

Desarrollos con Enfoque Mecatrónico



Juan Manuel Ramos Arreguín
José Emilio Vargas Soto
Jorge Enrique Orozco Ramírez

Desarrollos con Enfoque Mecatrónico

1ª Edición

**Ramos Arreguín J.M.
Vargas Soto J. E.
y Orozco Ramírez J. E.**

“Desarrollos con Enfoque Mecatrónico”

Ramos Arreguín Juan Manuel
Vargas Soto José Emilio
Orozco Ramírez Jorge Enrique

© Asociación Mexicana de Mecatrónica A.C.
Prolongación Corregidora Norte No. 168, interior E
Col. El Cerrito, C.P. 76160, Querétaro, Qro.
México.

Internet: <http://www.mecamex.net>

1ª Edición, 02 de Octubre, 2020.

ISBN: 978-607-9394-22-6

Derechos reservados.

Esta obra es propiedad intelectual de sus autores y los derechos de publicación han sido legalmente transferidos a la editorial. Las opiniones y la información que se muestran en los capítulos del libro son exclusivas de los autores de cada capítulo y no representan la postura de la Asociación Mexicana Mecatrónica A.C. Prohibida su reproducción total o parcial por cualquier medio sin la autorización por escrito del propietario de los derechos del copyright.

Impreso en México – Printed in Mexico



Índice

Capítulo 1	1
Educación de la conciencia de la sociedad en el comportamiento de las estructuras de 1 GDL Candia García Filiberto, Castillo Flores Martín, Méndez Flores Javier, Vivaldo de la Cruz Israel y García Sánchez Enrique.	
Capítulo 2	15
API testing usando REST Assured, Cucumber y Java Sánchez Mares Gilberto.	
Capítulo 3	27
Seguimiento de trayectorias de un robot móvil diferencial a través del sistema operativo robótico ROS Salazar-Hidalgo Eduardo, Castañeda-Camacho Josefina, Martínez-Torres Cesar y Martínez-Carranza José.	
Capítulo 4	39
Sistema de capacitación con realidad aumentada para el manejo de un torno paralelo modelo G079 Cárdenas López Miguel Martín, Paredes Orta Carlos Alberto, Valentín Coronado Luis Manuel y Acevedo Ramírez Gustavo Adolfo.	
Capítulo 5	53
Monitoreo a distancia de intensidad de luz basado en fotodiodo para la adquisición de datos en tiempo real Blas Hernández Marycruz y García Valdez Chayanne	
Capítulo 6	69
Diseño de una turbina eólica helicoidal usando técnicas computacionales Tello Campos Alejandro Roberto	



Índice

- | | |
|---|------------|
| Capítulo 7 | 79 |
| Diseño y construcción de un robot cuadrúpedo para la difusión de las Ingenierías en Electrónica y Mecatrónica
Santiago Espinosa Felipe y Barahona Ávalos Jorge Luis | |
| Capítulo 8 | 95 |
| <i>Verticallium</i> : Solución Integral para el Cultivo Vertical de Ajo Usando Compost Aplicando IoT
Castillo Javier, Grisales Andres Felipe, Martinez Ortega Sara Virginia, Ortega Solarte Leidy Dayanna y Ramirez Preciado Leider. | |
| Capítulo 9 | 113 |
| Diseño e implementación de reguladores saturados con ganancias variables para robots manipuladores
Sánchez García Brandon, Rodríguez Hernández Ernesto, Reyes Cortés Fernando, Al-Hadithi B. M. y Félix Beltrán Olga. | |
| Capítulo 10 | 131 |
| Errores en la relación de transmisión de los engranajes cilíndricos de dientes rectos
Reyes Méndez Alfonso Daniel, Félix Beltrán Olga Guadalupe y Mendoza Quiñonez Jesús Alberto. | |
| Capítulo 11 | 145 |
| Un primer acercamiento al diseño de sombreros asistido por una red adversaria-generativa
Salazar-Colores Sebastián, Carrillo-López David, Gerardo Flores, Moya-Sánchez E. Ulises y Ayar-Rojas Nadia. | |
| Capítulo 12 | 155 |
| Modelado y Control Lineal de Sistemas Físicos Mediante Prototipos: Una Estrategia de Aprendizaje
Velázquez Izguerra Joel, Pérez Silva Ricardo Bladimir, Soria Medina Abraham Estefano y Peñaloza Mendoza Guillermo Rey | |



Índice

- | | |
|--|------------|
| Capítulo 13 | 167 |
| Nuevo modelo de fricción para robots manipuladores basado en funciones hiperbólicas | |
| Rodríguez Hernández Ernesto Daniel, Sánchez García Brandon Erick, Reyes Cortés Fernando, Oviedo Barriga José Luis y Vargas Treviño María Aurora. | |
| Capítulo 14 | 179 |
| Dispositivo de adquisición y procesamiento de Señales Electromiografías | |
| Córdova Morales Guillermo, Montiel Jiménez, Elmar, Lobato Báez Mariana, Morales Suarez Humberto y Benito Carrillo Ismael. | |
| Capítulo 15 | 189 |
| Plano automatizado de reaprendizaje neuromotor para pacientes con Enfermedad Vascul ar Cerebral | |
| Ángeles Martínez Juan Pablo y Pérez Prado Edgar Eduardo. | |



Educar la conciencia de la sociedad en el comportamiento de las estructuras de 1 GDL

Candia García Filiberto^{✉1}, Castillo Flores Martín¹, Méndez Flores Javier², Vivaldo de la Cruz Israel², García Sánchez Enrique R².

¹Facultad de Ingeniería, Benemérita Universidad Autónoma de Puebla, Puebla México.

² Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla, Puebla México.

✉ filiberto.candia@correo.buap.mx

Resumen

La incertidumbre en cuanto a la seguridad personal y patrimonial del pasado sismo del 19 de septiembre de 2017, lleva a la realización del presente estudio experimental por simulación virtual del comportamiento de las estructuras de 1 GDL ante la presencia de vibraciones aleatorias y modificación de la masa no estructural. El propósito es proponer un procedimiento virtual que permita en las Instituciones Educativas (superior y posgrado), fomentar el criterio de certidumbre ante riesgos sísmicos y la prevención de daños estructurales, que pongan en peligro la vida humana, su salud emocional y la estabilidad de la infraestructura. La metodología es teórica-experimental, con base en el método científico-deductivo, los resultados se obtienen por simulación mediante software especializado basado en el Método de los Elementos Finitos. Se ha concluido que enseñar el comportamiento de estructuras de 1 GDL ante vibraciones aleatorias y la modificación de la masa no estructural, mediante un prototipo por simulación virtual, proporciona certidumbre con respecto al comportamiento de estructuras de 1 GDL.

Palabras clave: Estructuras 1 GDL, Respuesta dinámica, Vibraciones aleatorias.

1. Introducción

Este trabajo de orientación didáctica tiene relevancia tecnológica-social, ya que el objetivo principal es evaluar un prototipo virtual que permita en las Instituciones Educativas (superior y posgrado), fomenten el criterio de certidumbre ante riesgos sísmicos y la prevención de daños estructurales cuando se incrementa la masa no estructural en estructuras de 1 grado de libertad (GDL), que pongan en riesgo la vida humana, su salud emocional y la estabilidad de la infraestructura. Dentro de los objetivos específicos está el mostrar la correlación de los esfuerzos cortantes de tornillos contra la rigidez de las uniones, cuando la estructura no es sólida, pero que están presentes cuando la estructura contiene conexiones tornillo. Debido a que la idea coloquial/común de la población es que una estructura modificada por incremento de masa no estructural, mientras más rígida es más segura. Siendo relevante ser claros en estos supuestos, mediante la observación en prototipos virtuales con escenarios controlados del comportamiento de estructuras de 1 GDL ante vibraciones aleatorias.

El estado del arte aporta información sobre que las conexiones tornillo son un tema de mucho interés para la comunidad que se dedica al diseño estructural. Debido a que en muchas ocasiones este tipo de conexiones no flexibles en el plano transversal, su análisis es omitido o sustituido por una unión rígida. Por ello el empleo del Método de los Elementos Finitos (MEF) con simulación tridimensional (3D) es una alternativa que permite su estudio a profundidad y muestra los efectos de un aflojamiento o extremo apriete en estructuras metálicas de 1 GDL. La revisión analítica de los procedimientos (estático y dinámico) para determinar la respuesta dinámica de estructuras de acero de 1 GDL, es el fundamento



teórico que apoya la descripción del comportamiento ante la variación del diseño inicial y posteriormente las inadecuadas prácticas de fabricación que modifican la función y finalidad de la estructura.

La metodología establece una secuencia sistemática que permite la repetitividad de los resultados con alta precisión y exactitud. La ventaja de utilizar la simulación por MEF, es que permite establecer una comparativa que define una correlación entre el apriete de los tornillos y la generación de altos esfuerzos cortantes en las uniones en conexiones tornillo. El desarrollo involucra la modelación virtual y el análisis por medio del MEF, estas herramientas se configuran en escenarios diversos que buscan identificar las condiciones más desfavorables que causarían el mayor riesgo ante la eventualidad de una vibración aleatoria. Para ello en la tabla 1, se muestran los valores cambiantes de las aceleraciones en el eje transversal “x”, como los efectos de las condiciones de carga más desfavorables.

Tabla 1. Valores de aceleración detectados por una interface de monitoreo, que varían de acuerdo con la posición que tenga la carga en la estructura, ante un torque constante en las conexiones tornillo.

Localización de la carga en la estructura	Vista en Planta	Torque conexiones de la estructura	Valor promedio amplitud eje x (G)	Valor promedio amplitud eje y (G)	Valor promedio amplitud eje z (G)	Valor máximo de amplitud eje x		Valor máximo de amplitud eje y		Valor máximo de amplitud eje z	
						G	Hz	G	Hz	G	Hz
Frontal-Izquierda		23.03 kg-cm	0.005578	0.003749	0.001974	0.291	40.7	0.1446	20.07	0.04496	20.07
Frontal-derecha		23.03 kg-cm	0.002769	0.001676	0.001154	0.2258	20.03	0.1707	40.69	0.02582	40.09
Frontal-centro		23.03 kg-cm	0.004675	0.002549	0.00235	0.4907	20.03	0.09626	20.03	0.05102	20.03
Centro		23.03 kg-cm	0.00412	0.00335	0.001497	0.3116	19.99	0.1647	20.73	0.06316	19.99
Posterior-derecha		23.03 kg-cm	0.004933	0.005282	0.006117	0.2754	19.93	0.3372	19.93	0.6496	19.96
Posterior-izquierda		23.03 kg-cm	0.002822	0.002487	0.002377	0.1255	40.72	0.1248	40.72	0.02552	20.66
Posterior-centro		23.03 kg-cm	0.004508	0.006862	0.008587	0.256	40.69	0.1546	19.89	0.08914	2.233

Los resultados indican que el escenario donde existe aprietes extremos en las conexiones tornillo se tienen las condiciones más desfavorables y se contrastan contra los indicadores de seguridad como lo son el esfuerzo máximo a tensión y la tensión de rotura. En la discusión, se ha verificado que ha mayor apriete mayor esfuerzo cortante en las uniones, lo cual no representa altas deformaciones, aspecto que puede llevar a conducir a fallas por cortante del tornillo o de fractura de los elementos estructurales. Se concluye que el dispositivo/prototipo virtual es funcional, para crear conciencia en la población desde diversos niveles educativos, ya que es de fácil manejo y de manera eficiente y simple ilustra las consecuencias de alterar el diseño inicial de las estructuras de 1 GDL, ya sea por adición de masa o la inadecuada la rigidez en las uniones.

1.1 Estado del arte

Se considera en este trabajo que es relevante detallar la revisión de las conexiones por sujeción como los tornillos y pernos en estructuras metálicas de 1 GDL, debido a que: La frecuencia de operación de los sistemas dinámicos como lo son: las máquinas y estructuras mecánicas dedicadas a la producción de bienes y productos, generalmente dependen de la frecuencia de las fuerzas que provocan vibración no deseada. La identificación de los parámetros modales de dichos sistemas son útiles para construir y validar modelos que permitan entender y correlacionar la respuesta dinámica a diferentes

frecuencias de excitación [1]. Por ello se ha considerado que la evaluación de la respuesta dinámica es una alternativa viable, para determinar fallas estructurales, debido a que, en la simulación modal estructural, los parámetros modales son directamente relacionados con las mediciones de esfuerzos obtenidas por simulación virtual. Asimismo, por medio de la emulación virtual de condiciones catastróficas es posible la identificación de daños estructurales, la distribución espacial de daños estructurales y las mejores distribuciones de carga [2].

Con respecto a los resultados obtenidos por la Universidad de Alberta (2017) que realizó pruebas y análisis de elemento finito en placas de refuerzo atornilladas sometidas a cargas monotónicas y cíclicas. Se argumentan los resultados bajo este criterio, debido a que estos estudios resaltaron que una parte considerable de la energía es absorbida por el deslizamiento, y la cizalla de bloque de la placa de refuerzo. Estos estudios numéricos muestran que la falla de corte de bloque en la abrazadera o refuerzo conduce a buenos modos de falla dúctil. También se considera que se debe evitar el pandeo de la articulación, particularmente en ausencia de excentricidad y que el deslizamiento del perno aumenta claramente la disipación de energía en las conexiones [3]. Se coincide con [4] que el diseño de conexiones para marcos en estructuras de acero es un tema que requiere de un alto grado de conocimientos por parte del diseñador. Asimismo, que el modelado de una conexión de perno pretensado utilizando sólidos de contacto, superficies y vigas es una combinación de solución práctica y visualización real [5].

2. Metodología

La metodología es teórica-experimental, con base en el método científico-deductivo. En una propuesta correlacional los datos se comparan en el marco del comportamiento de tres escenarios: baja frecuencia de vibración, vibración en resonancia y alta frecuencia de vibración, con respecto a la frecuencia natural de la estructura. Para la repetitividad de los resultados en cada simulación se estableció el procedimiento de la figura 1b.

Pasos del procedimiento:

1. Configurar en la estructura de 1 GDL el escenario a evaluar
2. Por medio del modelo de elemento finito se adecuan las condiciones de contorno
3. Realizar el montaje y conexión del sensor en la estructura
4. Establecer las condiciones de simulación en el software
5. Arrancar la interface de monitoreo, figura 1a.
6. Correr la simulación virtual
7. Recoger datos en tablas comparativas
8. Evaluar la aproximación de los resultados
9. Describir y correlacionar el comportamiento de la estructura
10. Repetir el proceso para los diversos escenarios posibles

La interfaz de monitoreo se compone de los elementos mostrados en la figura 1. La cual permite tener control en caso de detectarse una condición insegura de trabajo, así como medir aceleraciones, para monitorear el estado estable de la estructura.

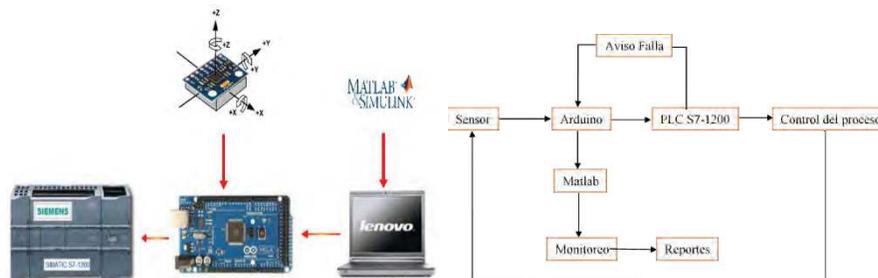


Figura 1a. interfaz de monitoreo para el control de vibraciones.

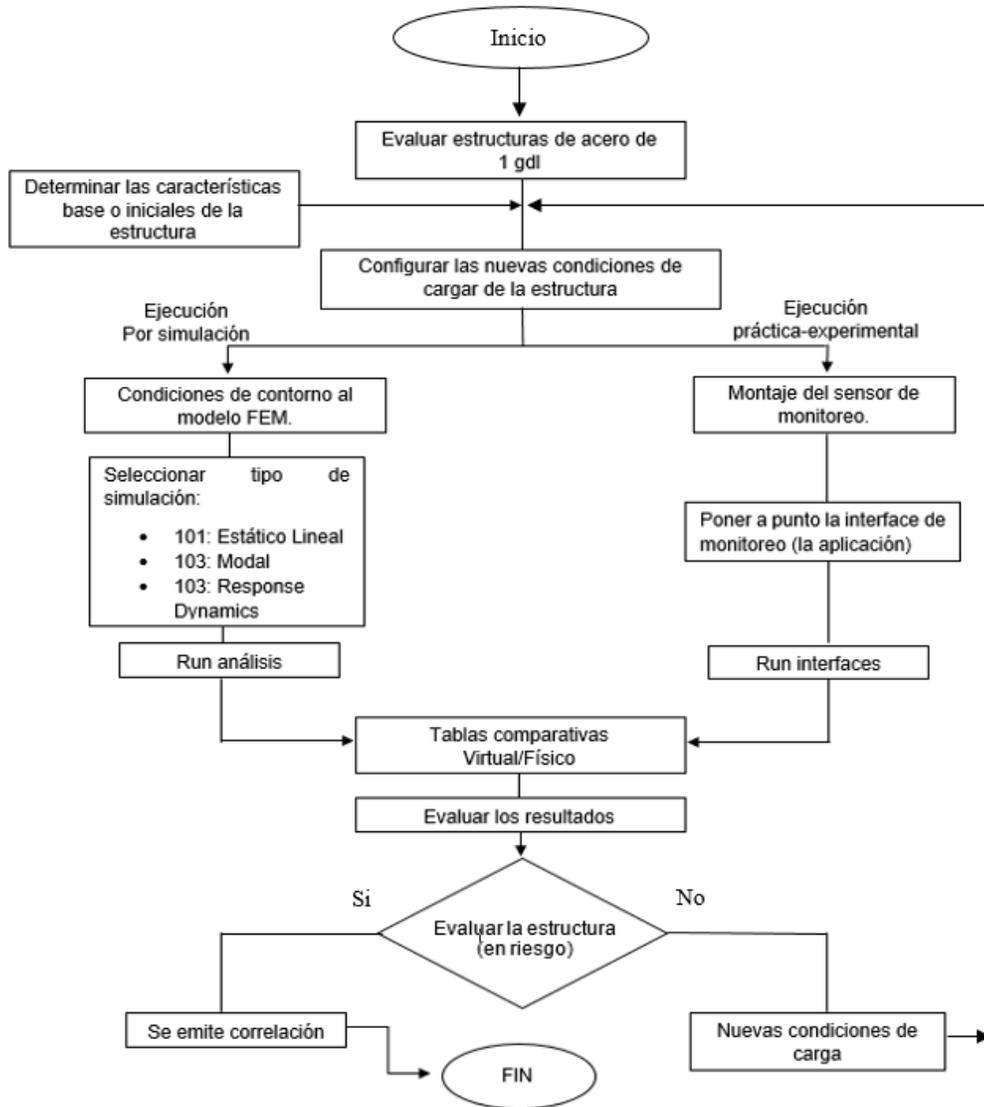


Figura 1b. Diagrama secuencial de desarrollo del proyecto.

2.1 Materiales

Se realizó un análisis estático lineal, debido a que el comportamiento de interés de la estructura de 1 GDL, se encuentra dentro de la zona lineal del acero, que se considera como el metal más importante utilizado para fines estructurales porque combina una alta resistencia, tanto en tensión como en compresión, con gran rigidez (módulo de elasticidad elevado), y facilidad de fabricación. Además, el acero es un material dúctil por naturaleza, que tiene, además un comportamiento estable bajo aplicaciones de carga y tiene una relación resistencia/peso favorable [6].

2.2 Descripción de la estructura de 1 GDL

La estructura metálica de alta rigidez (figura 2) fue ensamblada a partir de elementos estructurales fabricados de ángulo y solera comerciales. Material en el cual se ha considerado un módulo de elasticidad de 200 GPa y un módulo de Poisson de 0.26.

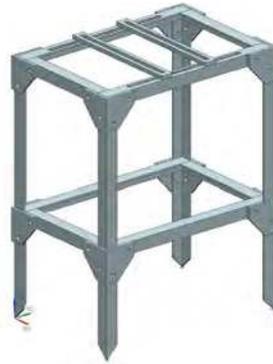


Figura 2. Representación gráfica de la estructura metálica de 1 GDL con alta rigidez.

Las características físicas y geométricas de la estructura de alta rigidez son:

<i>Volumen</i>	= 527913.04 mm ³
<i>Área</i>	= 377459.12 mm ²
<i>Masa</i>	= 4.13 kg
<i>Peso</i>	= 40.53 N
<i>Centro de masa</i>	= 103.18, 153.18, 256.06 mm
<i>Densidad</i>	= 0.00007831 kg/mm ³

En una segunda batería de simulaciones se realizaron modificaciones a la estructura de 1 GDL, para afectar su comportamiento dinámico, disminuyendo su rigidez (figura 3).



Figura 3. Representación gráfica de la estructura metálica de 1 GDL con baja rigidez.

En este segundo prototipo virtual se puede observar que sus propiedades de masa y volumen cambiaron, por lo cual en las gráficas de resultados sus variaciones con respecto a la frecuencia natural también existen diferencias. Las características físicas y geométricas de la estructura son:

<i>Volumen</i>	= 359343.55 mm ³
<i>Área</i>	= 256453.83 mm ²
<i>Masa</i>	= 2.81 kg
<i>Peso</i>	= 27.59 N
<i>Centro de masa</i>	= 103.18, 153.18, 302.13 mm
<i>Densidad</i>	= 0.00007831 kg/mm ³

Para comprobar las variaciones de frecuencia natural, se realizaron la simulación modal y la simulación de respuesta dinámica omitiendo la simulación estática-lineal (sin carga alguna), por

considerar y determinar que esta última modifica los valores de la frecuencia natural, observados en el cuadro 1.

2.3 Consideraciones sobre la configuración propuesta

Como parte de la validación de las simulaciones realizadas y sus resultados, se ejecutó la modalidad de corrida free-free modal-respuesta dinámica, sin cargas para determinar las frecuencias naturales (figuras 4, 5 y 6), encontrando que las incorporaciones de cargas estáticas influyen en la determinación de la frecuencia natural (Cuadro 1).



Figura 4. Corrida en modo Free-Free, sin cargas.

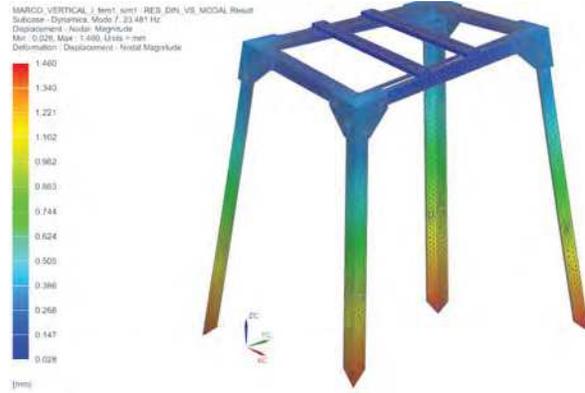


Figura 5. Corrida en modo Respuesta Dinámica, sin cargas.



Figura 6. Corrida en modo Respuesta Dinámica, con cargas.

Cuadro 1. Síntesis de los resultados de la corrida para obtener las frecuencias naturales.

Frecuencia natural	Hz
Modal Free-Free, sin carga motor	23.48
Respuesta Dinámica Free-Free, sin carga por motor	23.48
Respuesta Dinámica, con carga por motor	39.77



3. Resultados

3.1 Precarga tonillos

Para estudiar el comportamiento en las conexiones tornillo dentro de las simulaciones es necesario considerar una precarga en las mismas, una base teórica permite tener un resultado a esperar dentro de los diferentes escenarios a considerar. La teoría con respecto a la precarga de tornillo se puede idealizar de la siguiente manera [8]:

$$F_b = \frac{k_b * P}{k_b + k_s} + F_i \quad (1)$$

Donde:

- F_b = Fuerza axial total en el tornillo
- k_b = Rigidez del tornillo
- k_s = Rigidez de los elementos unidos
- P = Carga externa aplicada
- F_i = Precarga del tornillo

Los datos de entrada para la ecuación 1, se toman de la tabla 2. Asimismo, dentro de esta idealización, se considera la rigidez del tornillo igual a la rigidez de los elementos unidos, ambas se obtienen por medio de:

$$k_b = \frac{A * E}{L} \quad (1b)$$

Se asume que la precarga del tornillo y la carga externa aplicada actúan en la misma sección sólida del material, con esto se puede decir que la fuerza axial final en el tornillo precargado será la precarga del tornillo y una fracción de la carga externa. La ecuación 1 es válida siempre y cuando exista compresión entre las partes precargadas, si la carga externa ocasiona que los elementos precargados se separen entonces está carga estará aplicada únicamente en el tornillo. Debido a esto, es necesario conocer la fuerza entre las partes precargadas (F_s), la siguiente formula proporciona el dato:

$$F_s = \frac{k_b * P}{k_b + k_s} + F_i \quad (2)$$

Se considera que, si la precarga es igual o mayor a la carga externa aplicada, las partes precargadas (uniones) se mantendrán juntas. Una de las razones principales por la cual se aplica una precarga de tornillo es para evitar la fatiga del mismo, si se considera que el esfuerzo del tornillo es una combinación lineal de la precarga y una fracción de la carga externa, el esfuerzo se ve considerablemente reducido.

3.2 Gráficas de las 13 simulaciones (Sin marco medio) menor rigidez

Las siguientes gráficas son el resultado del comportamiento identificado mediante simulación en el software NX siemens. Con respecto a la evaluación de un marco de baja rigidez (como estructura de 1 GDL) y diversas configuraciones de la simulación en cuanto a la precarga de los tornillos y la evaluación de la respuesta dinámica, los esfuerzos y las deformaciones.

La figura 7, muestra el desplazamiento total (ejes X, Y, Z) de la estructura, donde se identifica que la configuración 4 y 5 (conexiones tornillo con contacto) considera la deformación que produce la precarga de 70 kilogramos y aprietes varios (respectivamente) aplicada a los tornillos. Por lo tanto, es coincidente que el resto de las simulaciones indiquen un desplazamiento prácticamente nulo, debido a que son rígidas y sólidas. Siendo posible afirmar que las conexiones tornillo con una alta precarga



representaran una condición de desplazamiento que tiende a los límites máximos de esfuerzo dentro de la zona lineal.



Figura 7. Gráfica que muestra el desplazamiento total de la estructura de menor rigidez.

La figura 8, corresponde a la fuerza de reacción total de los cuatro puntos de restricción, las cuales se realizaron en los ejes X, Y y Z. Sin embargo, para efectos de la observación y verificación del equilibrio externo se consideran válidos solo para el eje vertical, ya que en los ejes X y Y, la fuerza de reacción involucra el efecto de la precarga de los tornillos, sobre todo la asociada a la precarga de 70 kilogramos. Considerando entonces que la fuerza de reacción corresponde practicante a la fuerza estática aplicada a la estructura en la dirección del eje Z ($36.56 \approx 36.4$), con lo cual se verifica el equilibrio externo y se corrobora los resultados del desplazamiento, por Ley de Hooke. Por consiguiente, se comprueba que una excesiva precarga afecta el comportamiento estático lineal inclusive en las fuerzas de reacción.



Figura 8. Gráfica que muestra la fuerza de reacción de la estructura de menor rigidez.

La figura 9, corresponde al criterio de falla de Von Mises donde el esfuerzo de fluencia a la tensión es de 226 MPa y el máximo esfuerzo al cual son sometidas las conexiones de la estructura es de 5854 MPa, este se obtiene bajo una precarga de 70 kilogramos y considerando la carga del motor. Lo que nos muestra que bajo estas condiciones las zonas de las conexiones, sobrepasan los límites mecánicos, condición que fue verificada por un torquímetro, sobre las conexiones tornillos del prototipo de manera física, evidenciando falla por sobre fuerza axial. Reduciendo el factor de seguridad a $F_s=0.04$.



Figura 9. Gráfica que muestra el esfuerzo de Von-Mises de la estructura de menor rigidez.

En correspondencia al hecho de que las zonas de conexión sobrepasaron sus límites mecánicos, la deformación en la estructura es nula en casi todas las condiciones (figura 10), con excepción de cuando se considera una estructura idealizada como sólida.



Figura 10. Gráfica que muestra la deformación total de la estructura de menor rigidez.

La fuerza axial que desarrolla el tornillo ante la simulación de respuesta dinámica, se presenta cuando los aprietes son máximos por ejemplo (70 kilogramos) en comparativa cuando el apriete no es significativo (0.001 kilogramos) correspondiente a la conexión tornillo. Siendo de alta importancia que al ensamblar estructuras de 1 GDL por conexiones tornillo o perno (de sujeción), estas no sufran precargas que lleven a la conexión a sus límites mecánicos, como lo es este caso, en donde esas sujeciones llegan al punto de falla por fuerza axial, como se aprecia en la figura 11.



