estas habilidades implementar o a cuál se le dará prioridad, dependiendo de las pruebas en las que se busque participar. [9] De esta manera, es clara la importancia de la manipulación de objetos como un elemento básico en el comportamiento de un robot de servicio, ya que no contar con esta característica implicaría no ser capaz de interactuar con su ambiente de forma útil.

Una de las limitaciones de los robots de servicio se encuentra en la probabilidad de un cambio en las características de los objetos que deben ser manipulados. Por ejemplo, si durante una actividad es necesario que el robot realice movimientos con contenedores de líquidos, como puede ser el servir una bebida, en determinado momento del desarrollo de la tarea, los contenedores cambian una de sus características, dado que el volumen del líquido que se encontraba en uno de ellos ahora se ha transferido al otro, modificando el peso de ambos contenedores y limitando los movimientos que se pueden realizar con aquel que tiene ahora el líquido sin derramar su contenido. Un ejemplo distinto incluye la introducción de un nuevo objeto dentro del ambiente del robot, dando paso a la necesidad de un proceso de aprendizaje sobre sus características y las opciones que el robot tiene sobre la manipulación del objeto. Ésta última situación presenta un problema más complicado ya que para realizar el proceso de aprendizaje se requiere la intervención de un operador que modifique el catálogo de objetos que el robot conoce y el código que permitirá al robot seleccionar e interactuar con el objeto de la forma más óptima posible. Dentro de las mejoras que se realizan continuamente en el desarrollo de algoritmos de comportamiento de los robots de servicio, la manipulación de objetos sigue siendo un campo abierto a nuevas propuestas de solución para los desafíos que se encontrarán los robots al desempeñar sus labores de apoyo doméstico. [3].

Es muy común que los robots de servicio se apoyen primordialmente en las cámaras y sensores de proximidad con las que cuenta el robot, a través de los cuales reconoce su entorno, identifica los objetos que lo rodean y la distancia a la que se encuentran de él, no obstante, esta combinación de mediciones encuentra una limitación cuando el problema a resolver no es acerca de la apariencia o la distancia a un objeto, sino con otras de sus características. Anteriormente se mencionó el ejemplo de servir bebidas, utilizando recipientes como latas o envases de cartón. En este escenario, frecuentemente existirá la posibilidad de



encontrar dos o más recipientes cuya apariencia sea igual y que no sea viable diferenciar cuál de las múltiples posibilidades se encuentra lleno o vacío haciendo uso únicamente de análisis de imágenes. En casos como este es necesario incluir una opción más para que el robot utilice los datos que recibe continuamente de sus múltiples sensores y, de esta forma, obtener alternativas para que el robot interactúe de forma más eficiente con su entorno. Así, con la posibilidad de aprovechar información por medio de las mediciones que el robot recibe de sus elementos, como son los motores que permiten el movimiento de sus manipuladores, y las condiciones bajo las que están operando, así como los cambios que presentan durante la manipulación de un objeto. La reconstrucción de señales de falla permite utilizar estos datos mediante la diferencia entre el comportamiento nominal de los actuadores del robot y su comportamiento cuando están siendo utilizados para realizar una tarea. Observando el comportamiento de los motores, identificando cuándo se produce una alteración y, posteriormente, el tipo de alteración de la que se trata. Con esta nueva información es posible conocer características de los objetos que se manipulan, diferentes a la apariencia de los objetos, y con esto permitir una mejor, más rápida y más eficiente manera de realizar una tarea. [5]

## 2.2 Modelado y control de robots manipuladores

### 2.2.1 Manipulator

El manipulador que será utilizado pertenece al Robot "Justina", un robot de servicio construido en el laboratorio de Biorrobótica de la Universidad Nacional Autónoma de México, este robot ha participado en competencias tales como Robocup@Home league en varias ocasiones, desempeñando tareas como limpieza de una mesa, preparación de bebidas y actividades similares comúnmente requeridas por los humanos en un entorno doméstico. Justina cuenta con dos manipuladores, cada uno de siete grados de libertad conformados por servomotores Dynamixel. Estos motores permiten al usuario controlar la posición en que se encuentren, la velocidad a la que se moverán o modificar la corriente que recibirán los motores y con ello, el torque. Además, los motores Dynamixel cuentan con sensores de posición con una resolución de menos de un décimo de grado.