El efecto del esfuerzo cortante con precargas altas se expresa como un valor fuera de los límites mecánicos considerando que 113 MPa es el esfuerzo cortante máximo que soporta el tornillo el factor de seguridad con 2000 MPa (figura 12) es de $F_s=0.06$. Valor que establece una limitada condición de seguridad.



Figura 11. Gráfica que muestra la fuerza axial a la que están sometidos los tornillos de la estructura de menor rigidez.



Figura 12. Gráfica que muestra fuerza cortante a la que están sometidos los tornillos de la estructura de menor rigidez.

Las figuras 13, 14 y 15 muestran que el apriete de los tornillos no podrá igualar la respuesta dinámica de una estructura rígida y menos de la sólida. Por lo tanto, la idea adecuada de que la firmeza de las conexiones por sobre apriete es errónea y no garantiza la seguridad ante eventualidades de vibraciones aleatorias. Este hecho debe ser una consideración de alta relevancia entre la población, pues es requerido para cada modificación estructural sobre la original de un estudio detallado de las nuevas incorporaciones ya sean por medios de sujeción o por soldadura. Sistemas constructivos de alta recurrencia en las ampliaciones o modificaciones de las estructuras metálicas de 1 GDL.



Figura 13. Gráfica que muestra el primer modo de vibrar de la estructura de menor rigidez.



Figura 14. Gráfica que muestra el segundo modo de vibrar de la estructura de menor rigidez.



Figura 15. Gráfica que muestra el tercer modo de vibrar de la estructura de menor rigidez.



Tabla 2. Valores de entrada para la ecuación 1.

Área (mm ²)	15.83
Módulo de Young (E) (Pa)	2.00E+11
L (mm)	9.53
kb (rigidez del tornillo)	3.32E+11
ks (rigidez de la estructura)	3.32E+11
P (carga externa aplicada) (N)	36.40
Fi (precarga de tornillo) (N)	392.40
Fb (fuerza total axial en el tornillo) (N)	410.60
Fb (fuerza total axial en el tornillo) (kg)	41.86

Se realizaron comprobaciones mediante la tabla 4, correspondientes al “cálculo del torque” donde se mantiene el valor de 40 kg de precarga y se agrega una precarga de 70 kg, valor que ocasiona que el tornillo sobrepase sus límites mecánicos, teniendo como resultados lo siguiente:

Tabla 3. Determinación de la precarga mediante la ecuación 1.

Área (mm ²)	Módulo Young (E) (Pa)	L (mm)	kb (rigidez tornillo)	ks (rigidez estructura)	P (carga externa N)	Fi (precarga N)	kb *P	Kb+Ks	Fb Fuerza axial (N)	Fb Fuerza axial (kg)
15.83	2.00E+11	9.53	3.32E+11	3.32E+11	36.40	392.40	1.21E+13	6.65E+11	410.60	41.86

La siguiente fórmula (ec. 3) se ocupó para los cálculos del torque fue tomada del libro de “Diseño en ingeniería mecánica de Shigley” [7].

$$T = K * F * D \quad (3)$$

Donde:

- $T =$ Torque (en Nm)
- $k =$ Coeficiente del par de torsión
- $F =$ Fuerza axial aplicada (en kN)
- $D =$ Diámetro del tornillo

Tabla 4. Valores y resultados obtenidos con la ec. 3.

Fuerza axial (kg)	K	F (kN)	D (mm)	Torque (Nm)	Torque (kg.cm)
40	0.2	0.3924	6.35	0.498	5.08
70	0.2	0.6867	6.35	0.872	8.89

Por medio de la formulación anterior (ec. 3), se hace un procedimiento inverso para obtener la fuerza axial aplicada al tornillo (F):

$$F = \frac{T}{K D} \quad (4)$$



Tabla 5. Valores y resultados obtenidos con la ec. 4

K	D (mm)	Torque (Nm)	F (kN)	Fuerza axial (kg)
0.2	6.35	0.498	0.392	39.98
0.2	6.35	0.872	0.686	70

3.3 Resultados simulación NX

Tabla 6. Resultados de la simulación virtual con alta rigidez, NX Siemens [8].

	No.	Desplazamiento (mm)	Rotación (°)	Fuerza de reacción (N)	Stress Element Nodal (Mpa)	Strain Element Nodal	Beam Resultants Nodal (N)	ESFUERZO AXIAL TEÓRICO	Beam Resultants Nodal (N)	ESFUERZO CORTANTE TEÓRICO
		Magnitud	Magnitud	Magnitud	Magnitud	Magnitud	Axial Force NXX		Shear Force QXZ	
Respuesta dinámica sin precarga	1	0.031	0.020	9.14	5.43	2.26E-05	10.14	0.325	14.88	0.16
Respuesta dinámica con aprietes varios	2	0.031	0.021	9.12	30.65	0.0001277	405.09	12.996	13.84	6.50
Respuesta dinámica con precarga 40 kg	3	0.031	0.021	9.12	34.10	0.0001421	402.57	12.915	15.30	6.46
Respuesta dinámica con precarga 70 kg	4	53.28	14.73	86.44	5853.99	0.000000000	9421.74	302.27	1983.92	151.13

La tabla 6, muestra los resultados de la simulación del software NX Siemens, permitiendo establecer métricas de validación, mediante los datos ordenados en las columnas de la tabla. Los resultados Axial Force y Shear Force obtenidos por simulación, se contrastan con los valores teóricos de las columnas esfuerzo axial teórico y esfuerzo cortante teórico, observando el incremento de los valores con respecto al valor de incremento de la precarga. Siendo posible establecer la relación que a un mayor apriete de precarga existirá un mayor esfuerzo cortante y una mínima deformación, pues tiende a fallar por cizalladura la conexión tornillo. Lo cual es demostrable ya que el único valor que se decrementa es el de la deformación del elemento (Strain Element Nodal).

4. Conclusiones

Se concluye que el dispositivo/prototipo virtual es funcional, para crear conciencia en la población desde diversos niveles educativos de licenciatura y posgrado, ya que es de fácil manejo y de manera eficiente y simple ilustra las consecuencias de alterar el diseño inicial de las estructuras de 1 GDL, ya sea por adición de masa o reducción de la rigidez, cuando el medio de sujeción entre elementos estructurales son las conexiones tornillo.

Asimismo, la simulación virtual desde la perspectiva didáctica en la educación superior y de posgrado, permite fomentar el criterio de certidumbre ante riesgos sísmicos y la prevención de daños estructurales, que pongan en riesgo la vida humana, su salud emocional y la estabilidad de la infraestructura.

Agradecimientos

Al programa institucional “Proyectos VIEP” de la Benemérita Universidad Autónoma de Puebla, que proporcionó en financiamiento económico para realizar el proyecto.



Referencias

- [1] G. Vélez, G. E. S., D. Szwedowicz y J. Bedolla, «Identificación de modos cercanos de vibración,» MEMORIAS DEL XVI CONGRESO INTERNACIONAL ANUAL DE LA SOMIM, p. 10, 2010.
- [2] Y. Bao, Z. Chen, S. Wei, Y. Xu, Z. Tang y H. Li, «The State of the Art of Data Science and Engineering in Structural,» Elsevier LTD, p. 9, 28 Febrero 2019.
- [3] M. Couchaux, P. Martin y A. Rodier, «Static and cyclic behaviour of bolted gusset plate connections:,» ce/papers 1, nº 2 & 3, p. 10, 15 Septiembre 2017.
- [4] Corona Sandoval M.A. y Endo Hernández M.T. (2017). Conexiones de marcos para estructuras de acero. UNAM. Recuperado [28/09/2020] de:
<http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/12560/Tesis.pdf?sequence=1>
- [5] Günthel T. (2017). Modelado de una conexión tornillo pretensado. Dlubal. Recuperado [28/09/2020] de: <https://www.dlubal.com/es/soporte-y-formacion/soporte/base-de-datos-de-conocimientos/001499>
- [6] AHMSA, Manual de diseño para a construcción con acero, AHMSA, 2013.
- [7] J. E. Shigley, R. G. Budynas y J. K. Nisbett, Diseño en ingeniería mecánica de Shigley, Mc Graw Hill, 2012.
- [8] N. S. Gokhale, S. S. Deshpande, S. V. Bedekar y A. M. Thite, Practical Finite Element Analysis, The Leela Palace, Bengaluru: 4th India/ASEAN, 2008.



API testing usando REST Assured, Cucumber y Java

Sánchez Mares Gilberto.

Universidad DaVinci, Departamento de Sistemas Computacionales
gsanchez19@udavinci.edu.mx

Resumen

Las interfaces de programación de aplicaciones web (API) permiten que los sistemas interactúen entre sí a través de la red. Las API web modernas a menudo se adhieren al estilo arquitectónico REST, y se las conoce como API web RESTful. Las API web RESTful se componen en varios recursos (por ejemplo, un video en la API de Netflix) que los clientes pueden manipular a través de interacciones HTTP. Probar las API web es fundamental pero desafiante debido a la dificultad de evaluar la exactitud de las respuestas de la API. Para poder reducir la brecha de lo que se desarrolla con lo que el negocio quiere, se creó un framework que combina Behavior-Driven Development Utilizando Cucumber y Gherkin) y el patrón de diseño, Arrange, Act, Assert, los cuales se combinan con la biblioteca de REST Assured para crear pruebas automatizadas programadas en Java. La intención principal es que se pueda difundir, utilizar y mejorar. Las principales ventajas son: puede ser utilizado por ingenieros de pruebas que tengan conocimientos técnicos o aquellos que no los tengan, es de fácil implementación, con un tiempo de aprendizaje muy corto y reduce costos en las tareas de pruebas.

Palabras clave: Application Programming Interfaces, Automatización de Pruebas, Behavior-Driven Development, Cucumber, Gherkin, Java, REST, REST Assured.

1. Introducción

Las Application Programming Interfaces (API por sus siglas en inglés e interfaces de programación de aplicaciones por su traducción al español) especifican cómo acceder a los servicios y datos a través de la red, normalmente mediante servicios web [1][2]. Las API web están extendiéndose rápidamente como un elemento clave para fomentar la reutilización, la integración y la innovación, lo que permite nuevos modelos de consumo como las aplicaciones móviles web, internet de las cosas, etc. Empresas como Tesla, Facebook, Twitter, Google, Amazon o Netflix reciben miles de millones de llamadas API todos los días desde miles de aplicaciones y dispositivos de terceros diferentes, lo que constituye más de la mitad de su tráfico total [1]. Muchas empresas también están exponiendo sus activos existentes como API privadas, lo que permite a sus propios desarrolladores crear aplicaciones móviles, sociales o en la nube innovadoras [1][3]. Las API web suelen cumplir con el estilo arquitectónico REpresentational State Transfer (REST por sus siglas en inglés y Transferencia de Estado Representacional por su traducción al español), y se denominan API web RESTful [4]. Las API web RESTful comprenden un conjunto de los llamados servicios web RESTful, en el cual cada servicio implementa una o más operaciones de creación, lectura, actualización o eliminación (CRUD por sus siglas en inglés) sobre un recurso, por ejemplo, una orden de compra en la API de Mercado Libre. El sitio web ProgrammableWeb [5] un popular repositorio de API, indexa más de 5000 API web RESTful de múltiples dominios, como compras, finanzas, redes sociales o telefonía.

A medida que las API web se están convirtiendo progresivamente en la piedra angular de la integración de software, su validación se vuelve más crítica. En este contexto, la detección rápida de errores es de suma importancia para aumentar la calidad de los productos internos y las aplicaciones de terceros. Sin embargo, probar las API web es un desafío principalmente debido a la dificultad de evaluar si la salida de una llamada a la API es correcta, es decir, el problema de Oracle [6][7][8]. Por ejemplo, considerando la API web del popular servicio de transmisión de música Spotify [9]. Un ejemplo



que podemos tomar es una búsqueda de canciones con la consulta "regresa" devuelve 230 coincidencias en total: ¿Es correcto este resultado? ¿Todas las canciones del conjunto de resultados contienen la palabra clave? ¿Hay canciones que contengan la palabra clave no incluidos en el conjunto de resultados? Responder estas preguntas es difícil, incluso con conjuntos de resultados pequeños, y a menudo no es factible cuando los resultados se cuentan por miles o millones.

En este artículo, presentamos un framework (marco de trabajo por su traducción al español) de automatización de pruebas la detección de fallas en las API RESTfulWeb (en adelante también denominadas simplemente API web). Se observa que las API web tienen una semántica muy clara, especificada como operaciones CRUD sobre recursos, y un uso muy consistente de parámetros para operaciones estándar como filtrado, ordenamiento y paginación [2][10][11][12][13]. Como resultado, presentamos el uso combinado de herramientas, lenguajes de programación, patrones de diseño y frameworks para poder ser utilizados con facilidad a la hora de generar diferentes tipos de pruebas en las API. La herramienta que se utiliza en el presente trabajo es REST Assured, la cual nos permite validar y realizar pruebas de servicios REST con el lenguaje de programación Java, de una forma tan sencilla cómo se puede validar en lenguajes dinámicos como Ruby o Groovy [14]. Se utiliza el patrón de diseño Arrange, Act, Assert (AAA por sus siglas en inglés, Organizar, Actuar, Comprobar, por su traducción al español), el cual sugiere dividir un método de prueba en las tres secciones de las que recibe el nombre [15]. Para poder hacer el framework entendible por personas con conocimientos técnicos y personas con conocimiento del negocio en el que se implementará, se utiliza Cucumber y Gherkin el cuál utiliza Behaviour-Driven Development (BDD por sus siglas en inglés y Desarrollo Impulsado por Comportamiento por su traducción al español) que es un conjunto de prácticas diseñadas para reducir el comportamiento que no es útil en el desarrollo de software. En esencia, BDD implica pensar en los resultados deseados y la estructura API antes de escribir cualquier código. Las pruebas automatizadas son un subproducto del proceso que impulsa la implementación [16].

Se propone un framework para la aplicación del enfoque a cualquier API Web siguiendo el patrón arquitectónico REST. Está destinado a ser utilizada como complemento de las herramientas existentes y los métodos de prueba de vanguardia utilizados por los desarrolladores e ingenieros de prueba, ya sea en la empresa o en empresas de prueba de API de terceros.

2. Behavior-Driven Development

El Behavior-Driven Development (BDD), es un conjunto de prácticas diseñadas para reducir la brecha de lo que se construye con lo que se debe construir en el desarrollo de software. En esencia, BDD implica pensar en los resultados deseados y la estructura API antes de escribir cualquier código. Las pruebas automatizadas son un subproducto del proceso que impulsa la implementación [16].

BDD se puede dividir en dos partes:

Descubrimiento deliberado: una conversación entre propietarios del producto, analistas de negocios, usuarios, programadores, probadores y otros que culmina en ejemplos concretos que se pueden convertir en especificaciones ejecutables.

Test-Driven Development (TDD o Desarrollo basado en pruebas por su traducción al inglés): pruebas de lenguaje natural, a menudo escritas en Gherkin, que ayudan a los programadores y no programadores a explicar los resultados deseados de ejemplos concretos sin ambigüedad.

BDD a menudo ocurre al comienzo del ciclo de desarrollo de software después de que se escribe una historia de usuario. La reunión de descubrimiento deliberada produce pruebas altamente automatizadas que describen los resultados de una historia de usuario. Los desarrolladores utilizan esto como un punto de partida para el desarrollo basado en pruebas de nivel inferior, o TDD. Se centra todas las actividades de desarrollo en la entrega de comportamientos: lo que debe hacer un sistema, descrito de manera que los desarrolladores y los expertos en el dominio hablen el mismo idioma. Los marcos BDD permiten a los usuarios representar el comportamiento del sistema requerido como historias de



usuario ejecutables y los criterios de aceptación como escenarios ejecutables adjuntos a historias de usuario [17].

2.1 Gherkin

Cuando creamos software para personas (llamémosles partes interesadas), es notoriamente difícil averiguar exactamente qué quieren que construyamos. La parte más difícil de construir un sistema de software es decidir con precisión qué construir [18]. Al trabajar en proyectos en los que, debido a un malentendido, lo que se codificó, debió ser desechado. Una mejor comunicación entre los desarrolladores y las partes interesadas es esencial para ayudar a evitar este tipo de pérdida de tiempo. Una técnica que realmente ayuda a facilitar esta comunicación es el uso de ejemplos concretos para ilustrar lo que queremos que haga el software.

Al utilizar ejemplos del mundo real para describir el comportamiento deseado del sistema que se debe construir, se debe basar en el lenguaje y la terminología que tiene sentido para las partes interesadas: se debe hablar su idioma. Cuando se indican ejemplos, realmente se involucra a los interesados como si ellos usaran el sistema, y eso significa que pueden comenzar a darnos comentarios e ideas útiles antes de que se haya escrito una línea de código.

Gherkin utiliza un conjunto de palabras clave especiales para dar estructura y significado a las especificaciones ejecutables. Cada palabra clave se traduce a muchos idiomas hablados; en esta referencia usaremos inglés. La mayoría de las líneas de un documento comienzan con una de las siguientes palabras clave.

- **Feature**
- **Rule**
- **Scenario**
- **Given,When,Then,And, But** para los pasos
- **Background**
- **Scenario Outline** o **Scenario Template**
- **Examples**
- **“”** para documentar largas cadenas de caracteres
- **|** para tablas de datos
- **@** para etiquetas
- **#** para comentarios

Un ejemplo de documento es el siguiente:

Feature: Guess the word

The first example has two steps

Scenario: Maker starts a game

When the Maker starts a game

Then the Maker waits for a Breaker to join

The second example has three steps

Scenario: Breaker joins a game

Given the Maker has started a game with the word "silky"

When the Breaker joins the Maker's game

Then the Breaker must guess a word with 5 characters

La parte final (después de la palabra clave) de cada paso se corresponde con un bloque de código, denominado definición de paso.

2.2 Frameworks de BDD



Los frameworks de BDD leen especificaciones ejecutables escritas en texto sin formato (por ejemplo, Gherkin u otros lenguajes específicos del dominio, o DSL) y validan que el software cumple lo que prometen esas especificaciones. Se genera un informe que indica el éxito o el fracaso de cada especificación.

Existe un framework de automatización BDD para todos los principales lenguajes de programación. Estos marcos son generalmente derivados de Cucumber o JBehave, pero algunos son corredores de especificaciones que no son de Gherkin y tienen características similares a las de BDD. Todos se esfuerzan por lograr el mismo objetivo, pero utilizan diferentes metodologías [17].

En la Tabla 1 de muestran a algunos de los frameworks BDD más populares para cada lenguaje de programación principal.

Tabla 1. Frameworks BDD más populares y su lenguaje de programación.

Lenguaje	Framework	Lenguaje	Framework	Lenguaje	Framework
.Net y C#	SpecFlow	JavaScript	Jasmine	Python	Behave
	xBehave.net		Mocha		Lettuce
	LightBDD		Cucumber.js		Pytest-BDD
	NSpec		Yadda		Radish
	Concordion		Vows		
Lenguaje	Framework	Lenguaje	Framework	Lenguaje	Framework
Java & JVM	Cucumber-JVM	PHP	Behat	Ruby	Cucumber
	JBehave				Spinach
	JGiven		Codeception		
	Concordion				

3. RESTful Web APIs

REpresentational State Transfer (REST) es un estilo arquitectónico para sistemas hipermedia distribuidos como la Web [4]. Las API web que se agrega a las restricciones de la arquitectura REST se denominan API web RESTful. Las API web RESTful se descomponen en varios servicios web RESTful, donde cada servicio implementa una o más operaciones CRUD sobre un recurso. Un recurso es cualquier cosa que pueda estar expuesta a la Web, como un video, una foto o una orden de compra [19].

Los recursos generalmente se identifican mediante un Uniform Resource Identifier (URI o Identificador uniforme de recursos por su traducción al español), que los hace direccionables y manipulables mediante un protocolo de aplicación, generalmente HTTP. Un **endpoint** (punto final por su traducción al español) de API es un URI único que identifica uno o más recursos. La mayoría de las API web RESTful siguen un conocido conjunto de pautas de diseño [2][10][11][12][13] que incluyen la implementación de los métodos HTTP estándar de la siguiente manera:

- **GET**: se utiliza para recuperar uno o más recursos.
- **POST**: crea un recurso, si se tiene éxito, devuelve el recurso recién creado
- **PUT**: actualiza un recurso, si se tiene éxito, devuelve el recurso actualizado.
- **DELETE**: borra un recurso.



- **PATCH:** aplica una actualización parcial al recurso.

Como ejemplo, la siguiente solicitud obtiene información sobre un usuario en la página de pruebas GO REST [20]: “GET https://gorest.co.in/public-api/users/123”. El URI (es decir, el endpoint de la API) identifica al usuario mediante un identificador, y el método HTTP (GET) especifica la operación a realizar sobre el recurso (lectura). Los URI pueden incluir parámetros para realizar ciertas operaciones sobre recursos como el filtrado, la ordenación y la paginación. Los parámetros se pueden incluir como parte de la ruta de URI (como el identificador del usuario, adjunto al final de la URI anterior) o utilizando parámetros de URI estándar de la forma parámetro = valor.

Los recursos se pueden representar utilizando diferentes formatos como JSON, XML o XHTML. Nos centramos en JSON en este documento. La JavaScript Object Notation (JSON por sus siglas en inglés y notación de objetos JavaScript por su traducción al español) es un formato de intercambio de datos ligero y legible por humanos compuesto por pares propiedad-valor. La Figura 1 muestra un extracto del recurso del artista en formato JSON obtenido como respuesta a la solicitud de GO REST presentada anteriormente. Tenga en cuenta que los valores de datos pueden incluir objetos (delimitados con llaves), matrices (delimitado con corchetes) y referencias a otros URI, lo que permite navegar de un recurso a otro.

```
{
  "code": 200,
  "meta": null,
  "data": {
    "id": 123,
    "name": "Allasani",
    "email": "allasani.peddana@15ce.com",
    "gender": "Female",
    "status": "Active",
    "created_at": "2020-09-30T03:50:06.020+05:30",
    "updated_at": "2020-09-30T23:20:40.906+05:30"
  }
}
```

Figura 1. Representación de un JSON sobre GO REST.

Las API web suelen proporcionar documentación en línea que describe cómo utilizar la API. Esto generalmente incluye información sobre los recursos disponibles, URI, métodos HTTP, parámetros, formato de intercambio de datos, códigos de estado HTTP, datos de autenticación y posibles errores. Además, algunas API web incluyen ejemplos de solicitudes y respuestas o incluso una Interfaz web para que los clientes llamen a la API desde un navegador web y verifiquen la respuesta.

4. Pruebas de API

Son un tipo de prueba de software que valida las API's. El propósito de las pruebas de API es verificar la funcionalidad, confiabilidad, rendimiento y seguridad de las interfaces de programación. En API Testing, en lugar de utilizar entradas y salidas de usuario estándar (teclado), utiliza software para enviar llamadas a la API, obtener resultados y anotar la respuesta del sistema. Son muy diferentes de las pruebas de GUI y no se concentran en la apariencia de una aplicación. Se concentra principalmente en la capa de lógica empresarial de la arquitectura de software (Figura 2) [21].



Figura 2. Capas involucradas en un programa de software.

4.1 ¿Cómo probar una API?

Las pruebas de API deben cubrir al menos los siguientes métodos de prueba además del proceso ciclo de vida de pruebas del software habitual

- **Prueba de descubrimiento:** el grupo de pruebas debe ejecutar manualmente el conjunto de llamadas documentadas en la API, como verificar que un recurso específico expuesto por la API se puede enumerar, crear y eliminar según corresponda
- **Prueba de usabilidad:** esta prueba verifica si la API es funcional y fácil de usar. ¿la API también se integra bien con otra plataforma?
- **Prueba de seguridad:** esta prueba incluye qué tipo de autenticación se requiere y si los datos confidenciales están cifrados a través de HTTP o ambos
- **Pruebas automatizadas:** las pruebas de API deben culminar en la creación de un conjunto de scripts o una herramienta que se pueda utilizar para ejecutar la API con regularidad.
- **Documentación:** el equipo de prueba debe asegurarse de que la documentación sea adecuada y proporcione información suficiente para interactuar con la API. La documentación debe ser parte del producto final.

5. Resultados

El framework desarrollado se basa en las tecnologías que se muestran en la Figura 3, para generar las pruebas se utiliza Cucumber y Gherkin para poder entablar una relación de entendimiento directa entre el equipo de desarrollo y los encargados de las reglas de negocio, una vez elegida la herramienta para implementar BDD, se utiliza REST Assured en conjunto con el patrón AAA, para poder generar la implementación de los métodos que se encargarán de ejecutar los pasos que se hayan implementado en Cucumber.

La página que se utilizará para realizar pruebas es <https://gorest.co.in/>, la documentación ahí encontrada es suficiente para hacer pruebas, sin embargo, no es tan basta como para generar casos de prueba que chequen funcionalidades como la seguridad o el tiempo de ejecución de cada API, así como mensajes customizados para cada respuesta HTTP.

El proyecto se divide en la capa del framework y la capa de las pruebas (Figura 4), en la capa del framework tenemos bibliotecas de utilidades que no ayudan a leer archivos de configuración, a tener los métodos que se utilizarán para realizar los pasos dentro de los archivos feature. donde se mencionan los métodos GET, POST, etc.

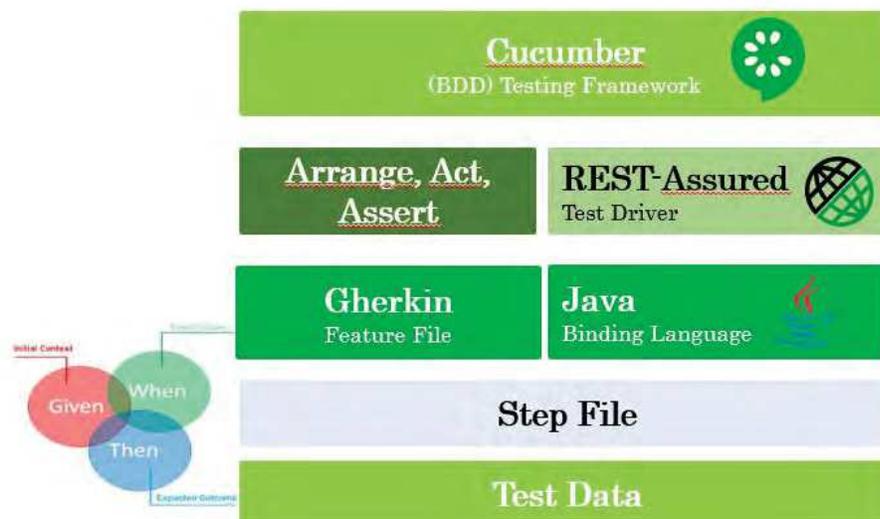


Figura 3. Tecnologías utilizadas en el framework automatizado.

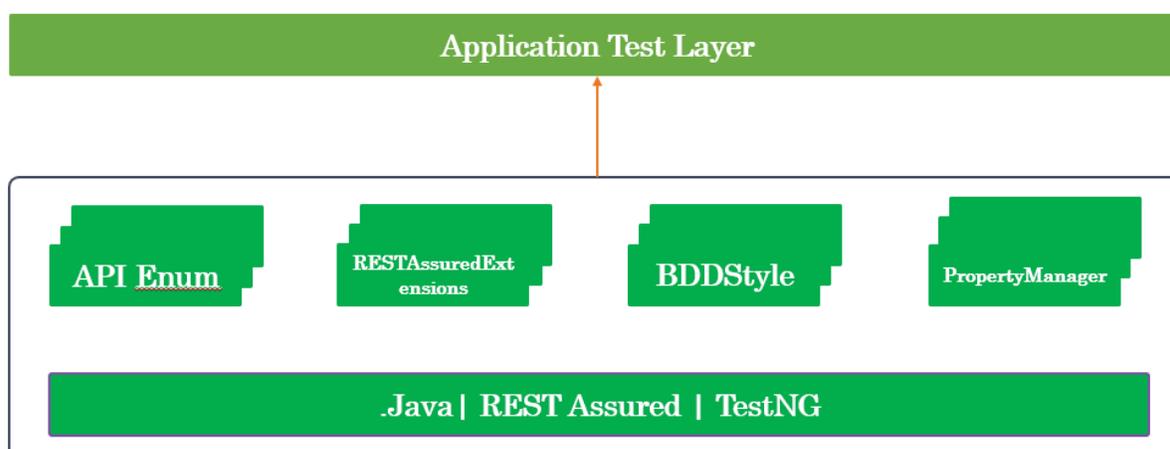


Figura 4. Capas del Proyecto.

Dentro de los folders donde está la capa de pruebas, el proyecto se divide en los siguientes folders (Figura 5):

- **Feature:** se encuentran los archivos .feature, en los cuales se implementan los pasos del negocio en un formato Gherkin.
- **Runner:** aquí se encuentra el archivo que ejecutará las pruebas.
- **Steps:** se encuentran los archivos .java en los cuales se implementa la lógica de los pasos que se crearon en el archivo .feature.

5.1 REST Assured.

REST Assured por sí solo, tiene un formato BDD para desarrollar las pruebas, aunque este formato es sólo óptimo para aquellos que tienen experiencia en el desarrollo. También nos brinda la oportunidad de crear métodos del tipo No BDD para poder utilizar otros frameworks y adecuarlos a nuestras necesidades.

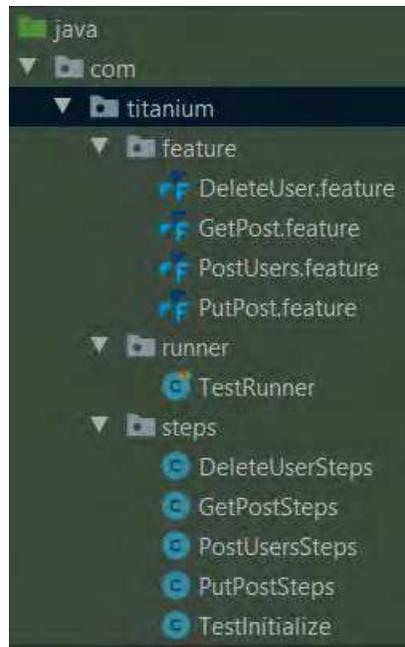


Figura 5. Estructura de folders de la capa de pruebas.

El framework cuenta con dos archivos .java, los cuales contienen las formas BDD y No BDD mencionadas con anterioridad, a continuación, se muestran métodos de ejemplo para visualizar los dos tipos de implementaciones mencionadas anterior mente.

- **BDDStyleMethod.java**

```
public static void simpleGetPost(String postNum){  
    given()  
        .contentType(ContentType.JSON)  
    .when()  
        .get(String.format("https://gorest.co.in/public-api/posts/%s", postNum))  
    .then()  
        .body("data.user_id", is(3));  
}
```

- **RestAssuredExtensions.java**

```
public static ResponseOptions<Response> getOps(String url) {  
    return request.get(url);  
}
```

5.2 Cucumber.

Es una herramienta de prueba que admite BDD. Ofrece una forma de redactar pruebas que cualquiera puede entender, independientemente de sus conocimientos técnicos. En BDD, los usuarios (analistas de negocios, propietarios de productos) primero escriben escenarios o pruebas de aceptación que describen el comportamiento del sistema desde la perspectiva del cliente, para que los propietarios de productos los revisen y aprueben antes de que los desarrolladores escriban sus códigos (Figura 6).

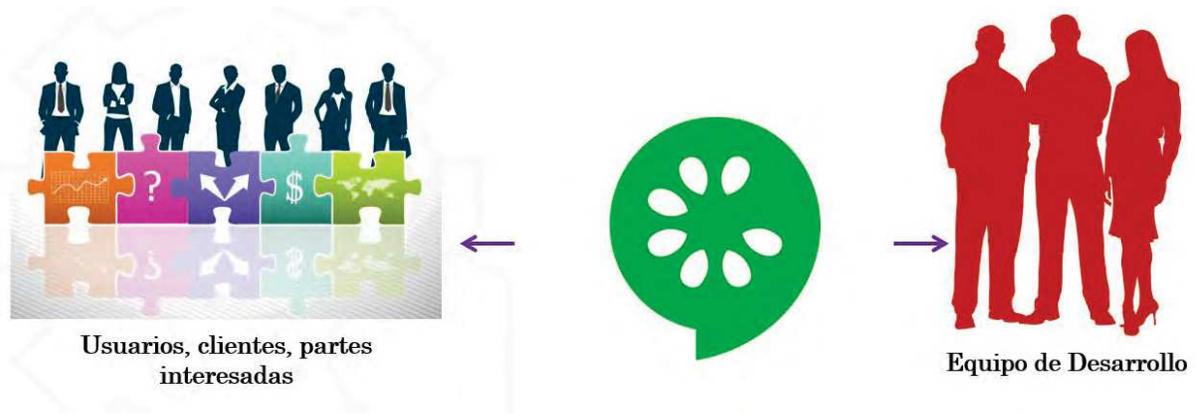


Figura 6. Cucumber entrelaza a los equipos de desarrollo y los usuarios y/o clientes.

Para el proyecto, se almacenan las funcionalidades en los archivos .feature, un ejemplo de implementación de pruebas son las que se muestran a continuación:

Feature:

Verify GET operation using REST-Assured

Scenario: verify one user of the post

Given I perform GET operation for "/posts"

Then I should see userid number as "12892147"

Feature:

Verify DELETE operation using REST-Assured

Scenario: verify DELETE operation after POST

Given I perform POST operation for "/users" with body

name	email	gender	status
Testuser2	test@email.com	Male	Inactive

And I perform DELETE operation for "/users/{id}"

name
Testuser2

And I perform GET operation with parameter for "/users/{id}"

name
Testuser2

Then I shouldn't see the body with name "Testuser2"

La forma en la que estos archivos trabajan, se da creando un archivo al cual se le denomina **Steps Definition** (Definición de paso, por su traducción al español), dentro del cual, mediante expresiones regulares, lee el texto que debe ejecutar la lógica para llevar a cabo nuestras pruebas (Figura 7), utilizemos de ejemplo uno de los pasos de las funcionalidades pasadas, utilizaremos el siguiente: Given I perform GET operation for "/posts", dicho paso se implementa de la siguiente forma:

```
Given("^I perform GET operation for \"([^\"]*)\"$", (String uri) -> {}
```

5.3 Patrón de Diseño Arrange, Act, Assert.

El patrón sugiere dividir una prueba unitaria (un método de pruebas) en tres secciones.

Cada una de las secciones tiene un cometido. Las palabras en inglés a utilizar de estas tres secciones es lo que da nombre al patrón.

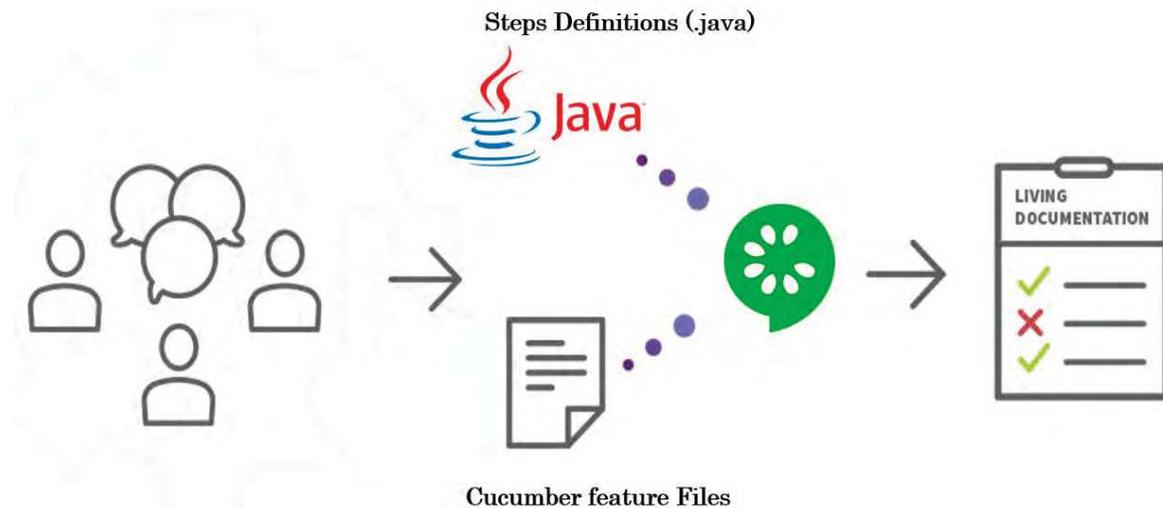


Figura 7. Proceso de interacción de los archivos steps definitivos y features.

Por eso, el patrón AAA se organiza y se implementa de la siguiente forma dentro del proyecto:

Arrange (Organizar/Inicializa): inicializa los objetos y establece los valores de los datos que vamos a utilizar en el Test que lo contiene.

```
RequestSpecBuilder builder = new RequestSpecBuilder();  
builder.setBaseUrl(PropertyManager.getInstance().getProperty("Api_Url"));  
builder.setContentType(MediaType.JSON);  
var requestSpec = builder.build();
```

Act (Actuar): realiza la llamada al método a probar con los parámetros preparados para tal fin.

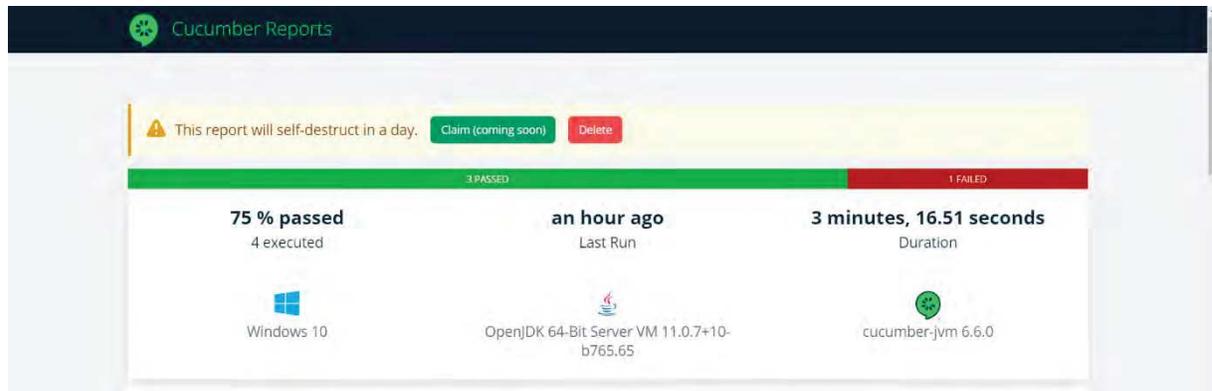
```
return request.get(url);
```

Assert (Confirmar/Comprobar): comprueba que el método de pruebas ejecutado se comporta tal y como teníamos previsto que lo hiciera.

```
assertThat(response.getBody().jsonPath().get("data.user_id"),hasItem(Integer.parse  
Int(userIdNumber)));
```

5.4 Reporte.

Una vez creadas nuestras pruebas e implementado la lógica para ejecutarlas, Cucumber ayuda a generar reportes de estatus (Figura 8), en el cual se indica información del sistema en el cual se encuentren las pruebas, nos muestra el tiempo que se tardó en ejecutar la suite de pruebas y nos muestra la documentación relacionada a cada archivo .feature, en el cual se despliega si el paso ejecutado pasó o falló.



Feature:

Verify DELETE operation using REST-Assured

Scenario: verify DELETE operation after POST

✓ **Given** I perform POST operation for `"/users"` with body

name	email	gender	status
Testuser2	test@email.com	Male	Inactive

✓ **And** I perform DELETE operation for `"/users/{id}"`

name
Testuser2

✓ **And** I perform GET operation with parameter for `"/users/{id}"`

name
Testuser2

✓ **Then** I shouldn't see the body with name `"Testuser2"`

Feature:

Verify GET operation using REST-Assured

Scenario: verify one user of the post

✓ **Given** I perform GET operation for `"/posts"`

✗ **Then** I should see userid number as `"12892147"`

```
java.lang.AssertionError:  
Expected: a collection containing <12892147>  
but: mismatches were: [was <4>, was <5>, was <5>, was <6>, was <8>, was <9>, was <10>, was <11>, was <12>, was <12>, was <14>, was <15>, was  
at org.hamcrest.MatcherAssert.assertThat(MatcherAssert.java:18)  
at org.hamcrest.MatcherAssert.assertThat(MatcherAssert.java:6)  
at com.titanium.steps.GetPostSteps.lambda$new$1(GetPostSteps.java:21)  
at *I should see userid number as "12892147"(file:///C:/Users/gilsa/IdeaProjects/TitaniumApiFramework/src/test/java/com/titanium/feature
```

Figura 8. Reporte de ejecución generado por Cucumber.

6. Conclusiones

La API consta de un conjunto de clases / funciones / procedimientos que representan la capa lógica empresarial. Si la API no se prueba correctamente, puede causar problemas no solo en la aplicación API sino también en la aplicación que realiza la llamada.

En el presente trabajo se mostró un framework de pruebas automatizadas que se puede utilizar para generar casos de prueba de velocidad, seguridad, reglas de negocio, etc. El cual tiene ventajas como, la implementación de Cucumber y Gherkin, en la que se reduce la brecha entre las personas involucradas en el negocio y los desarrollares y simplificar la forma de crear pruebas usando REST Assured. Se puede descargar desde: <https://github.com/gsanchezm/TitaniumApiFramework> y se pueden agregar o eliminar características dependiendo del proyecto en el que se vaya a implementar. Se toma en cuenta el reporte que se genera automáticamente y que se provee en línea, si embargo se



pretende implementar librerías para que los repostes puedan ser generados de forma local. El framework se puede ejecutar en ambientes de integración continua para que se ejecute de forma automática antes de enviar el código en producción.

Referencias

- [1] D. Jacobson, G. Brail, and D. Woods, *“APIs: A Strategy Guide”*. O’Reilly Media, Inc., 2011.
- [2] L. Richardson, M. Amundsen, and S. Ruby, *“RESTful Web APIs”*. O’Reilly Media, Inc., 2013
- [3] D. Jacobson and S. Narayanan, *“Netflix api: Top 10 lessons learned”* in Open Source Convention (OSCON), Portland, Oregon, July 2014. [Online]. Disponible: <http://www.slideshare.net/danieljacobson/top-10-lessons-learned-from-the-netflix-api-oscon-2014>
- [4] R. T. Fielding, *“Architectural styles and the design of networkbased software architectures”* Ph.D. dissertation, 2000.
- [5] *“ProgrammableWeb API Directory,”* accedido en Septiembre 2020. [Online]. Disponible: <http://www.programmableweb.com/>.
- [6] E. T. Barr, M. Harman, P. McMinn, M. Shahbaz, and S. Yoo, *“The oracle problem in software testing: A survey”* Software Engineering, IEEE Transactions on, vol. 41, no. 5, pp. 507–525, May 2015.
- [7] H. Liu, F.-C. Kuo, D. Towey, and T. Y. Chen, *“How effectively does metamorphic testing alleviate the oracle problem?”* Software Engineering, IEEE Transactions on, vol. 40, no. 1, pp. 4–22, Jan 2014.
- [8] E. J. Weyuker, *“On testing non-testable programs,”* The Computer Journal, vol. 25, no. 4, pp. 465–470, 1982.
- [9] *“Spotify Web API”* accedido en Septiembre 2020. [Online]. Disponible: <https://developer.spotify.com/web-api/>.
- [10] S. Segura, G. Fraser, A. Sanchez, and A. Ruiz-Cortes, *“A survey on metamorphic testing”* IEEE Transactions on Software Engineering, vol. 42, no. 9, pp. 805–824, Sept 2016.
- [11] M. Masse, *“REST API Design Rulebook”*. O’Reilly Media, 2011. [Online]. Disponible: <http://books.google.ch/books?id=eABpzyTcJNIC>.
- [12] L. Richardson and S. *“Ruby, Restful Web Services”*, 1st ed. O’Reilly, 2007.
- [13] S. Allamaraju, *“RESTful Web Services Cookbook”*. O’Reilly, 2010.
- [14] *“REST Assured”* accedido en Septiembre 2020. [Online]. Disponible: <https://rest-assured.io/>.
- [15] Geeks.ms, *“Unit Testing y el Patrón AAA”* [online]. Disponible: <https://geeks.ms/jorge/2018/08/25/unit-testing-y-el-patron-aaa/>.
- [16] A. Pascal, *“The Ultimate Guide to BDD Test Automation Frameworks”*, April 2019. [Online]: Disponible: <https://cucumber.io/blog/bdd/the-ultimate-guide-to-bdd-test-automation-framework/>.
- [17] L. Iazar, P. Simona, Basil. (2010). *“Behaviour-Driven Development of Foundational UML Components”*. Electr. Notes Theor. Comput. Sci. 264. 91-105. 10.1016/j.entcs.2010.07.007.
- [18] Brooks, Jr, Frederick. (1987). *“No Silver Bullet Essence and Accidents of Software Engineering”*. IEEE Computer. 20. 10-19. 10.1109/MC.1987.1663532.
- [19] J. Webber, S. Parastatidis, and I. Robinson, *“REST in Practice: Hypermedia and Systems Architecture”*, 1st ed. O’Reilly Media, Inc., 2010.
- [20] *“GO REST”* accedido en Septiembre 2020. [Online]. Disponible: <https://gorest.co.in/>
- [21] *“Guru99”* accedido en Septiembre 2020. [Online]. Disponible: <https://www.guru99.com/api-testing.html#8>
- [22] accedido en Septiembre 2020. [Online]. Disponible: <https://www.soapui.org/learn/functional-testing/scenario-based-testing/>



Seguimiento de trayectorias de un robot móvil diferencial a través del sistema operativo robótico ROS

Salazar-Hidalgo Eduardo¹, Castañeda-Camacho Josefina¹, Martínez-Torres Cesar²,
Martínez-Carranza José³

¹Benemérita Universidad Autónoma de Puebla

²Universidad de las Américas Puebla

³Instituto Nacional de Astrofísica, Óptica y Electrónica

Contacto: eduardo.salazarh@alumno.buap.mx

Resumen

En el presente trabajo se presenta el diseño de un sistema de control lineal para el seguimiento de trayectorias de un robot móvil diferencial haciendo uso del sistema operativo robótico ROS en este caso se trabajó con la versión Kinetic Kame. Básicamente el procedimiento se divide en tres etapas: La primera de ellas consiste en la generación de una secuencia de estados deseados, los cuales permitan guiar al robot hacia un punto arbitrario, tomando en consideración restricciones dinámicas y físicas del vehículo. La segunda etapa se encarga de estimar los estados actuales del robot con la ayuda de los sensores a bordo. Por último, en la etapa de control se evalúan los estados actuales del vehículo con respecto a los estados deseados y en base al error se toman decisiones para llevar a cabo acciones correctivas. El controlador propuesto consiste en un control lineal que hace uso del modelo dinámico para calcular las velocidades lineal y angular las cuales se consideran como entradas de control. Las pruebas experimentales se llevan a cabo sobre el robot móvil diferencial Turtlebot3 Waffle PI.

Palabras clave: robot móvil diferencial, control lineal, seguimiento de trayectorias, ROS, Gazebo, Turtlebot3 Waffle PI

1. Introducción

En los últimos años ha habido un aumento considerable en los trabajos publicados relacionados con robots móviles. Los robots móviles han sido exitosamente empleados en la industria, como robots de servicio, en tareas domésticas, exploración de áreas de difícil acceso o peligrosas para el ser humano, educación, así como en el sector del entretenimiento [1], [2], [3]. Especialmente los robots móviles diferenciales despiertan el interés de la comunidad científica debido a su estructura simple. La estructura de este tipo de vehículos consta de un par de ruedas actuadas fijadas al chasis principal, en combinación con una o dos ruedas pasivas.

Uno de los objetivos de la robótica consiste en dotar a robots de habilidades para la ejecución de tareas de manera autónoma. Actualmente existe un enfoque que busca crear robots capaces de efectuar navegación autónoma. Ésta es un área extensa y generalmente se divide en dos partes: Planeación de trayectorias y control. La etapa de planeación se enfoca en la generación de una trayectoria libre de colisiones para conducir al robot de un punto A hacia un punto B. El algoritmo de control es responsable de mantener al robot lo más cercano posible a la trayectoria previamente generada al calcular la velocidad lineal y velocidad angular adecuadas. El algoritmo de control depende directamente de los errores de estado por lo que resulta crucial conocer los estados actuales del robot, para ello existen diferentes métodos de localización tales como: odometría, SLAM o de sistemas de captura de movimiento.

Como se mencionó anteriormente la navegación autónoma depende de un conjunto de subsistemas los cuales deben trabajar de manera coordinada para lograr un objetivo común. Por tal motivo es necesaria una plataforma lo suficientemente robusta para el desarrollo e implementación de dichos algoritmos. ROS es un entorno consolidado en donde resulta relativamente sencillo implementar algoritmos para la generación de trayectorias, estimación de estados y control. Además, este sistema cuenta con robustos simuladores que permiten evaluar dichos algoritmos sin la necesidad de contar con una plataforma real. Recientemente ha habido trabajos en los que se hace uso de este sistema operativo para llevar a cabo el control de robots móviles diferenciales [4], [5].

El resto del trabajo se organiza de la siguiente manera: En la sección 2 se presenta el modelo dinámico del robot el cual está basado en el modelo cinemático. La plataforma de experimentación se describe detalladamente en la sección 3. La sección 4 describe el procedimiento llevado a cabo para el diseño del sistema de control. Los resultados experimentales obtenidos se presentan en la sección 5. Finalmente, las conclusiones se añaden en la sección 6.

2. Modelo dinámico de un robot móvil diferencial

Existen distintas formulaciones para la obtención del modelo dinámico de un robot móvil diferencial, fundamentalmente se pueden encontrar las que emplean las ecuaciones de movimiento de Euler-Lagrange (balance de energías) [6], [7] y las que parten del modelo cinemático para obtener modelos dinámicos basados en errores [1], [8], [9].

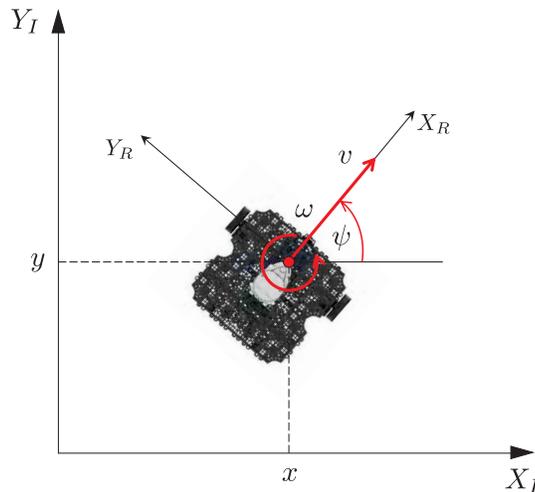


Figura 1. Sistemas de referencia global y local de un robot móvil diferencial.

La pose del robot consta de dos componentes para la posición (x, y) , y una componente para la orientación en ψ la cual indica el giro del robot alrededor del eje z .

$$\mathbf{q} = \begin{bmatrix} x \\ y \\ \psi \end{bmatrix} \quad (1)$$

Estas variables se encuentran definidas con respecto al sistema de coordenadas global. Para el caso de la orientación, el ángulo del vehículo resulta de la desviación existente entre el eje X_R del sistema de coordenadas del robot con respecto al eje X_I del sistema de coordenadas global, tomando como ángulos positivos los que siguen el sentido contrario del giro de las manecillas del reloj. El modelo cinemático de un robot con restricciones holonómicas como el de la Figura 1 está representado por las siguientes ecuaciones diferenciales [1], [8], [9]:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos \psi & 0 \\ \sin \psi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2)$$

Donde v y ω representan a la velocidad lineal y velocidad angular respectivamente, consideradas como entradas de control.

El problema del seguimiento de trayectorias consiste en hacer que el robot siga una trayectoria de referencia, la cual generalmente es conocida de antemano. La trayectoria debe ser generada considerando límites en velocidades y aceleraciones, así como las restricciones físicas del robot y del entorno. Es común que la etapa de generación de trayectorias sea efectuada de forma independiente por otro algoritmo o en tiempo real para un robot autónomo.

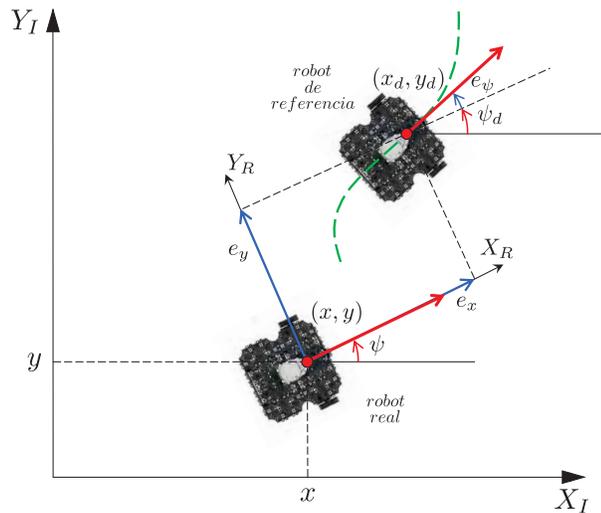


Figura 2. Seguimiento de trayectorias.

Básicamente el robot debe seguir la pose de referencia tal como se muestra en la Figura 2. El control debe ser capaz de mantener al robot lo más cercano posible a la referencia haciendo que los errores de estado tiendan a cero cuando el tiempo tienda a infinito.

3. Plataforma de experimentación

3.1 ROS

ROS (Robot Operating System) o sistema operativo robótico es un entorno para el desarrollo de software para robots que provee la funcionalidad de un sistema operativo en un clúster heterogéneo. ROS es considerado más bien como un meta-sistema operativo ya que es instalado sobre un sistema operativo ya existente. ROS se desarrolló originalmente en 2007 bajo el nombre de switchyard por el laboratorio de Inteligencia Artificial de Stanford. ROS provee los servicios estándar de un sistema operativo tales como abstracción del hardware, control de dispositivos de bajo nivel, implementación de funcionalidad de uso común, paso de mensajes entre procesos y mantenimiento de paquetes. Está basado en una arquitectura de grafos donde el procesamiento toma lugar en los nodos que pueden recibir, mandar y multiplexar mensajes de sensores, control, estados, planificaciones y actuadores, entre otros.

ROS está orientado para el sistema operativo Ubuntu Linux, a pesar de que existen adaptaciones para otros S.O. no está garantizada su estabilidad. Debido a su éxito, distintas instituciones y compañías han comenzado a adaptar sus productos para ser usados en este entorno. Existe una larga lista de



robots compatibles, mismos que se pueden encontrar en [10]. En dicho sitio se pueden encontrar una gran variedad de robots los cuales se subdividen en 4 categorías principales: Aéreos, Terrestres, Marinos y Manipuladores.

Existen varias herramientas para facilitar la programación de robots en ROS entre ellas se encuentra *Gazebo* el cual es un simulador el cual permite programar robots como si estos se trataran de plataformas reales y *Rviz* en donde se pueden visualizar los datos del conjunto de sensores que incorpora un robot.

Recientemente se lanzó una nueva versión de ROS denominada ROS2 la cual resulta ser la versión mejorada de su predecesora con varios cambios significativos.

3.2 Turtlebot3 Waffle PI

TurtleBot3 es una plataforma de hardware estándar de ROS enfocado para su uso en las áreas de investigación y educación (Figura 3). Este robot fue concebido para facilitar la programación de robots. El éxito de esta plataforma se debe principalmente a su tamaño, y a su relativo bajo costo. El objetivo principal del proyecto TurtleBot3 consiste básicamente en reducir el tamaño y precio de la plataforma sin sacrificar la calidad y prestaciones que ofrecían sus predecesores TurtleBot1 y Turtlebot2. Actualmente existen tres modelos del TurtleBot3: Burger, Waffle (descontinuado) y Waffle PI. En este trabajo se hará uso del modelo Waffle PI por lo que a continuación se presenta una descripción de sus características principales [11], [13].

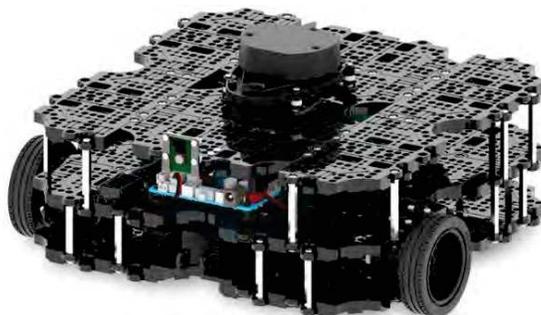


Figura 3. TurtleBot3 Waffle PI (tomada de [11]).

3.2.1 Componentes

Las partes principales que integran al robot TurtleBot3 Waffle PI consisten en una Raspberry PI cuyo modelo puede ser el B ó B+. Este robot incorpora un par de sensores que le permiten ejecutar tareas de localización, mapeo y navegación. El primero se trata de un sensor de distancia laser LIDAR modelo LDS-01. El segundo sensor se trata de una cámara la cual se conecta directamente a la Raspberry PI. Además, este robot cuenta con la tarjeta open CR 1.0, la cual sirve como interfaz en el caso de que se deseara añadir más sensores al robot, incluso ésta misma tarjeta puede ser usada como un Arduino Uno debido a que cuenta con los pines de expansión característicos de dicha tarjeta. El movimiento del robot se lleva a cabo gracias a la incorporación de un par de motores Dynamixel modelo XM430.

3.2.2 Especificaciones

En la Tabla 1 se enlistan las características más importantes del robot Turtlebot3 Waffle PI, más detalles sobre el robot se pueden encontrar en su página oficial [11].