Para realizar la simulación del manipulador se utiliza un archivo URDF (Universal Robot Description Format), el cual permite trabajar con un modelo del manipulador. Este archivo detalla las características del manipulador, las articulaciones y eslabones que lo conforman, el material y masa de sus componentes, las matrices de inercias asociadas a sus elementos y la posición en que se encontrará en la simulación. Para este documento se seleccionó el modelo del brazo izquierdo del robot, que fue aislado del archivo URDF del Robot Justina.

### 2.2.2 Modelo Dinámico

El modelo se obtiene del Lagrangiano del manipulador, obteniendo un modelo dinámico que se conforma por las posiciones angulares de los motores  $q \in \mathbb{R}^7$ , la matriz de inercias  $M(q) \in \mathbb{R}^{7 \times 7}$ , el vector de fuerzas centrífugas y de Coriolis  $C(q, \dot{q}) \in \mathbb{R}^{7 \times 7}$ , la matriz de coeficientes de fricción  $B \in \mathbb{R}^7$ , el vector  $G(q) \in \mathbb{R}^7$ , que contiene los pares debidos a la gravedad, el vector  $u$ , formado por los pares de entrada (señales de control) y el vector  $\Delta(q, \dot{q}, u)$  que contiene los errores debidos a incertidumbres, perturbaciones o señales de falla. [5]

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + B\dot{q} + G(q) + \Delta(q, \dot{q}, u) = u \quad (2.1)$$

## 2.3 Reconstrucción de fallas

Para realizar la reconstrucción de las señales de falla de un sistema, la detección e identificación de dichas fallas juega un rol vital proporcionando información acerca de las fallas o los errores del sistema que hacen posible que haya una apropiada reconfiguración en base a ellas. De esta forma, los sistemas de Detección y aislamiento se encargan de localizar la falla, de tal forma que la acción correctiva puede realizarse para eliminar o minimizar el efecto de la falla en el desempeño del sistema completo. El comité técnico IFAC, hace las siguientes definiciones,

- Detección de fallas: determinación de las fallas presentes en un sistema al momento de la detección.
- Aislamiento de fallas: Determinación del tipo, localización y tiempo de detección de la falla.
- Identificación de fallas: Determinación del tamaño y comportamiento variante en el tiempo de la falla.

## 2.4 La plataforma ROS

## Chapter 3

# Reconstrucción de fallas con un Observador por Modos Deslizantes

En la teoría de control moderno es frecuente encontrar diferencias entre el comportamiento real del sistema y la aproximación que se hace de él mediante un modelo matemático, para este tipo de situaciones se busca incluir técnicas e control robusto, reduciendo los errores derivados de las incertidumbres, parámetros desconocidos o perturbaciones del sistema. Usando este tipo de técnicas es posible generar un algoritmo de control de lazo cerrado con un desempeño adecuado para el objetivo que se esté persiguiendo. Dentro de las técnicas de control robusto se encuentran los modos deslizantes, los cuales podrían mencionarse como aquellos con mayor éxito en el manejo de incertidumbres o perturbaciones delimitadas. [7]

Para sistemas que requieren cierto nivel de desempeño y estabilidad en escenarios inesperados se suele necesitar una estrategia distinta, más allá de tener un control adaptativo o robusto, el enfoque del control tolerante a fallas ofrece una alternativa para adaptar las fallas en los componentes del sistema de forma automática. Así, se obtiene estabilidad del sistema y un desempeño aceptable dado el caso de dichas fallas. Dentro de los esquemas de control con tolerancia a fallas, se encuentran los pasivos y los activos, donde los pasivos operan de forma independiente a la información obtenida de las fallas y más bien explotan la robustez del controlador. Por otro lado, los esquemas activos reaccionan a las fallas, generalmente utilizando la información de un sistema de detección, aislamiento e identificación de las fallas e implementando cierta forma de reconfiguración. Es en este punto en que se unen las características de los esquemas de control tolerantes a fallas con los atributos de los controladores por modos deslizantes, dado que la robustez, adaptatividad y sus propiedades de resistencia a fallas, facilitan considerar a los SMO como una alternativa para control reconfigurable. [1]

### **3.1 Modelo del manipulador**

### **3.2 Diseño del observador**

### **3.3 Reconstrucción de la señal de falla**

### **3.4 Implementación en Simulink**

## Chapter 4

# Estimación de Parámetros

Este método consiste en utilizar las mediciones que se hagan de la salida del sistema y el conocimiento que se tenga de las entradas del mismo para obtener los parámetros que mejor se ajusten a las medidas realizadas. De esta forma y mediante iteraciones, se obtiene progresivamente la minimización de la diferencia entre el comportamiento del sistema con sus parámetros reales y aquella que se obtiene al utilizar los parámetros estimados. Al ser el modelo del manipulador un sistema no lineal, se utiliza un algoritmo que cuente con propiedades de convergencia en un tiempo razonable.