Tabla 1. Características del robot TurtleBot3 Waffle PI.

Características	Valor
Máxima velocidad traslacional	0.26 m/seg
Máxima velocidad rotacional	1.82 rad/seg
Máxima aceleración traslacional	2.5 m/seg ²
Máxima aceleración rotacional	3.2 rad/seg ²
Carga máxima	30 kg
Tiempo de operación	2 horas
Tiempo de recarga	2.5 horas
Alimentación	12V DC, 5 A

3.2.3 Estructura mecánica

La estructura principal del robot se compone de dos ruedas acopladas a la parte frontal del chasis principal y de un par de ruedas pasivas en la parte trasera. Por la configuración anterior, se considera que este modelo pertenece al grupo de robots móviles diferenciales. Las dimensiones del robot son de aproximadamente 30.6cm x 28.1cm x 14.1cm. El peso total del robot es de alrededor de 1.8 kg. Debido a la naturaleza *Open Source* del proyecto Turtlebot3, los diseños CAD de la estructura se encuentran disponibles, lo cual permite a los diseñadores acceder a dichos archivos para su descarga¹, incluso para hacer modificaciones propias.

3.2.4 Configuración del robot para su uso con ROS

Para efectuar este procedimiento, es necesario disponer de una PC remota desde la cual se efectuó el control del robot. El proceso consiste en instalar una distribución de Linux en ambas computadoras, a continuación, instalar ROS y finalmente establecer comunicación entre ambos dispositivos. Para acceder a la computadora del robot es necesario el empleo de comandos *ssh* desde la PC remota. La configuración empleada se ilustra en la Figura 4.

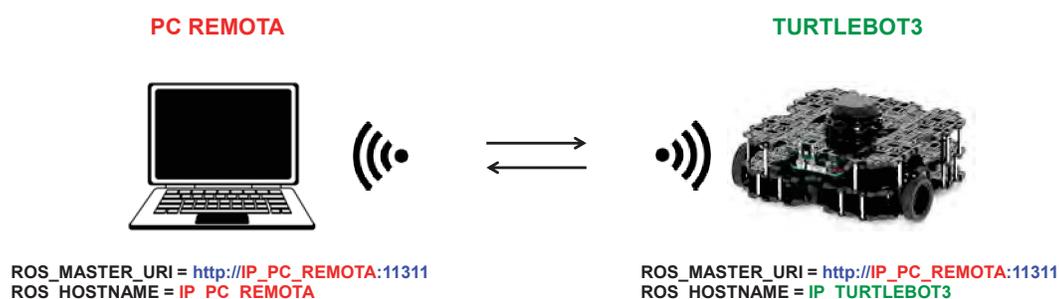


Figura 4. Configuración de red.

Una vez establecida la comunicación lo único que resta es ejecutar el *driver*² del robot (el conjunto de librerías encargadas de controlar los dispositivos de bajo nivel) desde la PC remota. Lo anterior abrirá paso a la transmisión de información sobre el estado de los sensores, de entre los más importantes se encuentran: el estado de la batería, la unidad de medición inercial IMU, el sensor laser,

¹ <http://www.robotis.com/service/download.php?no=678>

² <https://github.com/ROBOTIS-GIT/turtlebot3.git>



así como un sistema que se encarga de calcular la odometría del vehículo (posiciones y velocidades). El mismo *driver* es capaz de recibir comandos de velocidad lineal v y velocidad angular ω para el pilotaje del robot.

4. Desarrollo

El procedimiento para efectuar el control del robot TurtleBot3 Waffle PI se divide en tres etapas las cuales se describen detalladamente en las siguientes subsecciones.

4.1 Generación de estados deseados

El propósito de un generador de estados deseados consiste en calcular una secuencia de estados los cuales conduzcan al robot para que éste siga una trayectoria predefinida tomando en consideración las posibles restricciones físicas y límites en los actuadores [12].

En este trabajo se optó por dividir a la trayectoria completa en segmentos más pequeños, generar los estados deseados en cada segmento y al final concatenarlos para obtener la trayectoria completa. El procedimiento consiste en calcular la distancia entre dos puntos del plano 2D, a continuación almacenar los estados intermedios en un mensaje del tipo *nav_msgs::Odometry* para su posterior publicación.

Los estados calculados se componen de la posición en x , la posición en y , la orientación en ψ , así como las velocidades v y ω . Las velocidades se calculan de tal modo que sigan perfiles trapezoidales o triangulares como los de la Figura 5 con el objetivo de lograr que el robot efectúe transiciones suaves entre segmentos.

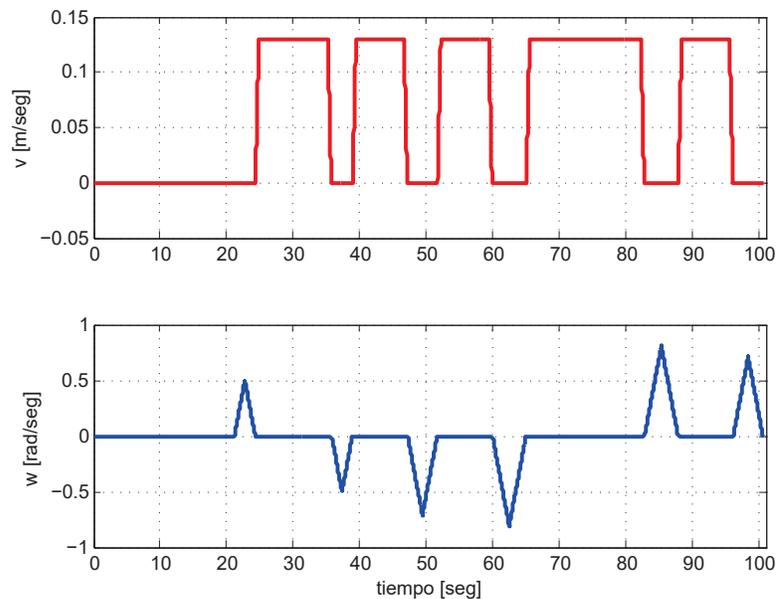


Figura 5. Perfiles de velocidad.

4.2 Estimación de estados del robot

Los estados del Turtlebot3 Waffle PI se obtienen directamente del *driver*. El nodo principal del robot incorpora un sistema que calcula la odometría del vehículo a partir de la información proporcionada por los encoders y de la unidad de medición inercial IMU. La información se publica en el tópico */odom* por medio de un mensaje *nav_msgs::Odometry*.



En la Figura 6 se muestra el trayecto generado por el robot tras recibir comandos de teleoperación. La trayectoria corresponde a la posición en el espacio 2D a partir de los datos de odometría calculados por el robot.

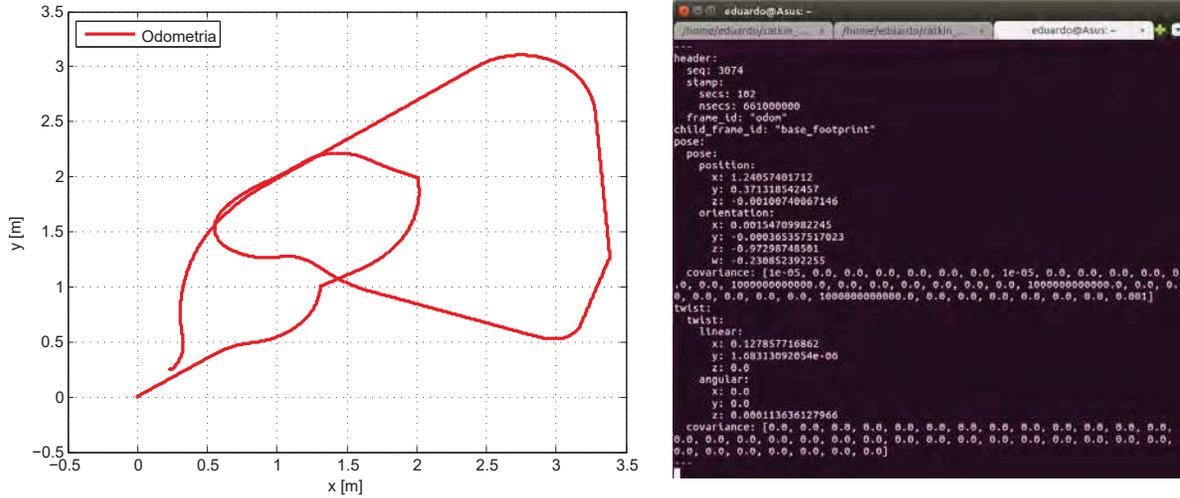


Figura 6. Odometría del Turtlebot3 Waffle PI.

4.3 Control

En esta sección se efectúa el diseño del sistema de control el cual tiene como tarea principal comandar al robot para que éste siga una trayectoria de referencia.

Como primer paso se procede a definir al vector de errores laterales d_{err} , y al error en yaw ψ_{err} relativos a un segmento de la trayectoria. Un ejemplo de un segmento de una trayectoria deseada es una línea que inicia en (x_0, y_0) tangente al vector t .

$$t = \begin{bmatrix} \cos(\psi_d) \\ \sin(\psi_d) \end{bmatrix} \quad (3)$$

Y normal al vector n .

$$n = \begin{bmatrix} -\sin(\psi_d) \\ \cos(\psi_d) \end{bmatrix} \quad (4)$$

A continuación, se calcula el error lateral del robot en la posición (x, y) con ayuda de la siguiente expresión:

$$d_{err} = \left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \right) n \quad (5)$$

El error ψ_{err} se calcula como la diferencia entre el ángulo en yaw actual y el ángulo en yaw deseado.

$$\psi_{err} = \psi - \psi_d \quad (6)$$

De esta forma el vector de errores queda definido de la siguiente manera:

$$e = \begin{bmatrix} d_{err} \\ \psi_{err} \end{bmatrix} \quad (7)$$

El objetivo del algoritmo consiste en calcular las velocidades v y ω necesarias para hacer que el vector de errores tienda a asintóticamente a cero.

De esta manera:

$$v = v_d + K_d e_2 \tag{8}$$

$$\omega = \psi_d + K_\psi e_3 + K_l e_1 \tag{9}$$

Donde K_d , K_l y K_ψ son las ganancias de control. Las ecuaciones (8) y (9) son las encargadas de calcular las velocidades v y ω aplicadas al robot y con ello cerrar el lazo de control tal como se muestra en la Figura 7.

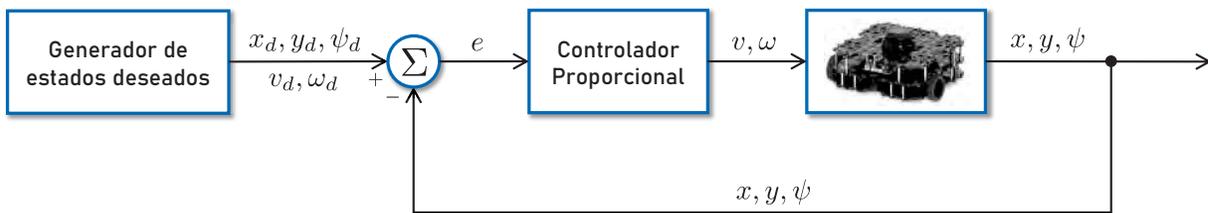


Figura 7. Lazo de control.

El controlador fue evaluado dentro del simulador *Gazebo* (Figura 8), considerando las siguientes ganancias de control (Tabla 2):

Tabla 2. Ganancias de control.

Parámetro	Valor
K_l	1.0
K_d	3.0
K_ψ	10.0

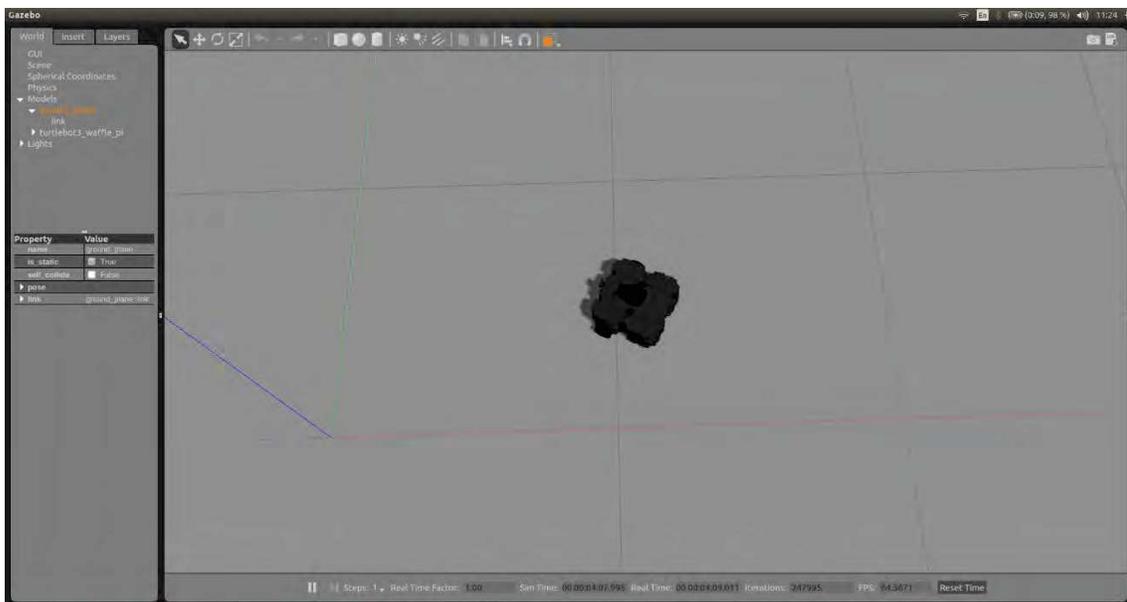


Figura 8. Robot Turtlebot3 Waffle PI dentro del entorno de Gazebo.



Tras ejecutar el programa principal el robot comenzó a seguir la trayectoria de referencia tal y como se esperaba. La Figura 9 se muestran los errores de estado durante el seguimiento de la trayectoria empleando la herramienta de ROS *rqt_plot*.



Figura 9. Errores de estado en *rqt_plot*.

La manera en que los nodos se encuentran interconectados se muestra en la Figura 10. Los nodos están representados por elipses y es donde se lleva a cabo el procesamiento de información. Los tópicos se representan por rectángulos y son los buses por los cuales se transmiten los datos. Por último, las flechas indican la dirección de flujo.

Como se puede observar la información fluye de acuerdo con la estructura del lazo de control de la Figura 7. El nodo */turtlebot_linear_control_node* se suscribe a los datos de odometría provenientes del tópico */odom*, así como los datos de los estados deseados provenientes del tópico */desired_states* para el cálculo las señales de control, las cuales se mandan directamente al robot a través del tópico *cmd_vel*.

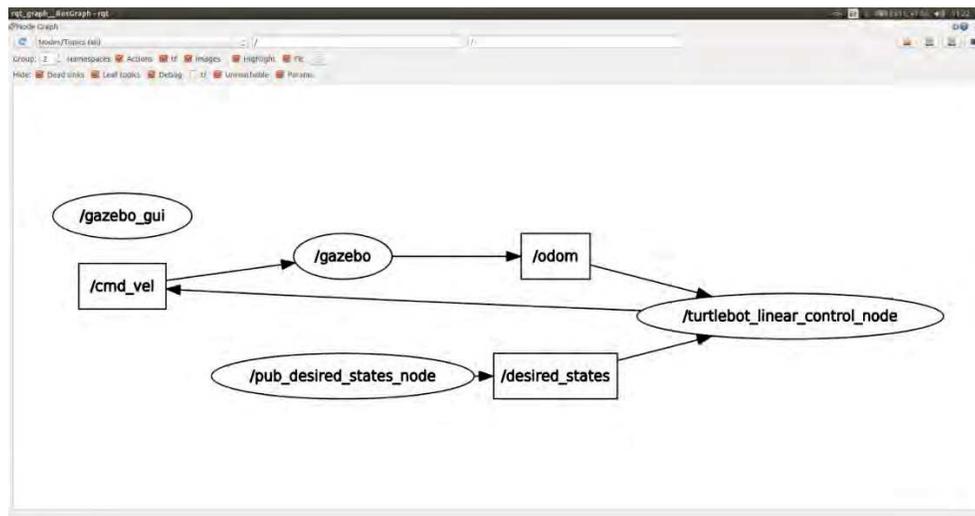


Figura 10. RQT Graph.



5. Resultados

El algoritmo de control fue programado como un nodo dentro de ROS. El programa principal se suscribe a los datos de odometría, así como los estados deseados y con esta información se obtiene el vector de errores. Lo errores de estado se usan en las ecuaciones (8) y (9) para el cálculo de las señales de control.

En la Figura 11 se aprecia la trayectoria trazada por el vehículo. La línea punteada indica la trayectoria deseada y la línea sólida el trayecto seguido por el robot. Como se puede notar, el algoritmo de control cumple de manera eficiente con su tarea al mantener al robot muy cercano a la trayectoria de referencia. Las flechas indican la dirección de avance, comenzando en el origen (0, 0) y finalizando nuevamente en el origen (0, 0).

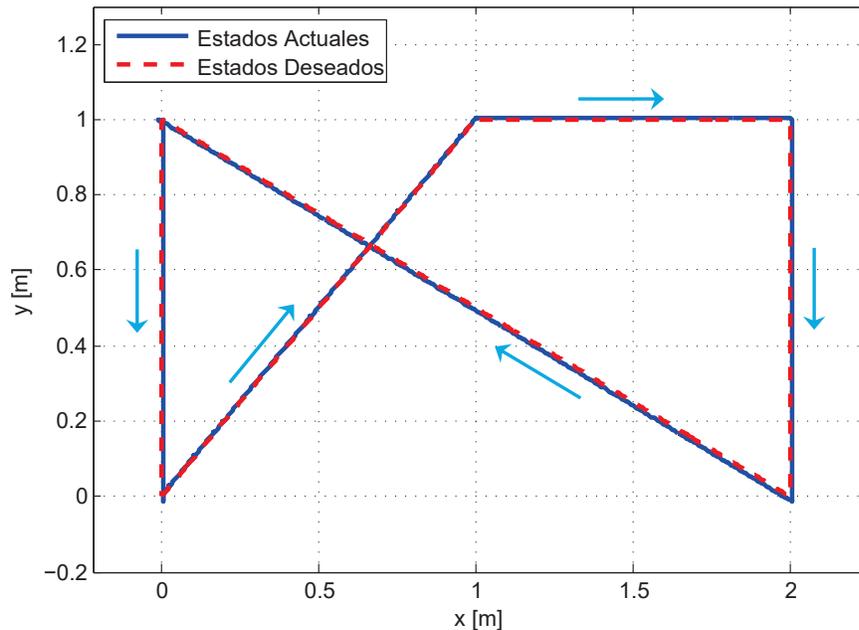


Figura 11. Trayectoria trazada por el robot.

Una forma más clara de observar el desempeño del controlador es a través de los errores de estado. Como se muestra en la Figura 12 la magnitud de los errores es muy pequeña y se mantienen acotados a medida que el tiempo evoluciona.

En la misma figura se pueden notar pequeñas protuberancias cada vez que el robot cambia de posición, sin embargo, al tratarse de sobreimpulsos de menor magnitud no afecta en el comportamiento final del sistema.

La implementación del controlador en la plataforma real y el desempeño de este mismo se puede ver en³.

El comportamiento de las entradas de control al trazar la trayectoria de la Figura 11 se muestra en la Figura 13. La gráfica de la parte superior indica la velocidad lineal del robot. Como se aprecia el perfil de velocidad obtenido es de tipo trapezoidal.

³ <https://youtu.be/gjtTbT0YgIY>

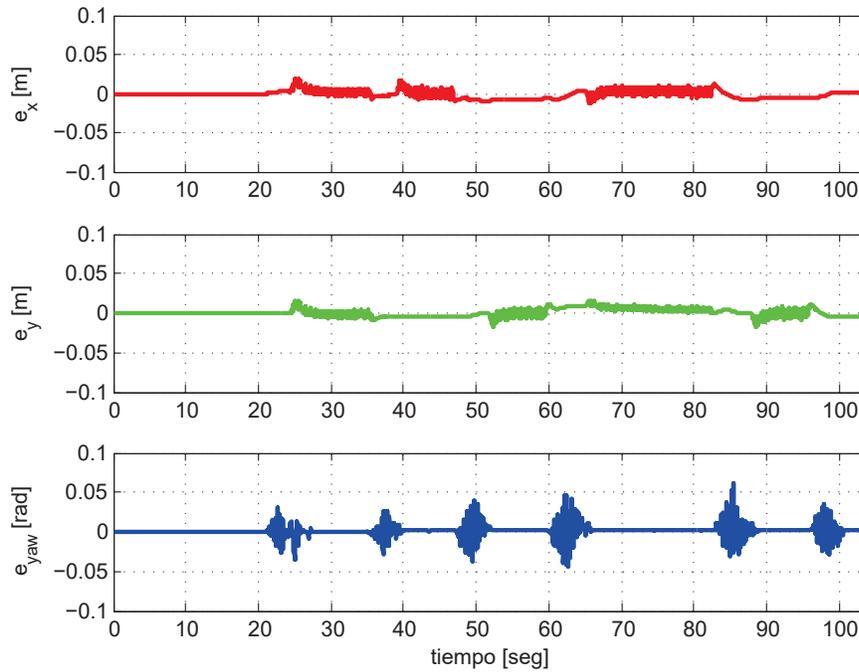


Figura 12. Errores de estado.

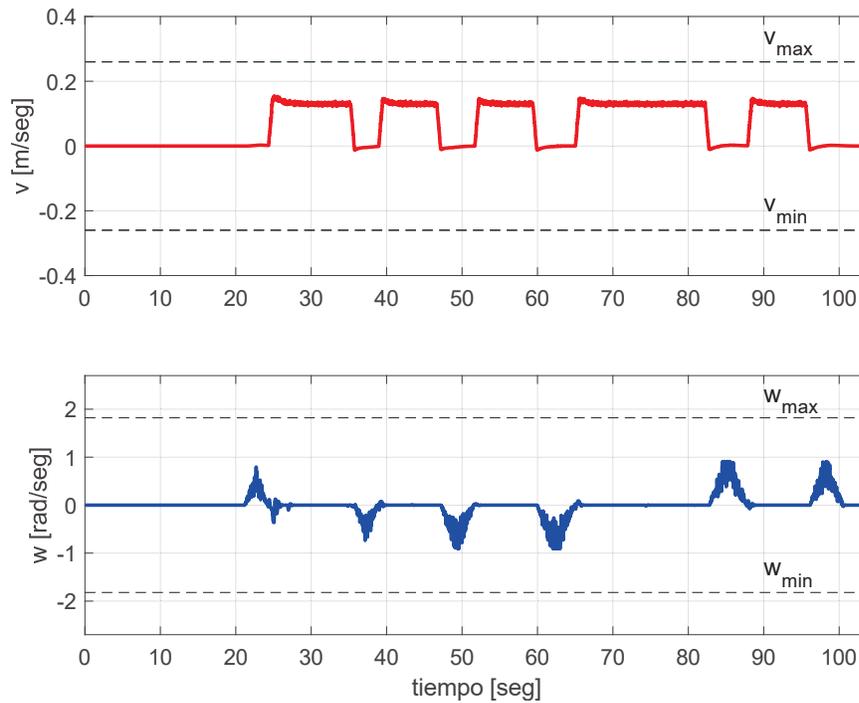


Figura 13. Entradas de control.

En la segunda grafica se muestra la velocidad angular del robot alrededor del eje z. En este caso el perfil de velocidad obtenido es de tipo triangular. Este tipo de perfiles permiten al robot efectuar transiciones suaves entre segmentos con la finalidad de evitar posibles colisiones. El plan consiste en iniciar con una velocidad baja y lentamente acelerar hasta alcanzar la velocidad máxima, cuando se esté a punto de llegar al final del recorrido del segmento nuevamente se reduce la velocidad de manera paulatina hasta llegar al estado de reposo. Lo anterior garantiza transiciones suaves entre segmentos.



Además, en la misma figura se aprecia que los límites en los actuadores (Tabla 1) nunca son violados, por lo que se puede tener la tranquilidad de que los motores nunca alcanzarán el límite de saturación.

6. Conclusiones

Los resultados obtenidos ilustran y validan el desempeño del controlador. El algoritmo es capaz de generar los comandos de velocidad adecuados para que el robot siga una trayectoria deseada. A pesar de que el controlador presentado se trata de un controlador lineal los resultados presentados son satisfactorios y cumplen con la tarea de seguimiento de trayectorias. Además, el algoritmo de control puede ser modificado para implementar leyes de control más robustos en la misma plataforma experimental.

En esta ocasión se hizo uso de los datos de odometría para estimar los estados del robot, sin embargo, se pueden utilizar otros métodos para mejorar la precisión en la estimación o incluso fusionar los datos proporcionados por los distintos sensores.

Como se mencionó en un inicio, la etapa de control solo se trata de una parte en el proceso de navegación autónoma, por lo que como trabajo futuro se puede proponer el diseño de un algoritmo inteligente capaz de generar trayectorias libres de colisiones, para posteriormente integrar ambas partes y así poder evaluar el sistema completo.

Referencias

- [1] Siegwart R. y col. *“Introduction to autonomous mobile robots”*, 2 ed., MIT Press, E.U. 2011.
- [2] Rubio F. y col. *“A review of mobile robots: Concepts, methods, theoretical framework, and applications”*, *International Journal of Advanced Robotic System*, págs. 1-22, 2019.
- [3] Crnokić B. y col. *“Different applications of mobile robots in education”*, *International Journal on Integrating Technology in Education*. Vol. 6, Num. 3, págs. 15-28, 2017.
- [4] Besseghieur K. L. y col. *“Trajectory tracking control for a nonholonomic mobile robot under ROS”*, *Journal of Physics*. págs. 1-5, 2018.
- [5] Bensaci C. y col. *“Nonlinear control of a differential wheeled mobile robot in real time-Turtlebot2”*, *Third International Conference on Technological Advances in Electrical Engineering*, 2019.
- [6] Elsayed M. y col. *“Real time trajectory tracking controller based on Lyapunov function for mobile robot”*, *International Journal of Computer Applications*. Vol. 168, págs. 1-6, 2017.
- [7] Dhaouadi R. y Abu H. A. *“Dynamic modelling of differential-drive mobile robots using Lagrange and Newton-Euler methodologies: A unified framework”*, *Advances in Robotics and Automation*, 2013.
- [8] Kuhne F. y Fetter L. W. *“Model predictive control of a mobile robot using linearization”*, *Proceedings of the IEEE Mechatronics and Robotics*. págs. 525-530, 2004.
- [9] Klančar G. y Skrjanc I. *“Tracking-error model-based predictive control for mobile robots in real time”*, *Robotics and Autonomous Systems*. Vol. 55, Num. 6, págs. 460-469, 2007.
- [10] <https://robots.ros.org/> consulta: 5 oct. 2020.
- [11] <https://emanual.robotis.com/docs/en/platform/turtlebot3/overview/> consulta: 3 de oct. 2020.
- [12] Newman W. S. *“A systematic approach to learning robot programming with ROS”*, CRC Press, E.U. 2018.
- [13] Pyo Y. y col. *“ROS robot programming”*, ROBOTIS, Corea, 2017.



Sistema de capacitación con Realidad Aumentada para el manejo de un torno paralelo modelo G079

Cárdenas López Miguel Martín, Paredes Orta Carlos Alberto,
Valentín Coronado Luis Manuel, Acevedo Ramírez Gustavo Adolfo.

Centro de Investigaciones en Óptica, A.C. Aguascalientes, México.
mmcardenas@cio.mx

Resumen

El presente trabajo muestra el desarrollo de una aplicación bajo la técnica de realidad aumentada implementada para un sistema de capacitación de un torno modelo G079. La importancia de personalizar el proceso de capacitación mediante la tecnología didáctica posibilita adquirir conocimiento de forma interactiva, permitiendo al usuario aprender de manera eficiente, esto debido a que cada usuario puede presentar distintos niveles de conocimiento a causa de su experiencia o nivel educativo, por lo que una plataforma de realidad aumentada coloca a los usuarios en un nivel general al inicio de su aprendizaje, puesto que muestra el contenido de forma visual siendo de mayor comprensión para los usuarios. La aplicación fue desarrollada mediante el software Unity 2018.1.0f2 y se exportó para una tableta Yoga Lenovo con sistema operativo Android, además, se utilizó dos etapas para la detección de marcadores de acuerdo al nivel I de realidad aumentada, en la primera etapa se utilizó el SDK de Vuforia y en la segunda etapa se utilizó el descriptor característico Orb de la librería OpenCV. La aplicación contiene cuatro módulos en realidad aumentada, los cuales son: seguridad, elementos del torno, elementos del taladro y mantenimiento preventivo, además, cuenta con un registro de usuario y una evaluación.

Palabras clave: Realidad aumentada, Capacitación, Unity, Vuforia, Torno paralelo, Sift, Surf, Orb.

1. Introducción

La técnica de realidad aumentada se ha desarrollado como consecuencia de los nuevos avances tecnológicos de la Realidad Virtual, la cual tiene sus inicios en 1973 donde el profesor informático Ivan Edward Sutherland [1] dio un primer esbozo de esta tecnología al crear el casco de realidad aumentada, pero no es hasta 1992 cuando el investigador Tom Caudell creó el término de realidad aumentada [2]. El objetivo principal de esta técnica es mejorar la percepción que tienen las personas sobre su interacción con el entorno, mezclando elementos virtuales con la realidad mediante dispositivos tecnológicos [3], esto permite que la técnica de realidad aumentada pueda ser implementada en el área del aprendizaje o capacitación, debido a que operaciones manuales o actividades que requieran de un conjunto de pasos ordenados, son visualmente mejor comprendidos.

El uso de dispositivos móviles puede ser una alternativa al uso de gafas de realidad aumentada, ya que presentan cada vez más una mayor capacidad de procesamiento, permitido así que la técnica de realidad aumentada sea introducida de manera más accesible en varias áreas, algunos de estos ejemplos son: la formación académica universitaria, como en medicina [4]; el turismo, donde podemos encontrar aplicaciones de instructores virtuales, los cuales permiten orientar a los usuarios y estar disponibles en cualquier momento [5]; en la industria, donde se ha usado como medio de entrenamiento con objetos virtuales [6]. En este último caso, la realidad aumentada permite una mayor familiarización al usuario, debido a que se mejora la percepción de



los operadores con las máquinas herramientas o procesos complejos que lo rodean, favoreciendo así un aprendizaje personalizado que iguala el nivel de conocimiento del personal de nuevo ingreso [7], por ello el presente trabajo muestra el interés de esta tecnología y se buscó desarrollar una aplicación bajo esta técnica aplicada a la capacitación de una máquina modelo, con el objetivo de un futuro implementarlo a nivel industrial o institucional utilizando dispositivos móviles más económico como la tableta Yoga Lenovo del presente proyecto.

La máquina utilizada como dispositivo de aprendizaje para la capacitación en realidad aumentada, fue el torno marca Grizzly modelo G079, ya que su manejo u operación requiere de un conjunto de actividades manuales ordenadas, además, la forma de trabajo de la máquina es el desbaste de material por rotación, lo que implica un alto riesgo para el operador como para los que lo rodean si es utilizada sin un conocimiento previo, un manejo inadecuado o sin el uso del equipo de seguridad necesario, por estas razones el módulo principal de la aplicación se enfoca en la seguridad.

2. Realidad aumentada

2.1 Definición de realidad aumentada

Realidad aumentada es una técnica que combina elementos virtuales generados por computadora colocándolos sobre una imagen del entorno real [9]. El objetivo principal es complementar el mundo real con elementos virtuales permitiendo al usuario interactuar en tiempo real con estos elementos.

2.2 Niveles de la técnica de realidad aumentada

Existe una clasificación de los niveles de realidad aumentada de acuerdo a las tecnologías utilizadas para renderizar y visualizar modelos 3D, en el presente proyecto se considera la clasificación siguiente [8], y se toma en cuenta los elementos del nivel I como base para la realización del proyecto, ya que es la primera opción para implementarla como base para futuros proyectos con los siguientes niveles.

Nivel 0: Las aplicaciones realizan un hiperenlace del mundo físico mediante lectores de códigos de barras.

Nivel I: Las aplicaciones utilizan marcadores como medio de activación, estos son generalmente imágenes en blanco y negro con bordes sobresalientes.

Nivel II: Las aplicaciones no utilizan marcadores, la activación de los modelos se da mediante sistemas de geolocalización o redes inalámbricas.

Nivel III: Las aplicaciones utilizan sistemas más avanzados de visión, geolocalización y redes inalámbricas, implementadas en gafas especializadas.

2.3 Elementos de la realidad aumentada para el nivel I

Los elementos principales para el nivel I son: *un dispositivo de visualización, una cámara digital, la unidad de procesamiento, y el marcador que servirá como elemento de activación.*

El *marcador* sirve como medio de activación para sobreponer el modelo 3D y puede ser utilizada una imagen de cualquier tipo para este propósito, pero, la calidad de la misma radica en los bordes o características que contenga, como el marcador es identificado por medio de las características de la imagen, se consideró realizar el trabajo en dos etapas, en la primera etapa se utilizó Vuforia para la detección de marcadores (Ver sección 4.4.1 para más detalle), y en la segunda

etapa se realizó un análisis de descriptores clásicos implementados en la librería OpenCV, obteniendo como resultado la selección del descriptor Orb, (Ver sección 4.4.2 para más detalle) esto debido al número de puntos característicos que puede generar y a la velocidad con la que los procesa.

Actualmente existen gafas tecnológicas que cuentan con los elementos necesarios, desarrolladas por corporaciones que muestran cada vez más interés por esta tecnología, algunas de ellas son: Google (Google glass), Microsoft (HoloLens), Moverio (BT 2000), entre otras, pero el costo de estas gafas puede ser demasiado elevado, por lo que una alternativa a ellas son los dispositivos móviles, su costo es menor y debido al gran avance que han tenido, presentan cada vez más una mayor capacidad de procesamiento. La Figura 1 muestra la idea general de funcionamiento de una aplicación de realidad aumentada en un dispositivo móvil.

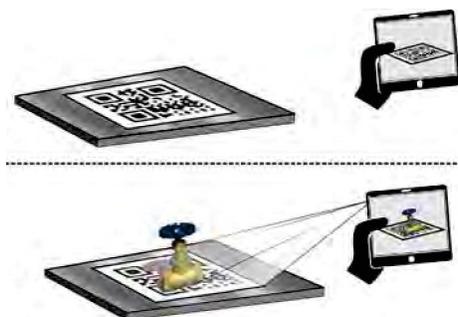


Figura 1. Aplicación en realidad aumentada.

2.4 Aprendizaje móvil

El avance tecnológico que han tenido los dispositivos móviles y la reducción en su precio ha permitido integrarlos de manera eficiente al aprendizaje móvil, la transmisión de conocimiento a través de medios visuales permite a los usuarios adoptar esta metodología fácilmente, sobre todo en la actualidad donde las nuevas generaciones se familiarizan de mejor manera con la tecnología [10]. En la industria, el aprendizaje a través de las capacitaciones es una área donde la técnica de realidad aumentada puede tomar un mayor impacto, la capacitación de los nuevos trabajadores es una necesidad constante de las empresas, por lo que una herramienta bajo la técnica de realidad aumentada puede generar beneficios en tiempo y costo, además, el uso de dispositivos móviles permite una mayor disponibilidad ya que dependiendo de las actividades de aprendizaje, no sería necesario tener siempre a un instructor personal físicamente impartiendo las capacitaciones, pero, para que el sistema sea adoptado completamente tiene que mostrar ventajas para los usuarios, debe ser de fácil manejo y comprensión, ya que de lo contrario se crearía una barrera para el uso de esta tecnología[11].

3. Procesamiento de imágenes con descriptores

3.1 Imagen digital

Una imagen digital de un solo canal puede definirse como una función bidimensional $f(x, y)$, donde las variables x y y representan las coordenadas espaciales en la imagen mostradas por píxeles, y la amplitud de f en cualquier par de coordenadas (x, y) representa la intensidad [12]. Las imágenes pueden mostrar visualmente elementos de distintas formas, tamaños, colores o texturas, por lo que las características para la extracción de información se pueden clasificar en dos grupos [13]:

1. Características globales: Son interpretadas como una propiedad que involucra todos los píxeles, como lo podría ser el color.



2. Características locales: Representa distintivamente la imagen basada en cambios inmediatos en un vecindario de píxeles con elementos sobresalientes.

3.2 Descriptores clásicos

Las características en una imagen se pueden encontrar en una posición en particular, cuando se realiza comparaciones entre imágenes las características permiten identificar a los elementos que lo contienen, pero, cuando uno de estos elementos sufre alguna modificación en su traslación, rotación o escala, las características no son suficientes para comparar imágenes, por lo que es necesario utilizar descriptores, los cuales son vectores característicos que dependiendo el algoritmo con el que son extraídos, definen sus propiedades de cada elemento [14], aunque existen varios, se seleccionaron los descriptores clásicos más utilizados:

Sift: Es un detector y descriptor de puntos característicos locales, al encontrar un punto de interés construye una pirámide de múltiples escalas de la imagen original aplicando el filtro Difference-of-Gaussian(DoG), realiza un muestra de 16 x 16 píxeles alrededor del punto característico y crea un histograma de la orientación de gradientes de la muestra realizando sub regiones de 4 x 4. El descriptor resulta en un vector que contiene los valores de todas las orientaciones y los gradientes dominantes de las subregiones, dando como resultado un vector de 128 elementos para cada punto característico [15].

Surf: Encuentra puntos de interés mediante filtros de un tamaño de 9 x 9 píxeles basado en el determinante de la matriz Hessian, estos filtros se realizan en muestras de 20_s píxeles, donde *s* representa el número en la escala en la que se encontró el punto de interés, posteriormente la muestra es dividida en subregiones de 4 x 4 píxeles, y para cada sub región se aplica la función de Harr wavelet en dirección horizontal y vertical, estas respuestas se ponderan con un filtro gaussiano centrado en el punto de interés para aumentar la robustez frente a deformaciones geométricas y errores de localización, obteniendo así un vector de 64 elementos para cada punto característico[16].

Orb: Es la combinación de dos algoritmos, el detector de puntos característicos FAST (Features from Accelerated Segment Test) y el descriptor BRIEF (Binary Robust Independent Elementary Features), Fast identifica cambios de variación en la intensidad alrededor de un píxel principal en un vecindario circular de 16 píxeles y Brief genera descriptores binarios con los puntos característicos obtenidos por Fast [17].

4. Metodología

4.1 Metodología del trabajo

La metodología mostrada en la Figura 2 fue desarrollada en 5 secciones, en la primera sección fue seleccionado el torno como dispositivo de aprendizaje, debido a que es una máquina que se opera de forma manual, y su uso requiere de un conjunto de actividades ordenadas. En la segunda sección se consideró el manual de operación del torno para tomar las actividades que fueran posibles a realizar en realidad aumentada. En la tercera sección se realizó el diseño del torno utilizando el software SolidWorks y algunas animaciones en Blender. En la cuarta sección para la activación de los modelos 3D a partir de marcadores se utilizó Vuforia y la librería OpenCV. Por último, en la quinta sección fue realizada la interfaz considerando las actividades seleccionadas de la guía de usuario, dando como resultado cuatro módulos principales: *seguridad, elementos del torno, elementos del taladro y por último el mantenimiento.*

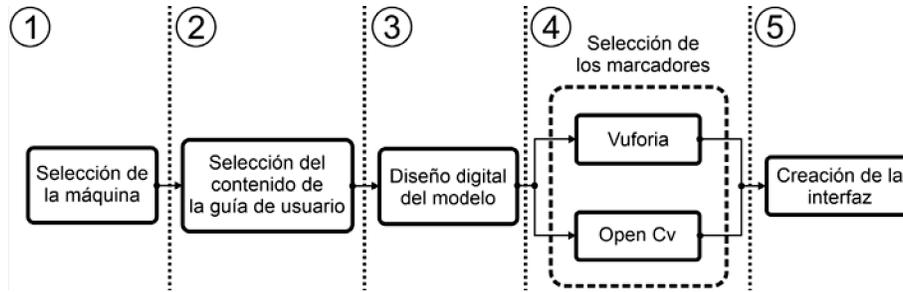


Figura 2. Metodología.

4.2 Metodología del procesamiento de imágenes

La metodología propuesta para el procesamiento de imágenes con la librería OpenCV es mostrada en la Figura 3, ésta presenta los pasos para la comparación de dos imágenes, la primera es la imagen modelo que se pretende identificar como el marcador, y la segunda es la imagen capturada en tiempo real, una vez obtenidas las imágenes son extraídos los puntos característicos y sus descriptores de cada una de ellas mediante el descriptor Orb, posteriormente se realiza la comparación de los descriptores utilizando el algoritmo Brute Force Matcher implementado para descriptores binarios, el cual realiza la comparación basada en la distancia de Hamming [18]. Al obtener la lista de descriptores que presentaron coincidencias entre la imagen modelo y la imagen capturada, se ordenaron y se extrajeron los puntos característicos que pertenecían a dichas coincidencias, para posteriormente calcular la homografía y la transformación de perspectiva.

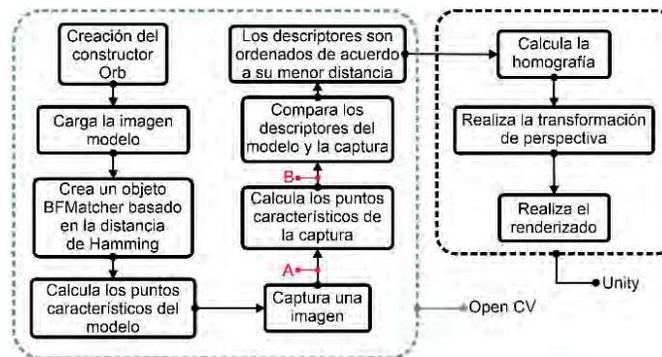


Figura 3. Metodología de procesamiento de imágenes.

4.3 Selección de la guía de usuario

El torno paralelo es una máquina que permite mecanizar piezas metálicas o de madera por medio de un movimiento de rotación denominado movimiento de trabajo, en el cual se hace girar la pieza sobre su propio eje mediante un cabezal, mientras una herramienta de corte realiza un movimiento regulado de avance contra su superficie generando así un desbaste en forma de viruta [19]. Debido al modo de trabajo y a las consideraciones del manual, se tomó en cuenta como módulo primordial la seguridad, de manera que todo usuario debe conocer el equipo necesario, las adecuaciones de la máquina y su entorno para trabajar. El maquinado de piezas cónicas y la elaboración de roscas con el torno son trabajos que no son considerados para la guía, debido a que la realización de estas actividades requiere de un mayor nivel de habilidad, por lo que implica grandes riesgos dejar al usuario en manos de un software sin supervisión. La Figura 4 muestra el contenido seleccionado para la guía de usuario.

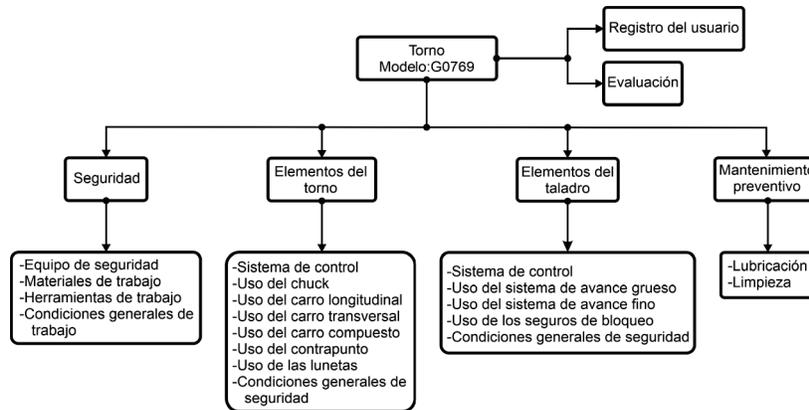


Figura 4. Contenido de la guía de usuario.

4.4 Selección de los marcadores

4.4.1 Marcadores con Vuforia

Los marcadores son uno de los elementos principales de acuerdo al nivel I de realidad aumentada, ya que son utilizados como medio de activación para los modelos 3D, además la mayoría de las aplicaciones de capacitación o mantenimiento utilizan marcadores [20]. Vuforia es una herramienta para el desarrollo de aplicaciones en realidad aumentada, permite descargar una base de datos con las imágenes que servirán como marcadores, Vuforia valora las imágenes a utilizar a través de la asignación de estrellas, las cuales representan su calidad para ser utilizadas como marcadores de realidad aumentada, a mayor asignación de estrellas implica mejor calidad de la imagen como lo muestra la Figura 5, esto permite que el sistema reconozca el marcador con una mayor facilidad y el modelo sea sobrepuesto de forma estable.



Figura 5. Valoración de marcadores por Vuforia.

4.4.2 Marcadores con OpenCV

Para la detección de marcadores con OpenCV, se realizó la comparación de los tres descriptores clásicos mencionados anteriormente (Sift, Surf, Orb), por lo que la primera actividad fue identificar las condiciones de iluminación en las que se encuentra el torno, esto con el objetivo de obtener el parámetro de intensidad con el que la aplicación en realidad aumentada sería empleada, para ello se utilizó un luxómetro calibrado obteniendo los valores de intensidad en varios puntos

cercanos al torno como lo muestra la Figura 6, debido a que existen dos fuentes de iluminación cercanas al torno se obtuvo un intervalo de intensidad de 311 a 359 luxes.

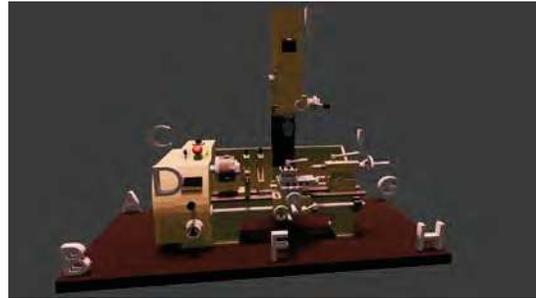


Figura 6. Puntos de medición de intensidad.

Obtenido el parámetro de iluminación se realizó la comparación de los descriptores Sift, Surf y Orb mediante una aplicación realizada en Unity, la aplicación obtiene los puntos característicos de las capturas tomadas por la cámara de 8 Mpx integrada al dispositivo móvil de trabajo, para esto se realizó un marcador con patrones de esquinas cuyas líneas tienen un ancho de 0.1 mm a 2.5 mm como lo muestra la Figura 7, el marcador fue colocado en la parte frontal del torno con el objetivo de tomar varias capturas con distancias de 20 cm a 100 cm con intervalos de 10 cm, y así obtener los puntos característicos.

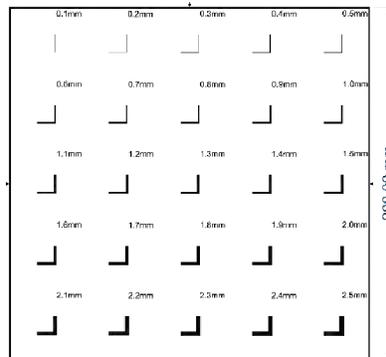


Figura 7. Marcador utilizado como patrón de esquinas.

La Figura 8 muestra una captura tomada con la aplicación a 70 cm de distancia, y los 3 descriptores aplicados a la región de interés de la imagen.

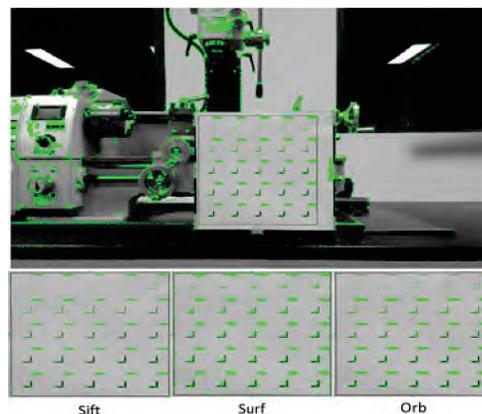


Figura 8. Puntos característicos del marcador patrón.

La Figura 9 muestra el número de puntos característicos obtenidos a las diferentes distancias con los 3 descriptores aplicados a la región de interés.

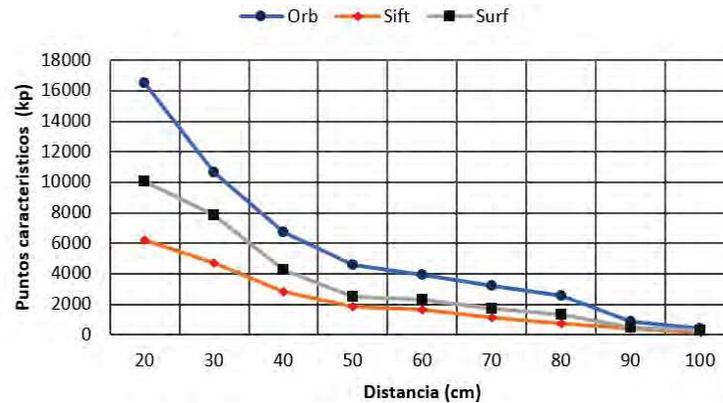


Figura 9. Comparación de puntos característicos.

De igual forma se obtuvieron los tiempos mostrados en la Tabla 1 de los tres descriptores aplicados solo a la captura mostrada en la Figura 10, el tiempo fue tomado entre los puntos A y B mostrados en la Figura 3, ya que solo se tomó en cuenta el tiempo de la extracción de puntos característicos que es lo que provoca la variación del tiempo de ejecución.

Tabla 1. Tiempo de ejecución

Descriptor	Puntos característicos	Tiempo (ms)
Sift	830	1631.27
Surf	674	189.05
Orb	3169	69.33

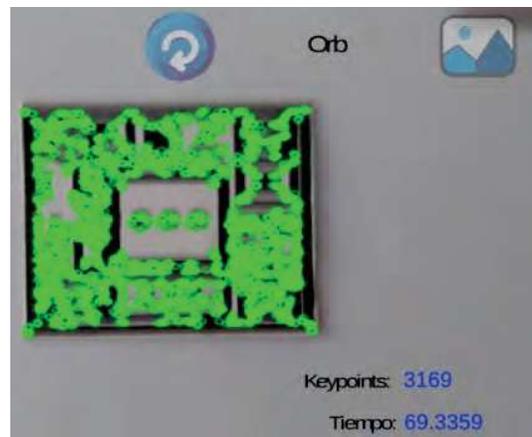


Figura 10. Puntos característicos obtenidos mediante Orb.

Debido a que el descriptor Orb presenta un mayor número de puntos y el consumo de procesamiento muestra para esta aplicación ser 3 veces más rápido que Surf y 23 veces más rápido que Sift [17], es considerado el algoritmo ideal para la aplicación, ya que se usará en un dispositivo móvil.

5. Resultados

5.1 Diseño de los marcadores

Una vez definido el algoritmo a utilizar (Orb), bajo las condiciones de iluminación (311 a 359 luxes) y a la distancia máxima tomada (100cm), se consideró seleccionar las líneas para los marcadores de 1.1 mm hasta 2.5 mm como válidas para la detección, las 3 filas que están dentro de las líneas punteadas en color azul de la Figura 11 representa dicho intervalo, por último, se definió un tamaño final del marcador de 5 x 5 cm buscando que no ocupara mucho espacio al colocarlo sobre las superficies correspondiente en el torno, de igual manera la Figura 11 muestra el diseño final de los marcadores, los cuales cuentan con la numeración correspondiente a cada módulo en la aplicación.

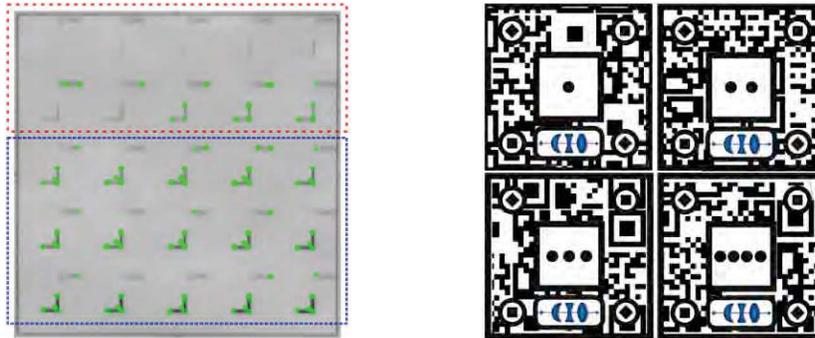


Figura 11. Selección de los marcadores.

5.2 Diseño de los modelos 3D

Se realizó el diseño 3D del torno mostrado en la Figura 12 empleando el software SolidWorks, el modelo contiene las partes principales que realizarán alguna animación o tendrán interacción con el usuario como lo son: *el taladro vertical, el contrapunto, el cabezal, los carros de desplazamiento y sistema de control.*

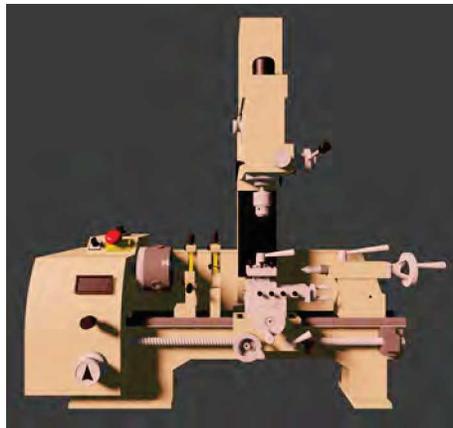


Figura 12. Modelo virtual del torno.

El modelo del personaje que realizará la interacción con el usuario como instructor virtual es mostrado en la Figura 13, el cual fue desarrollado en el software Blender, de igual manera las animaciones del modelo se realizaron mediante la técnica de animación por keyframes, la cual

consiste en mover de posición al modelo considerando una línea de tiempo determinada por el número de fotogramas por segundo.

La resolución en un modelo 3D determina su calidad de detalle visual, a mayor resolución el modelo se podrá observar más definido, pero al mismo tiempo, el tamaño del archivo se incrementará [21], por lo que se buscó que la mayoría de las piezas del torno y el modelo del instructor fueran polígonos regulares de baja resolución, ya que la aplicación está contemplada para dispositivos móviles de gama media y se pretende que el costo computacional para el renderizado sea menor.

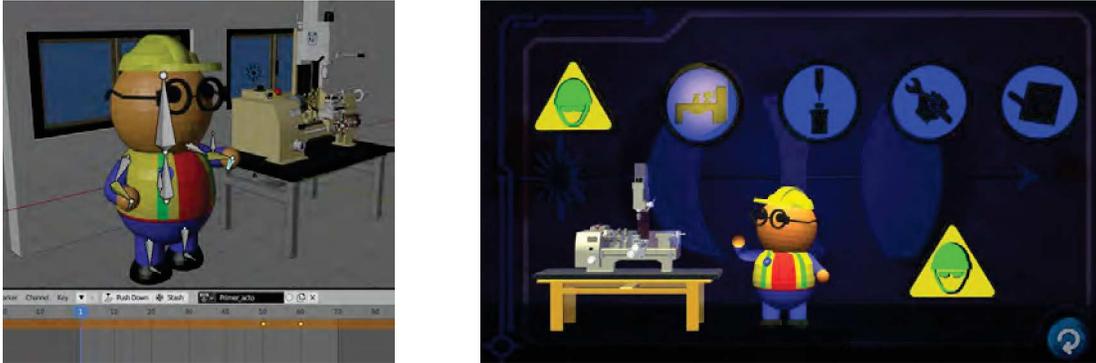


Figura 13. Instructor virtual.

El instructor virtual reproduce un audio describiendo lo que se observa en realidad aumentada de cada módulo, esto permite que el usuario tenga una mayor interacción con la aplicación y no solo lea las instrucciones, el audio fue generado mediante el sintetizador de voz Balbolka, el cuales un software gratuito que convierte el texto en audio. Dependiendo el módulo en el que se encuentre el usuario, tendrá que buscar con la cámara los marcadores indicados, ya que estos son propios de cada módulo y la visualización de los modelos 3D no será activada, los marcadores tienen un orden consecutivo el cual se indica en la pantalla de cada escena como lo muestra la Figura 14.



Figura 14. Pantalla inicial del módulo del torno.

5.3 Creación de la aplicación de realidad aumentada

La Figura 15 presenta el escenario principal de la aplicación creada en Unity, donde un menú desplegable muestra los módulos o escenas de la aplicación, en cada módulo se da una breve introducción y se indican los marcadores que se deben ubicar con la cámara como se mostró en la Figura 1.



Figura 15. Menú principal del sistema.

El módulo de registro mostrado en la Figura 16 permite al usuario registrarse, crear una cuenta y realizar una evaluación, para ser registrado se debe ingresar un nombre de usuario con un máximo de 20 caracteres y una contraseña con un mínimo de 7 caracteres la cual puede contener números y letras. El nombre de usuario se puede repetir debido a que se asigna un número de identificación consecutivo, pero las contraseñas no pueden ser las mismas.