### 4.1 Estimación de parámetros por mínimos cuadrados no lineales

Un método comúnmente usado para la solución de un sistema de ecuaciones no lineales es el método de los mínimos cuadrados, que puede basarse en diferentes algoritmos, Newton-Gauss, Levenberg-Marquardt o Región de Confianza. Para utilizar este procedimiento es conveniente que la función a tratar sea continua, doblemente diferenciable y estrictamente convexa, aunque estas características dependen del algoritmo con que se trabaje. [2] Matemáticamente, el problema de los mínimo cuadrados se describe de la siguiente forma:

$x \in \mathbb{R}^n$

$$S = \min f(x) = \frac{1}{2} R(x)^T R(x) = \frac{1}{2} \sum_{i=1}^m r_i(x)^2 \quad (4.1)$$

Donde  $R(x)$  es la función residual  $r_i(x)$  es la componente  $i$ -ésima de la misma, definido como:

$$r_i(x) = y_i - f(x_i, \theta) \quad (4.2)$$

$m$  es el número de observaciones y  $n$  es el número de parámetros. [4]

## **4.2 Mínimos cuadrados no lineales**

### **4.2.1 Algoritmo de Levenberg–Marquardt**

## Chapter 5

# Redes Neuronales

Las Redes Neuronales Artificiales ofrecen un acercamiento robusto a la aproximación de funciones objetivo de valor real, discreto o vectorial. Para ciertos tipos de problemas, tales como *aprender* a interpretar datos de sensores complejos del mundo real, las redes neuronales se encuentran entre los métodos de aprendizaje más efectivos actualmente conocidos.[8]

Una red Neuronal estándar se conforma por procesadores simples que se encuentran interconectados, conocidos como neuronas. Cada una de estas neuronas produce una secuencia de activaciones de valor real. Las neuronas de entrada se activan a través de sensores que perciben el entorno; otras neuronas se activan por medio de conexiones ponderadas de neuronas previamente activadas. Algunas neuronas pueden influir en el entorno al desencadenar acciones. El *aprendizaje* de la red Neuronal consiste en encontrar ponderaciones que hacen que la red neuronal muestre un comportamiento deseado, como puede ser manejar un auto de forma autónoma. [6]

### 5.1 Reconstrucción de fallas usando Redes Neuronales





## Chapter 6

# Resultados

- 6.1 Reconstrucción de la señal de falla
- 6.2 Desempeño del SMO
- 6.3 Desempeño del EKF
- 6.4 Estimación de masa con ambos métodos
- 6.5 Comparación de la efectividad entre métodos



## Chapter 7

# Discusión

### 7.1 Conclusiones

### 7.2 Trabajo Futuro



# Bibliography

- [1] Halim Alwi, Christopher Edwards, and Chee Pin Tan. *Fault detection and fault-tolerant control using sliding modes*. Springer Science & Business Media, 2011.
- [2] Óscar Cornejo Zúñiga and Rodrigo Rebolledo Vega. *Estimación de Parámetros en Modelos No Lineales*. Universidad EIA, 2016.
- [3] S. Hamidreza Kasaei et al. *The state o lifelong learning in Service Robots*. 2020.
- [4] C. T. Kelley. *Iterative methods for Optimization*. Society for Industrial and Applied Mathematics Philadelphia, 1999.
- [5] Marco Negrete, Jesús Savage, and José Avendaño. *Mass Estimation in Manipulation Tasks of Domestic Service Robots using Fault Reconstruction Techniques*. 2020.
- [6] Jürgen Schmidhuber. “Deep learning in neural networks: An overview”. In: *Neural Networks* 61 (2015), pp. 85–117. ISSN: 0893-6080. DOI: <https://doi.org/10.1016/j.neunet.2014.09.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0893608014002135>.
- [7] Yuri Shtessel et al. *Sliding mode control and observation*. Springer, 2014.
- [8] M. Mitchell Tom. *Machine Learning*. McGraw-Hill Science/Engineering/Math, 1997.
- [9] Thomas Wisspeintner et al. *RoboCup@Home: Results in Benchmarking Domestic Service Robots*. 2018.