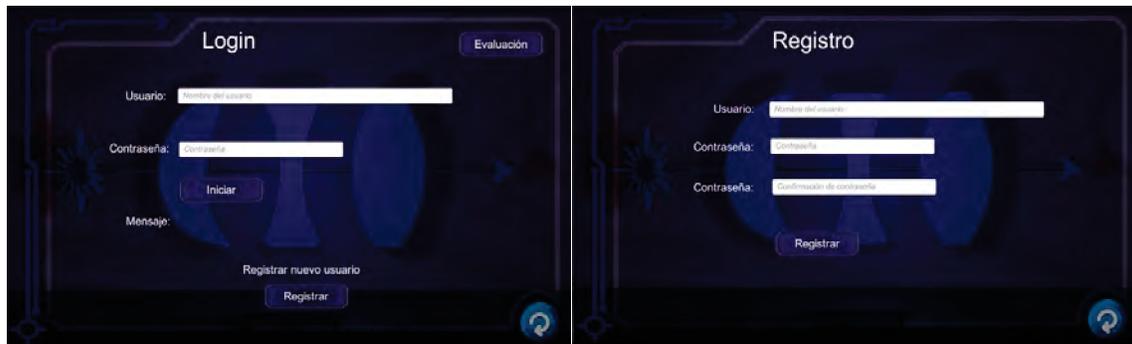


Figura 16. Inicio de sesión y registro de usuario.

El módulo de evaluación consta de 10 preguntas con respuestas de opción múltiple, y solo pueden acceder a ella los usuarios registrados, el objetivo de la evaluación es obtener evidencia de que el usuario adquirió el conocimiento básico mostrado en la guía de usuario en realidad aumentada, los resultados son presentados en una tabla como muestra la Figura 17.

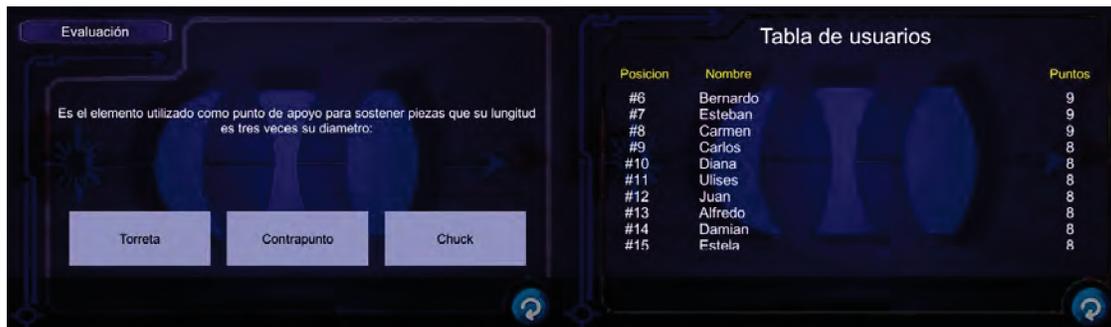


Figura 17. Módulo de evaluación.

Uno de los elementos del torno es el contrapunto mostrado en la Figura 18, debido a que por el momento no se tiene acceso a él, se realizó un modelo similar en papel ilustración con las mismas dimensiones que el real, y se colocó el marcador en su posición correspondiente.

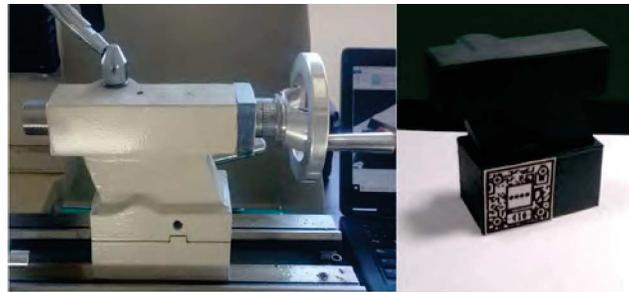


Figura 18. Contrapunto.

La Figura 19 muestra la animación del modelo 3D en realidad aumentada para el contrapunto, en ella el instructor virtual por medio de audio describe por pasos las actividades que son necesarias para su uso, de igual forma se muestra los elementos que contiene y el instructor describe su funcionamiento, siendo éste el resultado de la etapa I descrita en la introducción.

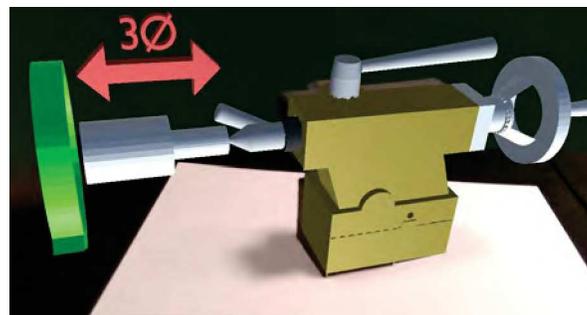


Figura 19. Contrapunto en realidad aumentada con Vuforia.

La Figura 20 muestra los puntos característicos del modelo realizado en papel mediante la aplicación implementada con OpenCV, y de igual forma se muestra la animación del modelo 3D utilizando la metodología de la Figura 3, siendo esto el resultado de la etapa II descrita en la introducción. El objetivo de utilizar OpenCV es porque es una librería de código abierto, por lo que es posible en un futuro, realizar bajo algunas consideraciones aplicaciones en realidad aumentada con código propio para la industria.

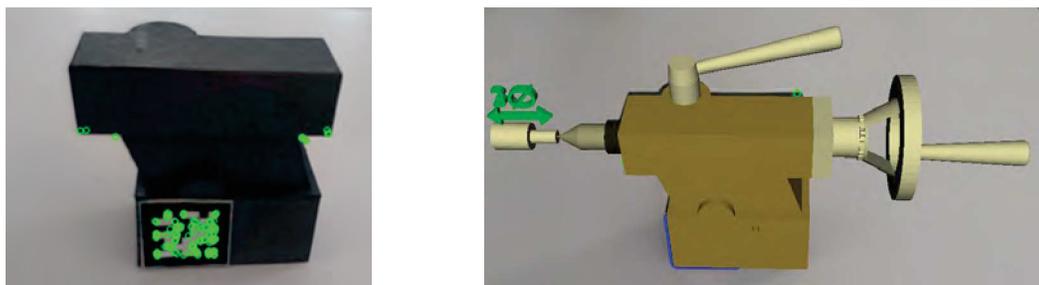


Figura 20. Contrapunto en realidad aumentada con OpenCV.

6. Conclusiones

El trabajo realizado muestra una interfaz funcional con cuatro módulos principales y un registro de usuario instalada en un dispositivo móvil con sistema operativo Android, concluyendo así la etapa I, la interfaz sirve como guía rápida para la capacitación del torno marca Grizzly consiguiendo mejorar



el aprendizaje del mismo mediante medios tecnológicos, permitiendo igualmente ser la base de futuros trabajos en la etapa II. Se generaron los marcadores apropiados para la aplicación a utilizar bajo las condiciones mencionadas en los resultados. La velocidad para el procesamiento del descriptor seleccionado, mostró mantener el efecto de realidad aumentada, es decir, el modelo renderizado se puede percibir en tiempo real. Se comprobó que en este caso el descriptor Orb es el ideal bajo las condiciones mencionadas en los resultados para esta aplicación.

7. Trabajo futuro

El trabajo realizado forma la base de futuros proyectos en realidad aumentada con algoritmos propios, ya que existen ciertas mejoras que se encontraron en el desarrollo del mismo. La iluminación es uno de los elementos que mayor afecta la detección de marcadores, por lo que se pretende buscar una metodología que contribuya a reducir este efecto. La renderización del modelo 3D a partir del descriptor, presenta cierta inestabilidad cambiando repentinamente su posición, por lo que este efecto, es una opción de mejora para la etapa II. Se planea en una etapa futura realizar la valoración de la aplicación con los usuarios, ya que debido a las condiciones de contingencia no se lograron realizar.

Referencias

- [1] Billinghurst M, Clark A, Lee G, et al. *A survey of augmented reality*. Foundations and Trends in Human-Computer Interaction, 8(2-3):73–272, 2015.
- [2] Caudell T, Mizell D. *Augmented reality: An application of heads-up display technology to manual manufacturing processes*. In Hawaii International Conference on System Sciences, pages 659–669, 1992.
- [3] Azuma R, Bailiot Y, Behringer R, Feiner S, Julier S, and MacIntyre B. *Recent advances in augmented reality*. IEEE computer graphics and applications, 21(6):34–47, 2001.
- [4] Cabero J, Barroso J, and Obrador M. *Realidad aumentada aplicada a la enseñanza de la medicina*. Educación médica, 18(3):203–208, 2017.
- [5] Yovcheva Z, Buhalis D, and Gatzidis C. *Smartphone augmented reality applications for tourism*. E-review of tourism research (ertr), 10(2):63–66, 2012.
- [6] Lee K. *Augmented reality in education and training*. TechTrends, 56(2):13–21, 2012.
- [7] Jetter J, Eimecke J, and Rese A. *Augmented reality tools for industrial applications: What are potential key performance indicators and who benefits*. Computers in Human Behavior, 87:18–33, 2018.
- [8] Blázquez A. *Realidad aumentada en educación*, España, 2017.
- [9] Azuma R. *A survey of augmented reality*. Presence: Teleoperators & Virtual Environments, 6(4):355–385, 1997.
- [10] Fabregat R. *Combinando la realidad aumentada con las plataformas de e-learning adaptativas*, Enl@ce Revista Venezolana de Información, Tecnología y Conocimiento, 9 (2), 69-78,2012.
- [11] Aklesh B, and Sailesh K. *Mobile learning adoption: A systematic review*. Education and Information Technologies, 24(1):471–487, 2019.
- [12] Gonzalez R, Woods E, and Eddins S. *Digital image processing using MATLAB*. Pearson Education, India, 2004.
- [13] Hassaballah M, Abdelmgeid A, and Alshazly H. *Image Features Detection, Description and Matching*. Springer, Cham.2016
- [14] Fan B, Wang Z, Wu F, et al. *Local image descriptor: modern approaches*, volume 108. Springer, 2015.
- [15] Lowe D. *Object recognition from local scale invariant features*. In Proceedings of the seventh IEEE international conference on computer vision, volume 2, pages 1150–1157. Ieee, 1999.
- [16] Bay H, Tuytelaars T, and Van Li. *Surf: Speeded up robust features*. In European conference on computer vision, pages 404–417. Springer, 2006.
- [17] Rublee E, Rabaud V, Konolige K, and Bradski G. *Orb: An efficient alternative to sift or surf*. In 2011 International conference on computer vision, pages 2564–2571. Ieee, 2011.



- [18] Jakubović A, and Velagić J. *Image feature matching and object detection using brute-force matchers*. In 2018 International Symposium ELMAR, pages 83–86. IEEE, 2018.
- [19] Gerling H. *Alrededor de las máquinas herramienta*. Reverté, México, 2000.
- [20] Palmarini R, Erkoyuncu J, Roy R, and Torabmostaedi H. *A systematic review of augmented reality applications in maintenance*. Robotics and Computer-Integrated Manufacturing, 49:215–228, 2018.
- [21] Bryden D. *CAD and rapid prototyping for product design*. Laurence King Publ., 2014.



Monitoreo a distancia de intensidad de luz basado en fotodiodo para la adquisición de datos en tiempo real

Blas Hernández Marycruz, García Valdez Chayanne

Facultad de Ingeniería, Universidad Autónoma de Querétaro, Querétaro, Qro.
marycruzblas91@gmail.com

Resumen

En la actualidad el monitoreo a distancia es una realidad que hace unos años se creía fuera del alcance tecnológico. Actualmente es una herramienta de gran utilidad cuando se tienen fenómenos físicos del interés de algún investigador o especialista que se encuentra físicamente separado en el espacio y/o el tiempo del fenómeno de investigación. El propósito de este trabajo es presentar el desarrollo y aplicación realizados en software de código abierto donde se obtuvo y proceso una señal de salida digital que se tomó del Fotodiodo OPC101 el cual es alimentado por 20 VCD, teniendo una señal análoga que captura la intensidad de la luz ambiental y artificial, esta señal se mandó a la entrada digital de una placa de desarrollo de hardware que servirá solo como interfaz por el puerto serial a 9600 baudios, se conectó una tarjeta Raspberry Pi 3 B+ donde se visualizan e implementan los códigos para procesar esta señal. Con ayuda de Python y Base de Datos (DB) se envió a Amazon Web Services (AWS), de este modo se crea el acceso para monitorear en tiempo real los datos adquiridos digitalmente. Se pretende que el prototipo funcione en un futuro para el monitoreo de fenómenos físicos en la agricultura.

Palabras clave: Monitoreo a distancia, fotodiodo, aplicación de sensor, intensidad de luz, Python, base de datos, AWS, tiempo real, adquisición de datos, IoT.

1. Introducción

La vida a la que el ser humano estaba acostumbrado en los últimos años tuvo un cambio repentino, perjudicando a distintos sectores de la población. Dado a esto las nuevas tecnologías conocidas hasta ahora son una herramienta de gran utilidad para el distanciamiento. Algunas herramientas para la convivencia, el manejo de información, la muestra de noticias instantáneas o datos de interés particular que se requieren en tiempo real se han estado manejando desde hace ya algunos años atrás. Según [1] actualmente, varias tecnologías disponibles posibilitan la modernización de un sistema, que van desde el uso de soluciones embebidas con alto poder de procesamiento, de software conectados, hasta la utilización de plataformas de Tecnologías de la Información (TI) para el almacenamiento, análisis y administración de los datos en la nube. El uso de dispositivos como tablets o smartphones, sumado a la transmisión de datos vía Bluetooth o WiFi, han facilitado de manera sustancial el manejo de la información y los tiempos de respuesta.

De acuerdo con [2] al ver un crecimiento sorprendente durante los primeros años del siglo XXI, con respecto al uso y creación de nuevas computadoras, se requiere una gran cantidad de expertos que conozcan los lenguajes de programación modernos, para que la computadora:

- Ejecute aplicaciones
- Realice cálculos matemáticos complicados

Por esta razón es indispensable estar a la vanguardia en todos aquellos avances tecnológicos y actualización de herramientas que ayuden a dar un paso más para seguir a paso firme estas nuevas



tecnologías. Una de las nuevas herramientas creadas hace un par de décadas atrás fue Python que es un lenguaje de scripting de plataforma independiente y orientada a objetos ofreciendo ventajas como rapidez de desarrollo de proyectos y mayores velocidades de ejecución.

La tecnología que está prosperando en nuestros días es el Internet de las cosas por sus siglas en inglés, *Internet of things* (IoT) e industria 4.0 esto involucra a los sistemas embebidos que son en la actualidad mayores facilitadores para generar nueva tecnología y son usadas como herramientas de gran apoyo para los ingenieros e investigadores que progresan en las ciencias y la tecnología.

En este artículo se da a conocer el desarrollo un sistema de monitoreo con el sensor OPT101 *Monolithic Photodiode* adquiriendo datos y enviándolos a través de un servidor AWS para visualizar el monitoreo en una página web en tiempo real de intensidad de luz desde una PC o un smartphone. Este proyecto da a conocer parte de la aplicación que se puede desarrollar e implementar con sistemas embebidos para el monitoreo de intensidad de luz en tiempo real el cual en un futuro se implementará en el área de la agricultura.

2. Metodología y Desarrollo

La metodología que se utiliza para el desarrollo del trabajo de se muestra en la figura 1.

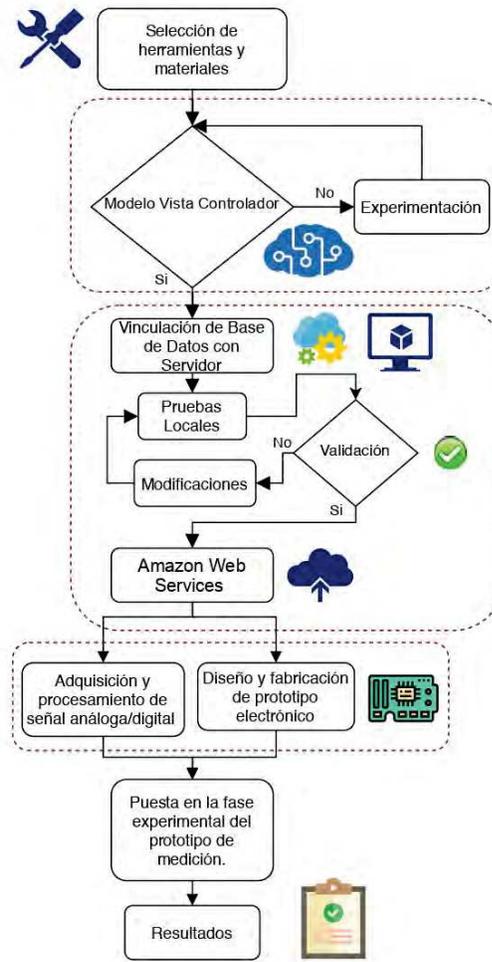


Figura 1. Metodología utilizada para el monitoreo a distancia de intensidad de luz en tiempo real.



La metodología que se emplea en este proyecto consiste en 5 importantes secciones, la primera se basa en la búsqueda de todas aquellas herramientas y materiales adecuados para el desarrollo del proyecto. La segunda sección es crear el Modelo Vista Controlador en donde se estuvo experimentando para poder tener bases sólidas en el manejo de información de una base de datos. El siguiente paso fue vincular la base de datos con el servidor, para ello, se realizó una tercera sección en dos partes, la primera fue la validación con un servidor local y una vez que se tuvo respuesta favorable se vincula en una segunda parte con el AWS. Siguiendo el esquema presentado en la Figura 1 se ejecutan a la par, en una cuarta sección, la adquisición y procesamiento de la señal analógica a digital y el diseño/fabricación del prototipo electrónico. Por último, en una quinta sección se pone puesta en marcha la fase experimental del prototipo de medición dando resultados favorables los cuales se presentan en este artículo en el apartado de resultados.

2.1 Herramientas y materiales

A gran medida se desarrollan lenguajes de programación de mayor factibilidad y accesibilidad un claro ejemplo de ello es Python siendo un lenguaje de scripting independiente de plataforma y orientado a objetos en el cual se puede realizar cualquier tipo de programa, debido a estas grandes ventajas que ofrece Python se colocó como una de las herramientas sobresalientes de este proyecto ya que ofrece una rapidez de desarrollo, la sencillez y velocidad con la que se pueden crear programas y como plus es gratuito.

Una vez establecido el lenguaje de programación se decide trabajar con *Visual Studio Code* (VSC) ya que es un editor de código fuente para Windows, Linux y macOS, esto nos da una gran ventaja para trabajar en dos sistemas totalmente distintos cuando se genera el desarrollo del proyecto, de igual manera es gratuito y de código abierto. De acuerdo con [3] el código combina la interfaz de usuario optimizada de un editor moderno con asistencia y navegación de código enriquecido y una experiencia de depuración integrada, sin la necesidad de un IDE completo.

Otra herramienta indispensable en este proyecto es un software que nos ayude a la creación de un proyecto web de alta calidad, se optó por utilizar Django, según [4] es un *framework web* de alto nivel que permite desarrollar sitios web seguros y mantenibles, es gratuito y de código abierto.

El proyecto se centra en tener una gran cantidad de datos y poder contenerlos en una base de datos por tal motivo se utilizó PostgreSQL 12 como el sistema gestor de base de datos puesto que presenta fácil accesibilidad, es multiplataforma y está disponible para su utilización en casi todos los sistemas operativos.

La última pieza de las herramientas de software utilizadas para este proyecto es una potente plataforma en la nube que nos ofrece más de 175 servicios integrales de centros de datos a nivel global, esta plataforma es AWS. Para entender un poco más lo que AWS nos ofrece tenemos que entender ¿Qué es la informática en la nube?, de acuerdo con [5] menciona que: "La informática en la nube es la distribución de recursos de TI bajo demanda a través de Internet mediante un esquema de pago por uso. En vez de comprar, poseer y mantener servidores y centros de datos físicos, puede obtener acceso a servicios tecnológicos, como capacidad informática, almacenamiento y bases de datos, en función de sus necesidades a través de un proveedor de la nube como *Amazon Web Services* (AWS)." Entendido este concepto se puede definir el modelo DevOps con base en [6] las operaciones de desarrollo dentro de una organización tienen que ser de velocidades altas y se pueden trabajar en todo el ciclo de vida de la aplicación, desde el desarrollo y las pruebas hasta la implementación y las operaciones. Es por estas razones que se elige a AWS como servidor del proyecto. Tanto las herramientas como los materiales a utilizar se mencionan en la Tabla 1.

En la parte electrónica como se muestra en la Tabla 1 en la sección de materiales se utilizó el módulo del sensor OPT101 el cual integra un fotodiodo y un amplificador operacional. Este sensor cuenta con un paquete transparente donde el fotodiodo tiene una superficie de detección más grande con mayor sensibilidad y un amplio rango de respuesta espectral como se muestra en la Figura 2b

(infrarrojo a luz visible 300nm-1000nm) [7]. Las conexiones que se utilizaron en el módulo del sensor están mostradas en la Tabla 2 haciendo referencia a las conexiones vistas en la Figura 2a.

Tabla 1. Materiales y herramientas para el sistema desarrollado.

Característica	Grafico	Herramienta	Material	Pz	Descripción
Python 3.8		X		1	Lenguaje de scripting independiente de plataforma y orientado a objetos.
Visual Studio Code		X		1	Editor de Código fuente.
Django 2.0		X		1	Framework web de alto nivel.
PostgreSQL 12		X		1	Gestor de bases de datos relacional y orientado a objetos.
Amazon Web Services		X		1	Plataforma en la nube.
Módulo de sensor opt101			X	1	Fotosensor de iluminancia.
Arduino UNO			X	1	Placa de desarrollo de hardware de código abierto.
Raspberry Pi 3 Model B+			X	1	Ordenador de bajo costo y tamaño reducido.
Cable USB tipo B a Tipo A			X	1	Cable USB Tipo B a Tipo A Para Arduino UNO.
Adaptador de Corriente Universal Perfect Choice			X	1	Eliminador de donde se obtienen 20 VCD.
Cable HDMI UltraHD 4K de Belkin (2 m)			X	1	Cable HDMI a HDMI.
Monitor I2781FH 27 pulgadas			X	1	Monitor para tener interfaz visual con Raspberry Pi 3 .
Laptop ASUS VivoBook S S510U			X	1	Ordenador con Windows 10 a 64 bits, microprocesador Intel Core i5 8th Generación, tarjeta de video NVIDIA GEFORCE 930 MX y RAM 8GB.

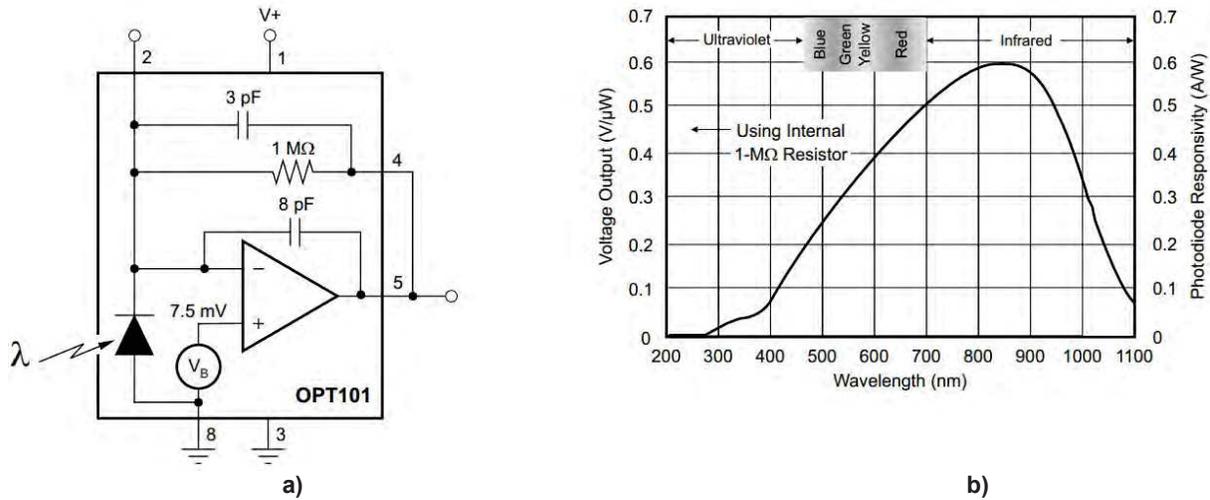


Figura 2. Diagrama de bloques (a) y Respuesta espectral (b) de OPT101, tomada de [5].

Tabla 2. Conexiones de pines del OPT101.

Numero de Pin	1	2	3	4	5	6	7	8
V_s	x							
$-I_n$		x						
$-V$			x					
1Mohms Feedback				x				
Output					x			
Common								x
NC						x	x	

2.1.1 Arduino como herramienta de adquisición de datos análogos

Arduino actualmente es una plataforma de código abierto que facilita las interfaces que nos pueden servir en proyectos diversos, en este caso se utiliza por ser una placa de desarrollo basada en hardware y software libre, flexible y práctica. Como señala [8] el microcontrolador de Arduino posee lo que se llama una interfaz de entrada, que es una conexión en la que podemos conectar en la placa diferentes tipos de periféricos. Como expresa [9] la información de estos periféricos que se trasladara al microcontrolador, el cual se encargará de procesar los datos que le lleguen a través de ellos. También cuenta con una interfaz de salida, que es la que se encarga de llevar la información que se ha procesado en el Arduino a otros periféricos, los cuales son utilizados en este proyecto.

2.1.2 Raspberry PI 3 model B+ como herramienta para procesar datos digitales

Cuando se habla de Raspberry es muy conocido que se hable del concepto de “hacerlo uno mismo”. Como lo hace notar [10] este pequeño ordenador potente de bajo costo y tamaño practico nos da la facilidad de poder crear sistemas de gama alta en el mundo de la tecnología actual. La Raspberry Pi 3 B+ está compuesta por un SoC, CPU, memoria RAM, puertos de entrada y salida de audio y video, conectividad de red, ranura SD, conexiones para periféricos de bajo nivel, etc. Es elegida para este proyecto por todas estas características las cuales con muy factibles para el desarrollo de procesos de datos digitales [11].



2.2 Modelo vista controlador (MVC)

El proyecto en VSC que es el editor de código fuente se denomina “Clorofila” dentro de este se desprende “Cloro” que es donde se encuentran los archivos .py y .html para la creación y el funcionamiento de la página web, mostrado en la Figura 3.

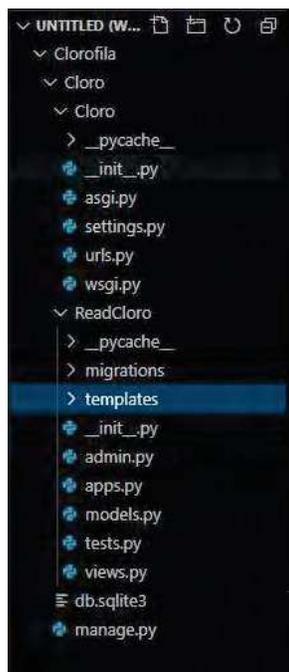


Figura 3. Scripts a utilizados con Python

Con el *framework* de aplicaciones web Django se desarrolló un Modelo Vista Controlador (MVC) con lo cual se crea el servidor y dentro de este se creó una página web la cual mostrara una gráfica de datos almacenados en una base de datos. La Figura 4 indica como es el funcionamiento de la aplicación, en el diagrama en la sección 4 el cliente realiza una petición, dicha petición le llega al controlador, sección 1, desarrollado en Django. El controlador se encarga en pedir y/o mandar información al modelo, sección 2, la información que se encuentre en la base de datos será direccionada al controlador, de allí el controlador procesara dicha información y se mostrará en las vistas (normalmente páginas web), estas vistas mandarán la respuesta grafica al cliente de manera que se pueda interpretar por el usuario.

Las secciones que contiene el diagrama de flujo de la información en MVC, se pueden entender de la siguiente manera:

1. Controlador: En esta parte se encuentran los controles como los *paths* de programación y las urls que son utilizadas para la interconexión entre la Vista y el Modelo.
2. Modelo: Dentro de esta sección esta la base de datos donde se contiene la información adquirida por el fotodiodo/Arduino UNO y procesada por Raspberry Pi 3 B+. “RedCloro” es el modelo creado para la adquisición de datos digitales enviados al controlador.
3. Vista: Aquí podemos encontrar todos aquellos archivos html que sirven para la visualización de gráficos.

4. Cliente: Son los dispositivos como tabletas, *smartphones*, laptops o computadoras de escritorio de donde el usuario puede acceder a la respuesta grafica del MVC.
5. Microprocesador: Envía información procesada al servidor en la nube.

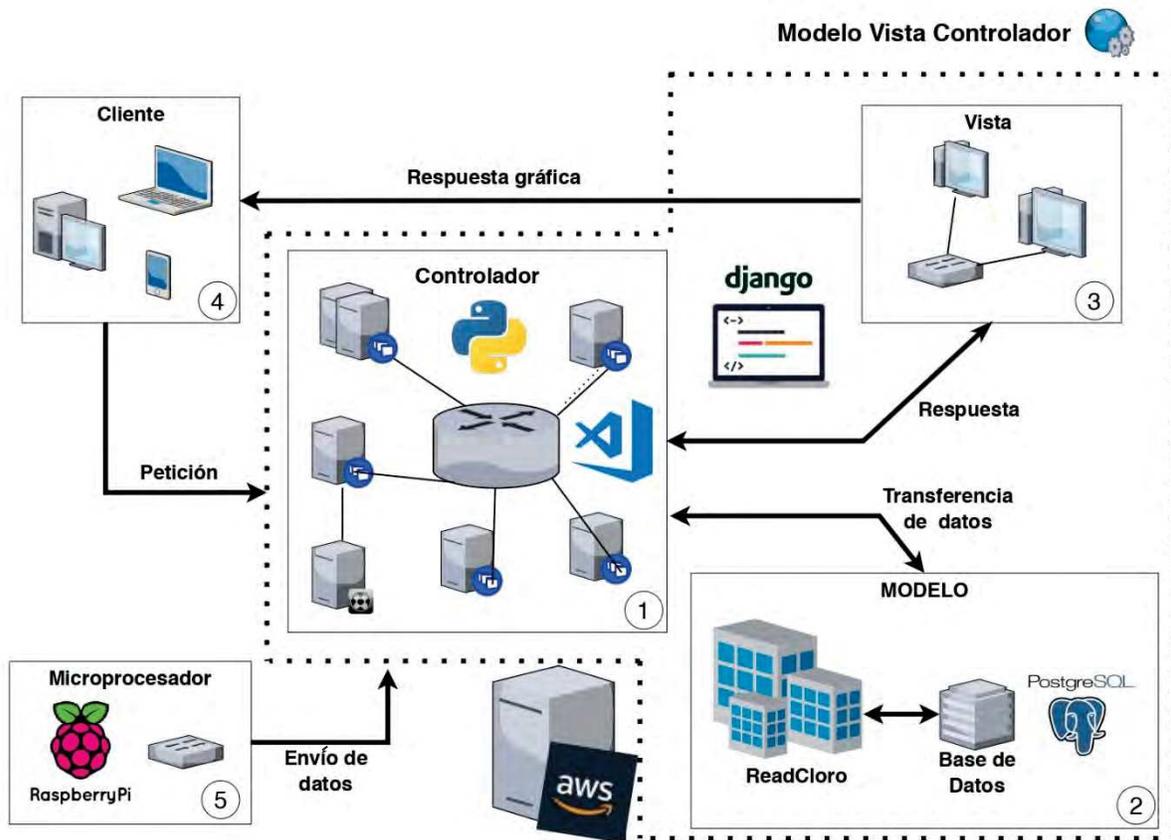


Figura 4. Diagrama de flujo de la información en MVC.

Se crean las bases de la aplicación en un apartado del proyecto que es llamada "ReadCloro" apoyado del lenguaje de programación Python. Los *templates* se dividen en dos partes, una donde se encuentran las plantillas para darle formato a la página web, en donde se utilizaron los archivos .html mostrados en la Figura 5.

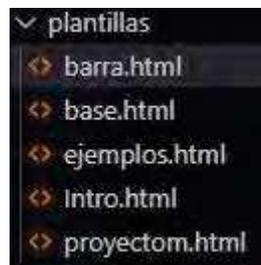


Figura 5. Plantillas utilizando el lenguaje .html

La segunda parte consta de los .html que se utilizaron para poder insertar la información de la base de datos, direccionarlos por medio de botones que muestren estos datos en un gráfico y otro botón de igual manera para poder borrar la información. Se muestran en la Figura 6 los archivos utilizados.



Figura 6. Templates a utilizar con formato html.

2.3 Vinculación de Base de Datos con Servidor

Por último, encontramos el archivo “manage.py” que es el archivo principal del proyecto. Una vez terminado todo el código que se utilizó ligaremos nuestro proyecto con PostGres 12 a un servidor local como se muestra en la Figura 7 donde se realizaron las primeras pruebas de visualización de la página web en la dirección: <http://127.0.0.1:8000/>. Además, podemos observar que los controladores están direccionados en: <http://127.0.0.1:1335/browser/>.

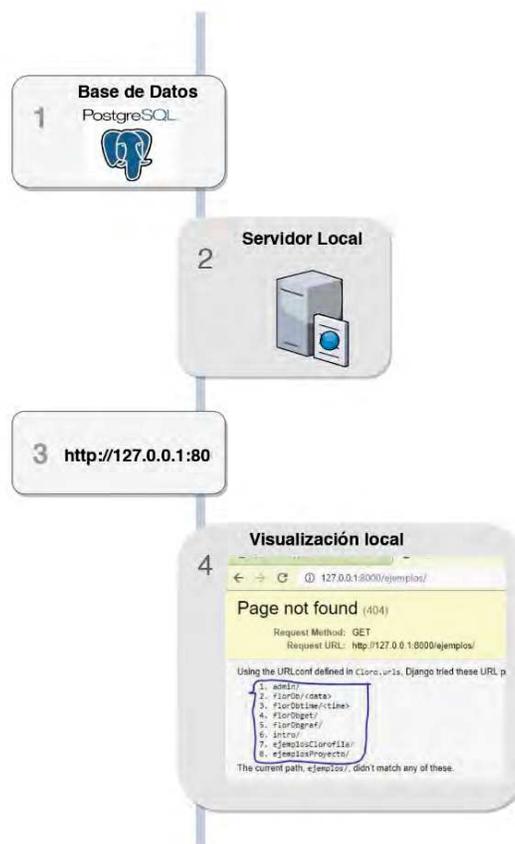


Figura 7. Vinculación de Base de Datos con Servidor Local.

Una vez que se realizan las pruebas locales, dando resultados favorables para el proyecto y se verificó que todo funciona de acuerdo con lo previsto se generan las instancias para poder activar la página del servidor migrando la información de la base de datos a el servidor de Amazon para AWS, véase Figura 8.

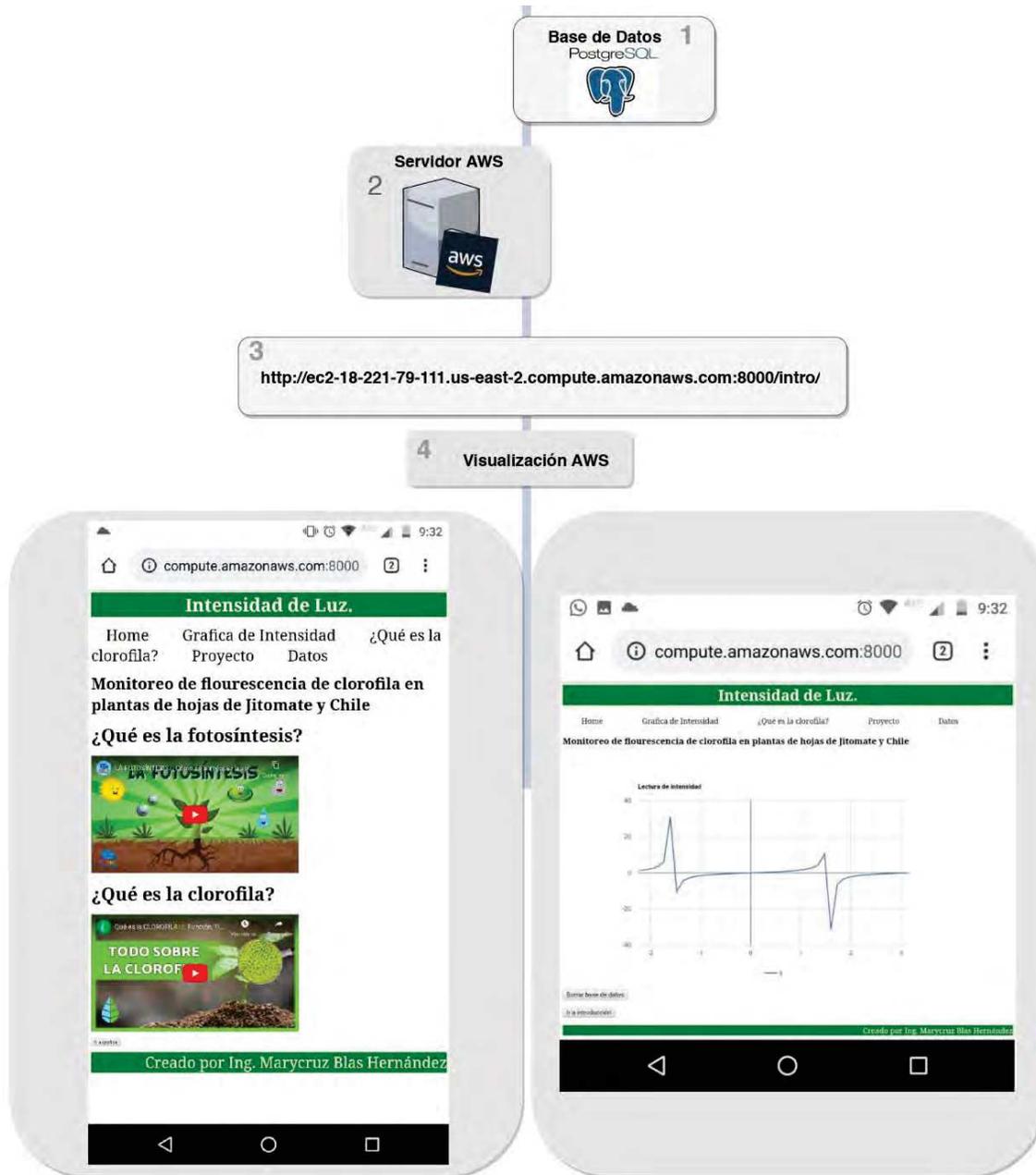


Figura 8. Vinculación de Base de Datos con Servidor de Amazon.

Ahora si podemos ver la página web en el servidor de Amazon. Figura 8 sección 3 en: <http://ec2-18-221-79-111.us-east-2.compute.amazonaws.com:8000/intro/> y en la sección 4 se visualizan los datos de prueba graficados desde un *smartphone*, donde de muestra la página de inicio de la página web y la sección de grafica de datos, tomando en cuenta que todavía no es en tiempo real.

2.4 Adquisición y procesamiento de señal analógica/digital

Ahora que la página web se encuentra en el servidor se procede a realizar esta misma acción en Linux desde el Raspberry Pi OS llamado Raspbian y así adquirir datos para poder monitorear y visualizar en tiempo real.



Para adquirir los datos primero se realizaron las conexiones electrónicas del OPT101 que es un fotosensor que capta la intensidad de luz, este sensor se alimenta de 2.7 a 36 VCD de acuerdo al rango de intensidad que se desea captar, en esta ocasión alimentaremos a 20 VCD el cual da un rango de 0 – 1023 que es la resolución que guarda la información $10 \text{ bits } 2^{10} = 1024$, contando el número 0, en un rango de 0-1023, visto en mV se divide $20 \text{ VCD}/1023$ que nos da 0.019550 mVCD de resolución.

A fin de captar los datos de la señal que nos manda el sensor, se optó por adquirirla por medio de la entrada analógica del Arduino UNO y una vez que era adquirida, convertida digital mandarla por medio del puerto serial a 9600 baudios a la Raspberry para poder procesar los datos. Véase Figura 9 y Figura 15.

Una vez que se realizaron las conexiones electrónicas de los dispositivos utilizados se generan los archivos .py que ayudaran a la adquisición de datos en el entorno de Raspbian, vistos en las Figuras 9 y 10. El archivo Mary_final.pem es el archivo que tiene integrado el proyecto que se generó en VSC pero se puede acceder en el desde el *shell* de Python.

El archivo Prueba5_intensidad.py es el archivo final donde se ejecuta la adquisición de datos. Este se puede visualizar en la Figura 10 en Raspbian y Figura 11 en Windows. En el archivo Intensidad.txt visto en la Figura 9, se guardan los datos obtenidos del puerto serial a 9600 baudios el cual se toma como archivo de referencia para poder procesar los datos en Prueba5 intensidad.py. La Figura 12 presenta una prueba preliminar del procesamiento de datos y finalmente, los datos son enviados al servidor por Prueba5_intensidad.py. por la sintaxis mostrada en la Figura 10.

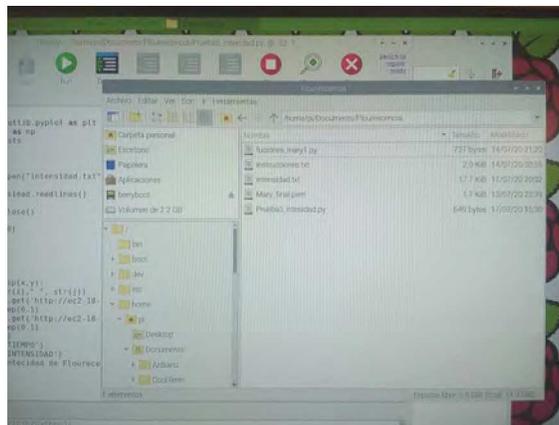


Figura 9. Archivos utilizados en Raspbian para adquirir datos digitales y crear el monitoreo visualizándolos en la página web.

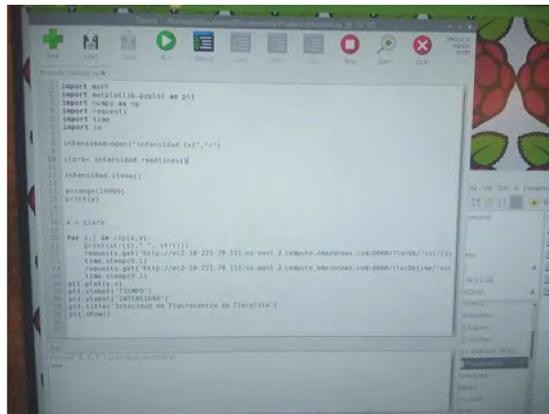


Figura 10. Script Prueba5 intensidad.py. en Raspbian.



```
Prueba5_intensidad.py - C:\Users\Marycruz.Blas.Hdez\Documents\M. en C. Instrumentación y Control Automatico\3 ER SEMESTRE\SISTEMAS EMBE...
File Edit Format Run Options Window Help
import math
import matplotlib.pyplot as plt
import numpy as np
import requests
import time
import io

intensidad=open("intensidad.txt","r")

cloro= intensidad.readlines()

intensidad.close()

y=range(10000)
print(y)

x = cloro

for i,j in zip(x,y):
    print(str(i), " ", str(j))
    requests.get('http://ec2-18-221-79-111.us-east-2.compute.amazonaws.com:8000/FlorDb/'+str(i))
    time.sleep(0.1)
    requests.get('http://ec2-18-221-79-111.us-east-2.compute.amazonaws.com:8000/FlorDbtime/'+str(j))
    time.sleep(0.1)
plt.plot(y,x)
plt.xlabel('TIEMPO')
plt.ylabel('INTENSIDAD')
plt.title('Intensidad de Flourecencia de Clorofila')
plt.show()
Ln: 1 Col: 0
```

Figura 11. Script Prueba5 intensidad.py en Windows y Python.

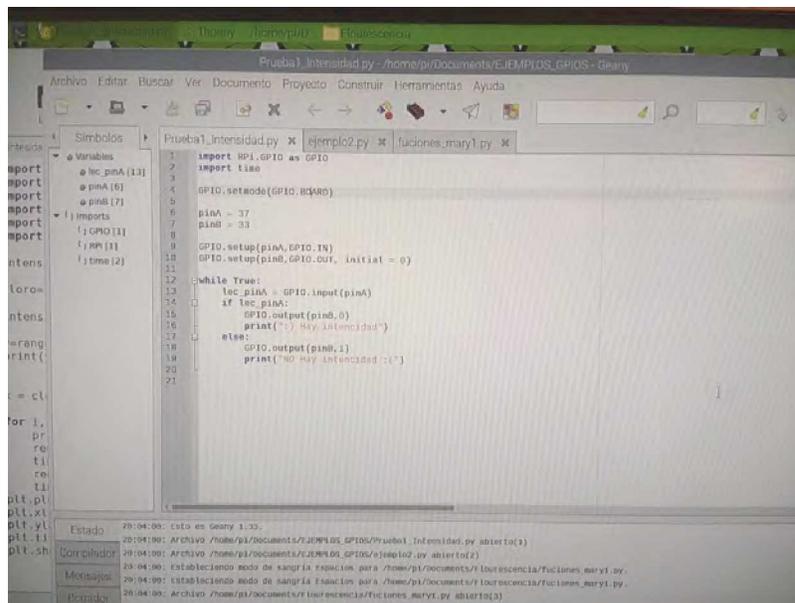


Figura 12. Primeras pruebas experimentales para detección de datos digitales.

2.5 Diseño y fabricación de prototipo electrónico

A partir de tener el servidor listo y adquirir datos digitales desde las placas de desarrollo se integra cada pieza desarrollada en un solo prototipo funcional, como lo podemos ver en la Figura 13. Donde tenemos el sensor OPT101 que capta la intensidad de luz en un rango de infrarrojo a luz visible 300nm-1000nm [7] el cual es alimentado por 20 VCD la señal de salida del sensor es enviada al Arduino UNO para después ser enviada y procesada a la Raspberry Pi 3 B+, conectando una PC para poder visualizar y ejecutar los scripts para que el MVC ejecute el servidor. Con esto el usuario puede solicitar la información por el link del servidor y visualizar ya sea en un teléfono inteligente o una computadora conectados a internet.

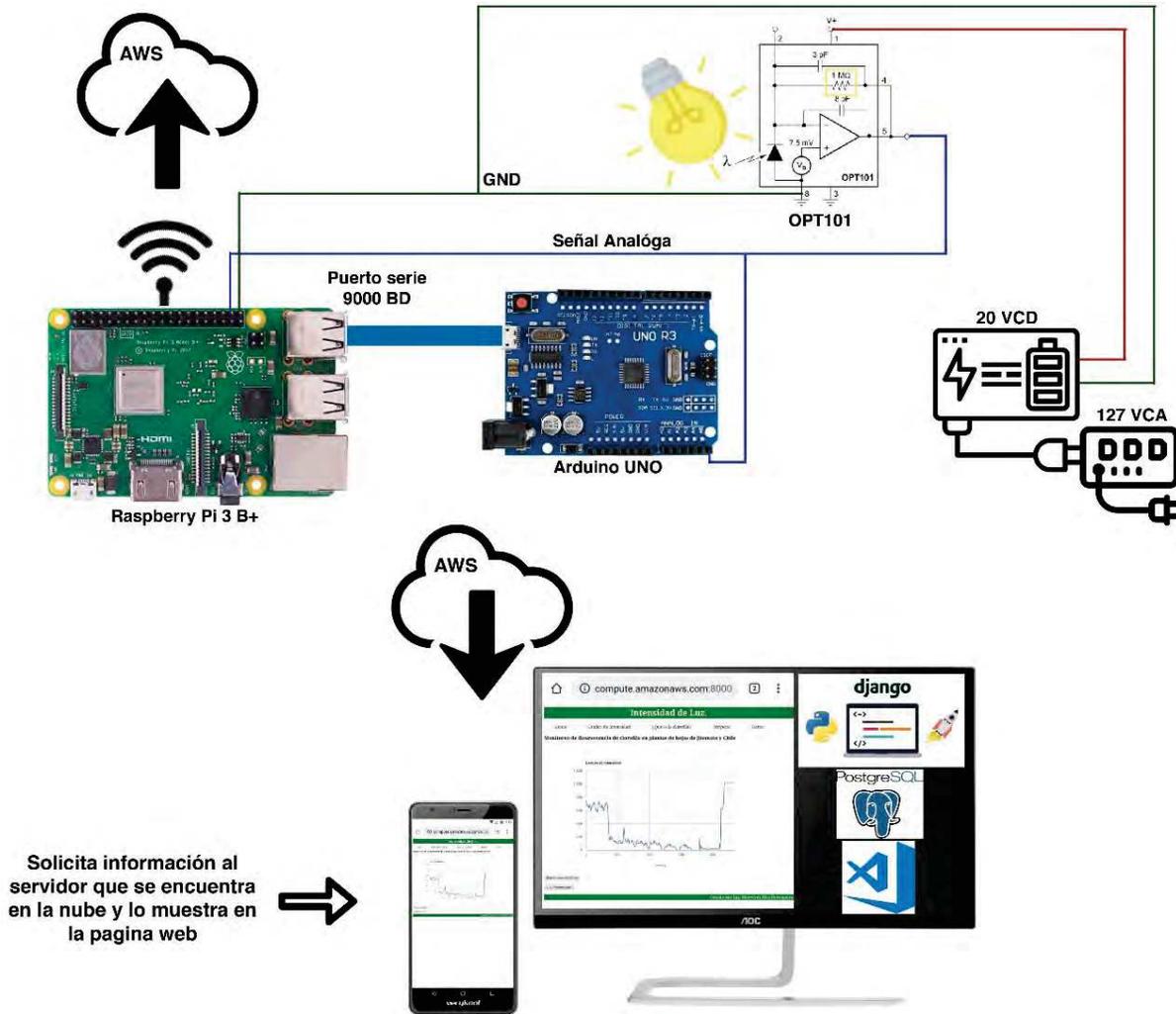


Figura 13. Diagrama de conexiones electrónicas.

3. Resultados

Teniendo el sensor activado y corriendo los scripts `Mary_final.pem` y `Prueba5_intensidad.py` podemos ir a la url : <http://ec2-18-221-79-111.useast-2.compute.amazonaws.com:8000/intro/> y poder obtener la gráfica del monitoreo de nuestras lecturas del fotodiodo ver Figura 14,15 y 17. En la captura de intensidad de luz se pueden visualizar los resultados gráficos en cualquier dispositivo que cuente con red de internet y quiera verlos en cualquier momento y así poder visualizar los datos adquiridos en ese momento un claro ejemplo lo tenemos en las Figuras 14 y 15 en donde se tiene un acceso por medio de un *smartphone*, en la Figura 14 nos muestra el arranque del sistema siendo las 8:45 am ya en las Figura 15 siendo las 9:17 am de ese mismo día se aprecia el sistema de monitoreo más estable. Para las Figuras 17 y 18 podemos visualizar la misma prueba solamente que desde una PC. En la Figura 17 se muestran algunas pruebas iniciales y en la Figura 18 pruebas más estables. En la Figura 16 se observa la página web de inicio del proyecto accediendo desde una PC. No menos importante se coloca una sección para poder borrar la gráfica visualizada en caso de que se tenga algún problema con la red y se pueda acceder nuevamente al monitoreo sin problema. Ya en la Figura 20 se muestra las conexiones eléctricas del prototipo final.

Como lo menciona [7] podemos observar en las gráficas que las lecturas se encuentran en un rango de infrarrojo a luz visible 300nm-1000nm, el eje de ordenadas representa el rango de luz visible expresadas en nm y el eje de abscisas el tiempo transcurrido.



Figura 14. Pruebas iniciales con datos monitoreados de la página de la grafica del proyecto Clorofila accediendo desde un teléfono celular.



Figura 15. Pruebas continuas y más estables con datos monitoreados de la pagina de la gráfica del proyecto Clorofila accediendo desde un teléfono celular.

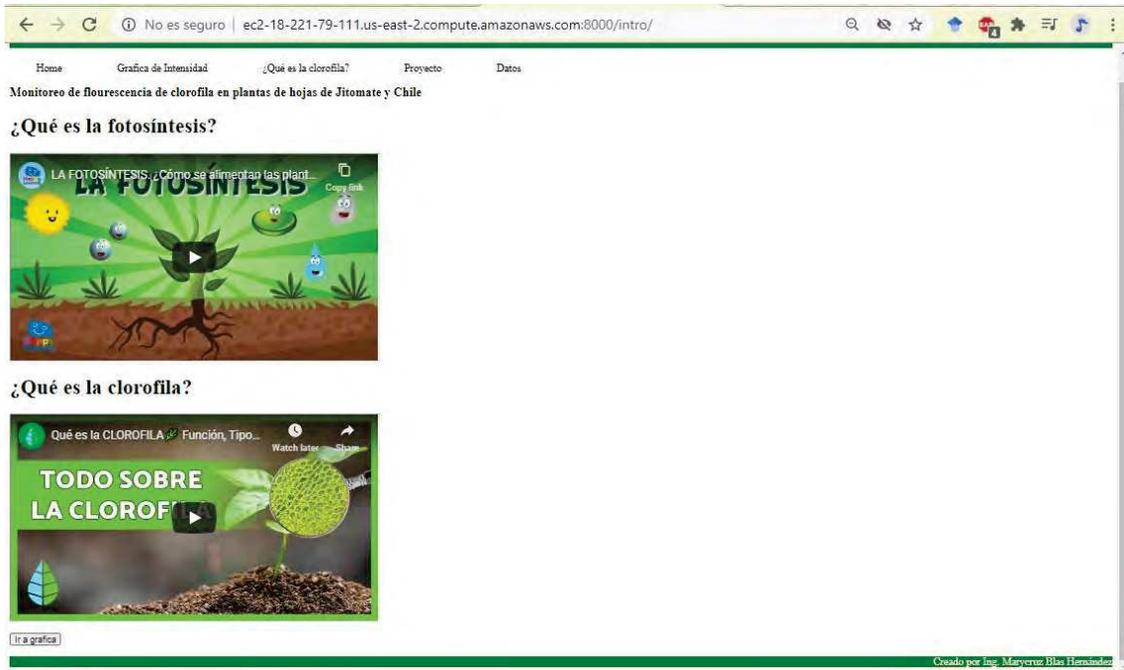


Figura 16. Acceso a la página principal del proyecto Clorofila desde una Pc.

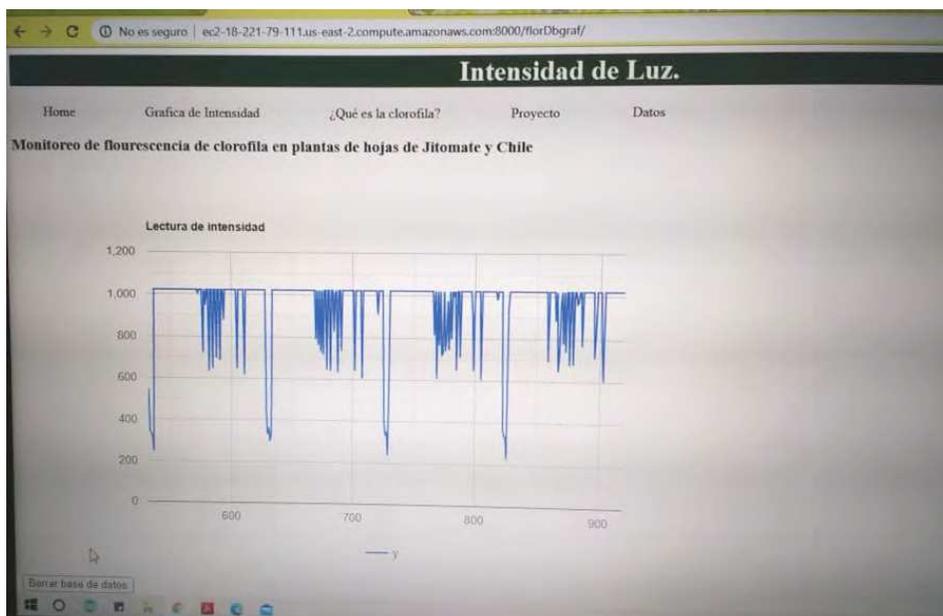


Figura 17. Primeras pruebas de monitoreo de datos accediendo desde una Pc.

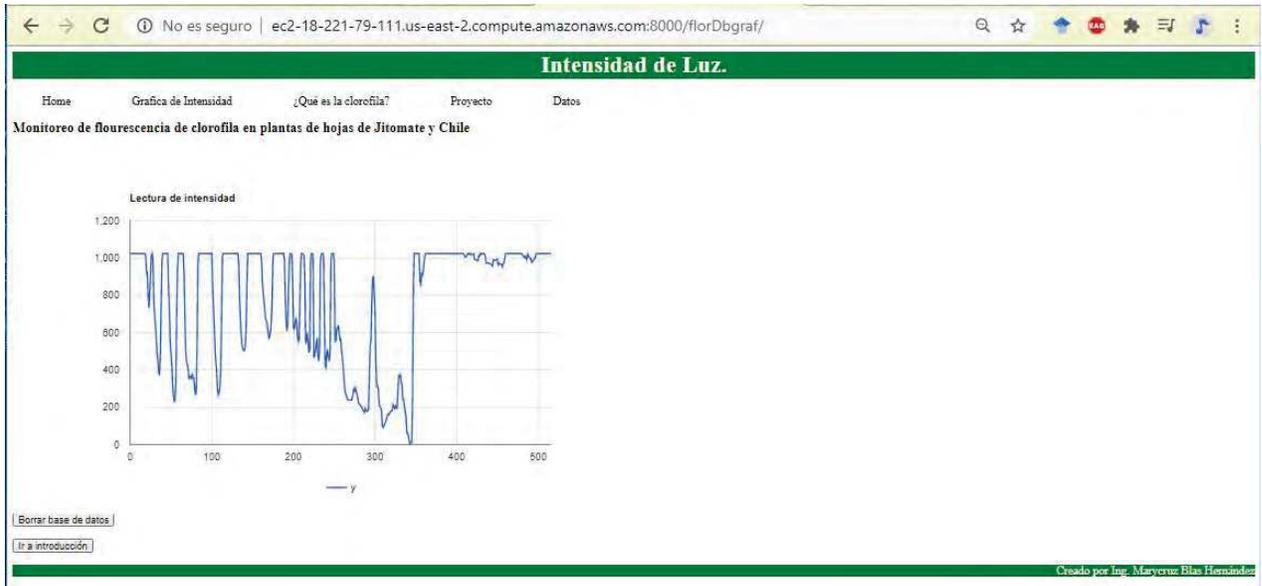


Figura 18. Monitoreo de datos en tiempo real de intensidad de Luz accediendo desde una PC.



Figura 19. Borrando datos de la gráfica accediendo desde una PC.

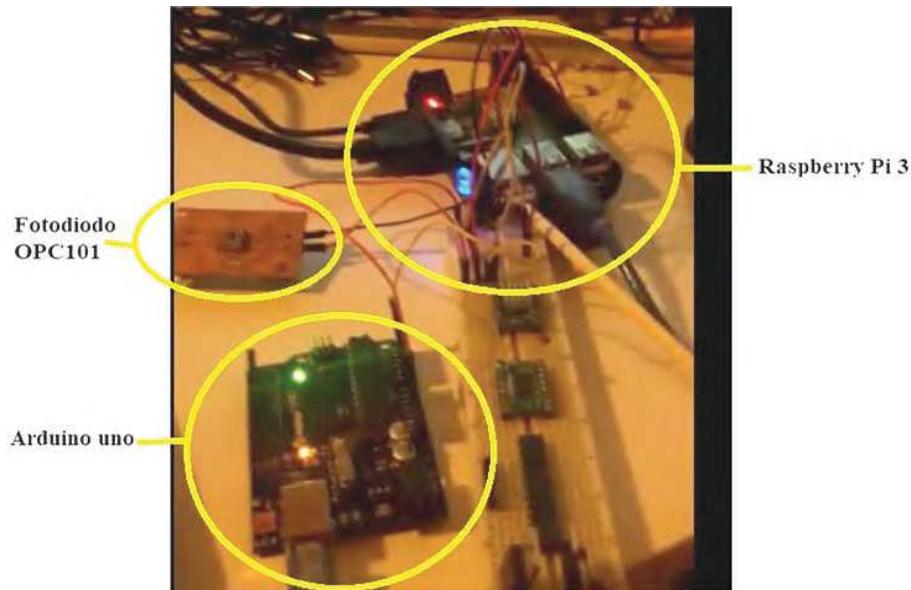


Figura 20. Conexiones eléctricas físicas del sistema de monitoreo



4. Conclusiones

Actualmente los sistemas en tiempo real no son tan en "tiempo real" sino que siempre será un tiempo por muy pequeño que sea desfasado del real, tendremos que considerar implementar un sistema de control que realmente nos indique que estamos monitoreando en tiempo real ya que en otro tipo de proyectos en donde si se requiera que sea totalmente exacto el tiempo como por ejemplo en el área de la medicina y la salud, tendremos muchas decadencias. Sin embargo, el desfase que se tuvo en tiempos de este proyecto fue debido a las conexiones a la red de internet que se tenían del usuario estando en un espacio/tiempo distinto al del prototipo.

Para la validación del proyecto y tener resultados satisfactorios, se realizan varias pruebas en donde se le pidió a varios usuarios ubicados en distintos puntos de la república mexicana que accedieran a la dirección electrónica del servidor para poder visualizar el monitoreo, algunos lugares fueron Jalisco, Tamaulipas, Edo. de México e Hidalgo mientras que el prototipo se encontraba en Querétaro. De igual manera se tuvieron usuarios de Qro, en todos los casos los resultados son satisfactorios.

Referencias

- [1] "SISTEMAS DE MONITOREO EN LÍNEA Mejorando la eficiencia y durabilidad de los motores eléctricos". Expo Energía 2017. Electro Industria Soluciones Tecnológicas para la Minería, Energía e Industria., XVI(190), 28-30. [En línea]. Disponible en <http://www.microbyte.cl/elec/flipbook/201708/>, Santiago de Chile,(2017).
- [2] Ceballos, F. J. "Enciclopedia del lenguaje C". México: Alfaomega/RaMa.. "Lenguajes de Progradation", (2004).
- [3] D. de Luca. "Visual Studio Code: características principales" [En línea]. Disponible en: <https://damiandeluca.com.ar/visual-studio-code-caracteristicas-principales>. [Accedido: 1-oct-2020]
- [4] "Curso Django" . 2020. [En línea]. Disponible en: <https://www.youtube.com/watch?v=7XO1AzwkPPE&list=PLU8oAIHdN5BmfvwXFO7HdPciOCmmYneAB> [Accedido: 30-jun- 2020].
- [5] "¿Qué es la informática en la nube?", 2020. [En línea]. Disponible en: <https://aws.amazon.com/es/what-is-cloud-computing/?nc2=h ql le int cc>. [Accedido: 4-oct-2020]
- [6] "Modelo de DevOps definido",2020. [En línea]. Disponible en: <https://aws.amazon.com/es/devops/what-is-devops/?nc1=f cc>. [Accedido: 4-oct -2020]
- [7] "OPT101 Monolithic Photodiode and Single-Supply Transimpedance Amplifier datasheet (Rev. B)." 2015. OPT101 Monolithic Photodiode And Single-Supply Transimpedance Amplifier. [En línea]. Disponible en: https://www.ti.com/lit/ds/symlink/opt101.pdf?ts=1601994175394&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FOPT101. [Accedido: 5-oct -2020]
- [8] Ríos Caicho, A. G., & Peñafiel Adrián, R. F. "Prototipo de un sistema automatizado para aplicaciones industriales de pequeña escala basado en una interfaz gráfica móvil controlada por arduino (Doctoral dissertation, Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería En Networking y Telecomunicaciones)". (2018).
- [9] Tapia Ayala, C. H., & Manzano Yupa, H. M. "Evaluación de la plataforma arduino e implementación de un sistema de control de posición horizontal" (Bachelor's thesis). (2013).
- [10] Tzuc, O. M., Quijano, J. G., López, J. D. O., & Cetina, R. Q. "Sistema De Monitoreo Inalámbrico De Bajo Costo Para Módulos Fotovoltaicos Empleando Raspberry Pi. Pistas Educativas", 38(120). (2018).
- [11] Vignolo, E. E., Valdez, A. D., Miranda, C. A., Schlesinger, P. L., Chiozza, J. A., & Miranda, C. V. Monitoreo de variables analógicas con Raspberry Pi B+. "Extensionismo, Innovación y Transferencia Tecnológica", 3, 39-45. (2016).



Diseño de una turbina eólica helicoidal usando técnicas computacionales

Tello Campos Alejandro Roberto

Universidad Politécnica del Valle de México, Depto. de Ingeniería Mecatrónica
alejandro.tello@upvm.edu.mx

Resumen

En este diseño de turbina eólica se propone un diseño novel helicoidal de 3 alabes. Se estudia la aerodinámica de la turbina haciendo énfasis en la obtención del torque generado con ayuda de la herramienta Solidworks™. Se obtuvo una curva de torque generado contra velocidad del viento. La aerodinámica de la turbina se observó en los gráficos de presión y velocidad de los vientos obtenidos. Se realizaron simulaciones para 10, 15 y 20 m/s de velocidad del viento. Se enuncia el tipo de control de una turbina de este tipo. Asimismo, se realiza un análisis de esfuerzos generados en la turbina debido a la carga de viento y se obtiene el máximo esfuerzo generado así como su deflexión máxima del material considerando acero aleado, aluminio T1060, plástico y madera balsa. Desde el punto de vista de esfuerzos se encontró que el aluminio T1060 está sujeto al menor esfuerzo de Von Misses de 78615 N/m² y el mínimo desplazamiento máximo de 7.76x10⁻⁷ mm.

Palabras clave: Turbina eólica de eje vertical, simulación, CFD, esfuerzos.

1. Introducción

La generación de energía eléctrica a partir del viento data de la época de los molinos de viento que usaban los egipcios en 2 A.C. [1] para proporcionar viento a un órgano. La civilización persa desarrolló el primer molino de eje vertical en 600 D.C. Sin embargo no fue hasta que Savonius desarrolló el primer aerogenerador de eje vertical para producir energía eléctrica y que posteriormente Darrius patentó su diseño. Finalmente USA realizó el diseño de un aerogenerador de eje horizontal de 1250 kW y Dinamarca creó un aerogenerador que produjera corriente alterna. El diseño de turbina de eje vertical de tipo hélice de 50 kW fue desarrollado por una empresa finlandesa para suministrar energía al clima de un centro comercial. En una línea de investigación que usa CFD para analizar los diseños de turbinas de eje vertical rectas con respecto a las de eje vertical helicoidales [2] se obtuvieron resultados de coeficiente de torque para diferentes ángulos de torcimiento de los álabes helicoidales. Se concluyó que el más apropiado era el que tenía un ángulo de 60° y generaba vórtices de forma más uniformemente distribuidos. Otros investigadores [3] han estudiado de forma experimental y computacional la turbina de eje vertical helicoidal con 2, 3 y 4 álabes y concluyeron que para las relaciones de velocidad de punta (TSR) bajas de 0.02 a 0.06 el diseño de 2 álabes ofrecía más potencia y menos pérdidas mientras que para relaciones de velocidad de punta del orden de 0.4 a 0.6 se obtenía mayor potencia con los diseños de 3 y 4 álabes. Otras técnicas [4] se han propuesto para el análisis del comportamiento aerodinámico a través de turbinas de eje vertical que usan la teoría de variable compleja para obtener las descripciones del flujo a través de los alabes. También otros investigadores [5] han propuesto el diseño de dos rotores de tipo Darrius en tándem y han realizado análisis del flujo de aire con CFD y han obtenido los esfuerzos de Von Misses máximos y deformaciones máximas haciendo análisis estático y dinámico de los rotores. En este trabajo se presenta un análisis aerodinámico de una turbina de eje vertical helicoidal de 3 álabes haciendo uso de CFD y por medio de Solidworks™ se obtuvo el comportamiento estático y dinámico de la misma.

2. Control de turbina eólica

El viento contiene energía que es susceptible de aprovecharse la cual esta expresada como:

$$P = \frac{1}{2} \rho A c_p V^3 \quad (1)$$

Dicha energía está limitada por el llamado límite de Betz. Y la determinación del coeficiente c_p depende directamente de la geometría de la turbina y está definida por la relación de velocidad de la punta a la velocidad del viento, o bien:

$$TSR = \frac{wR}{V_{viento}} \quad (2)$$

El control de la turbina se puede realizar de forma activa por medio de deflectores del viento o bien manejando la turbulencia del aire por medio de la forma geométrica misma de la turbina. En forma pasiva se realiza por medio de medios electrónicos que realizan el acoplamiento de la turbina que genera potencia a una velocidad variable y la red que opera a una frecuencia fija. Básicamente, el esquema de control de una turbina sería el dado en la Figura 1.

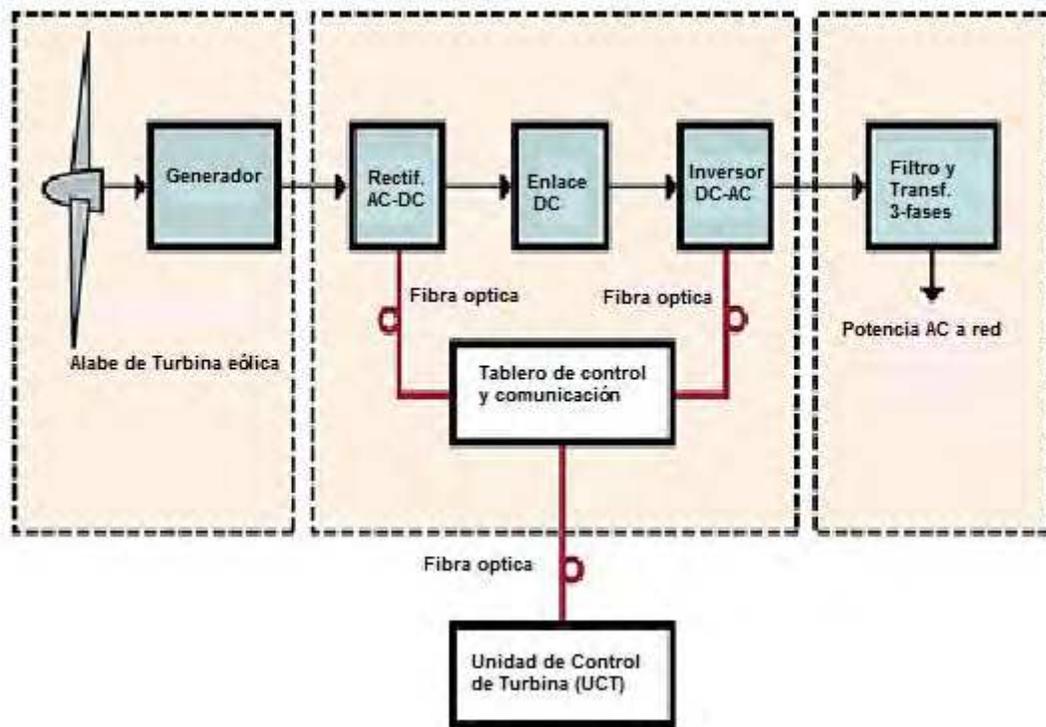


Figura 1. Esquema de control de una turbina eólica Fuente: Avago Technologies

Cuando se usa el control de par de la turbina se puede realizar controlando el par desarrollado por medio del control del generador eléctrico como se muestra en la Figura 2. Para este tipo de control se estaría controlando el par desarrollado alrededor de un punto máximo que se realiza mediante un monitoreo del punto de operación máximo para cada velocidad del viento.

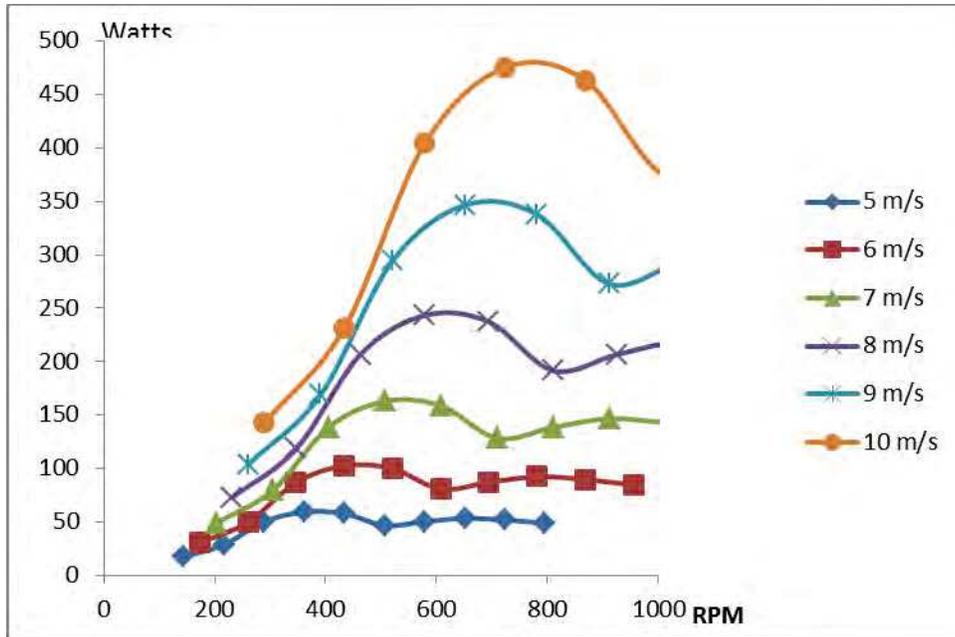


Figura 2. Curva de Potencia aerodinámica-RPM de turbina eólica.

El control usando un generador de inducción se basa en el uso de control del lado del generador y otro control del lado de la red como se muestra en la Figura 3. Este control se basa en poder generar un voltaje de DC constante y generar un voltaje de AC en el lado del inversor de acuerdo a un convertidor “back to back”.

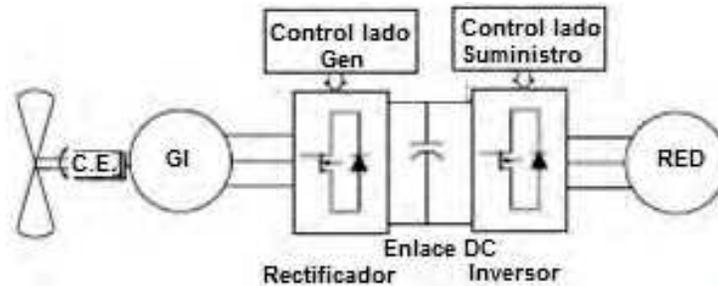


Figura 3. Control de generador de imanes permanentes Fuente: Power Conditioning, University of Strathclyde Glasgow.

3. Modelación aerodinámica

Se diseñó una turbina helicoidal de eje vertical de tres alabes en Solidworks™ que posteriormente se usó para el análisis aerodinámico haciendo uso del mismo Solidworks™. Para el modelo aerodinámico se consideró la resolución de las ecuaciones de Navier-Stokes, continuidad y turbulencia que se discretizan usando la técnica del elemento finito [6]. El modelo discretizado en Solidworks™ se muestra en la Figura 4.

Se definió una caja que representó un túnel de viento donde se instaló la turbina eólica. Las condiciones en la entrada de la caja fueron la velocidad del viento la cual se varió en un rango de 10,15 y 20 m/s. Las condiciones de salida fueron de presión igual a 101 kPa y en las paredes de la caja se estableció como condiciones de cero velocidad en las mismas. Ver Figuras 4 y 5.



Figura 4. Discretización de la turbina helicoidal de eje vertical.

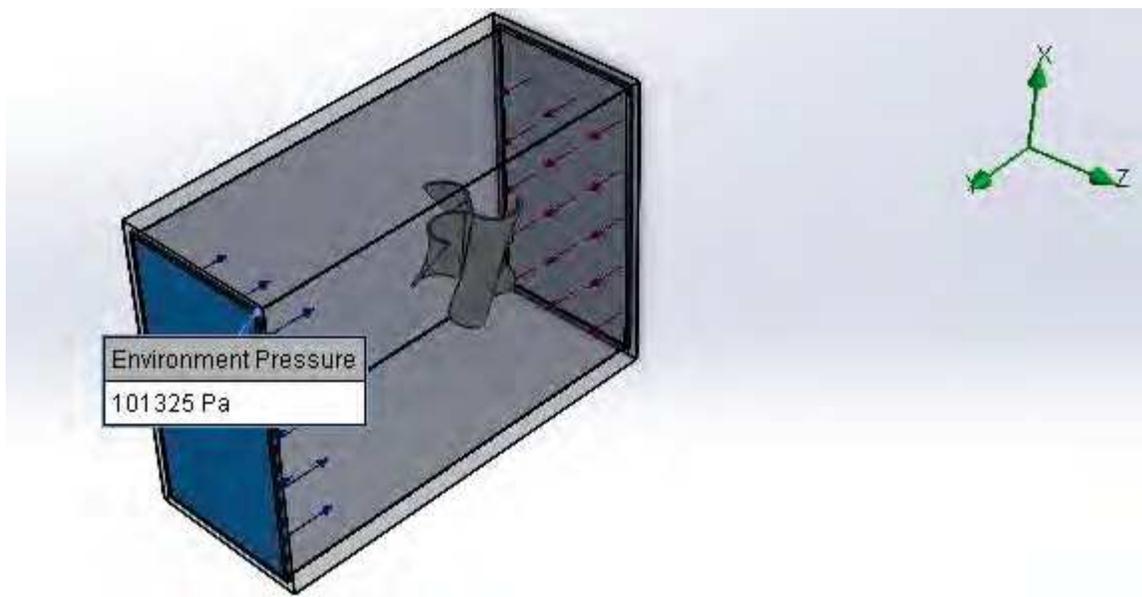


Figura 5. Definición de dominio a simular.

Se definieron los parámetros a observar que en este caso fueron las velocidades del aire así como las presiones y torques desarrollados por el efecto del viento. El análisis se realizó de forma que la turbina estaba estática. Los resultados obtenidos se muestran en las Figuras 6 y 7 como gráficos de velocidades y presiones. Las presiones que se muestran no son uniformes en toda la turbina. Se concentran más en las secciones que encaran el viento. Ahí son mayores que las que se presentan en el álabo que está ubicado aguas debajo de la turbina.

4. Análisis de esfuerzos

En este análisis se usó el campo de presiones derivado del análisis aerodinámico y se obtuvieron los esfuerzos máximos en diferentes materiales como aluminio T-1060, acero aleado, plástico y madera balsa. En las Figuras 8 a 15 se muestran los gráficos de esfuerzos y deflexiones máximas encontradas.

De la Figura 8 se observa que los esfuerzos máximos se presentan en las secciones pegadas a la parte central de la turbina y bordes exteriores de la misma. Estos valores son consistentes con las

presiones ejercidas por el viento en dichas secciones. En ningún punto de la turbina se rebasa el límite elástico del acero aleado. La Figura 9 presenta las deformaciones máximas que se hacen patentes en la parte superior de la turbina.

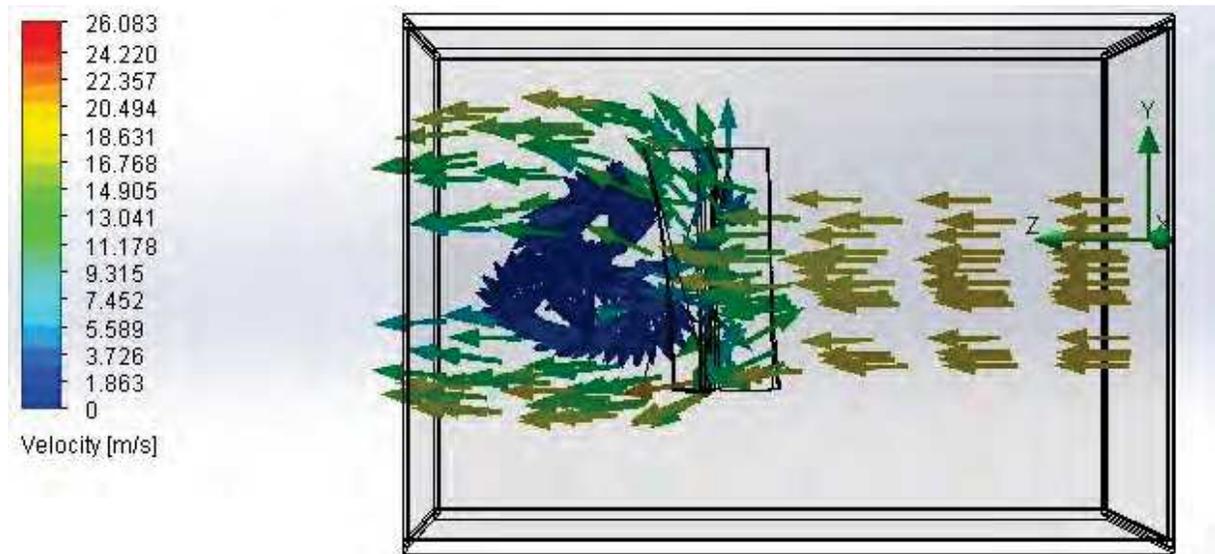


Figura 6. Gráfico de velocidades.

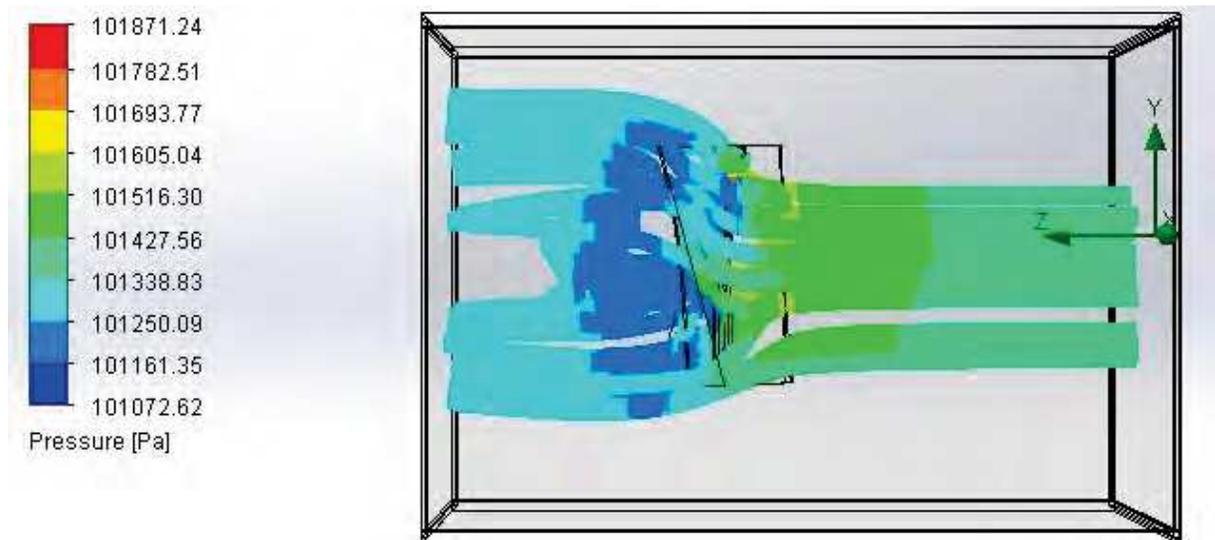


Figura 7. Gráfico de presiones.

En la Figura 10 el esfuerzo máximo se presenta en las secciones central y bordes de la turbina. El límite elástico del aluminio no se rebasa y las deformaciones máximas se presentan en la sección superior de la turbina.

En la Figura 12 se muestra el esfuerzo máximo en las secciones central y bordes exteriores de la sección inferior turbina. Las deformaciones máximas observadas en la Figura 13 igualmente se presentan en la parte superior de la turbina.



Nombre del modelo: flow turbine 2
Nombre de estudio: Análisis estático 1 (-Predeterminado-)
Tipo de resultado: Análisis estático tensión nodal Tensiones1
Escala de deformación: 8.1614.5

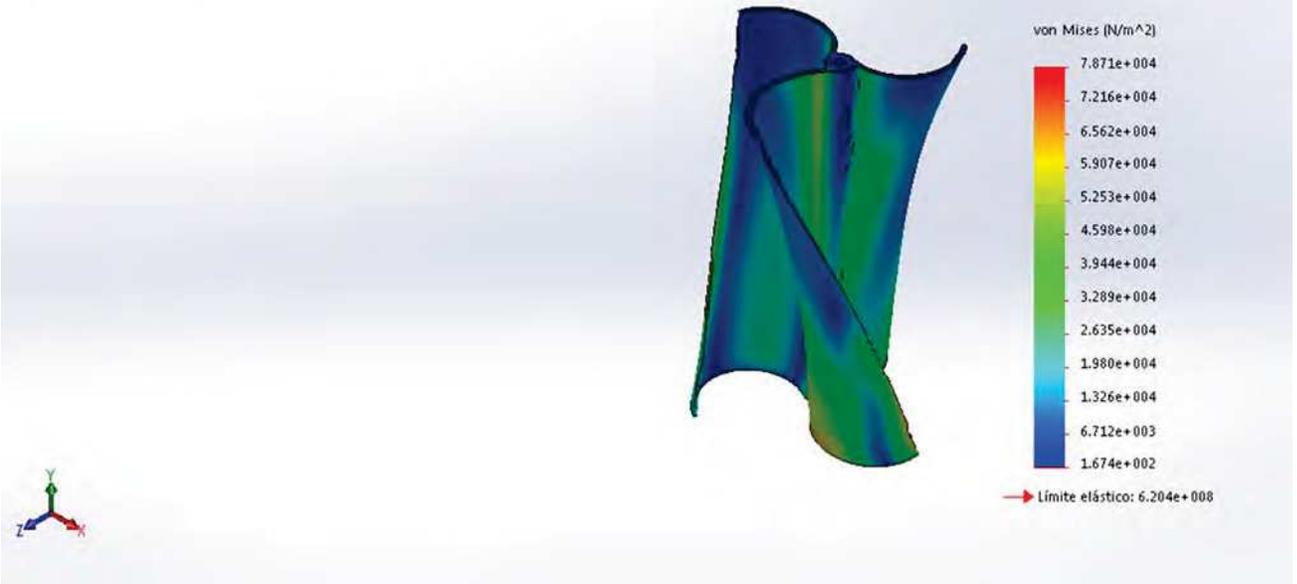


Figura 8. Grafico de esfuerzos de turbina de acero aleado.

Nombre del modelo: flow turbine 2
Nombre de estudio: Análisis estático 1 (-Predeterminado-)
Tipo de resultado: Desplazamiento estático Desplazamientos1
Escala de deformación: 8.1614.5

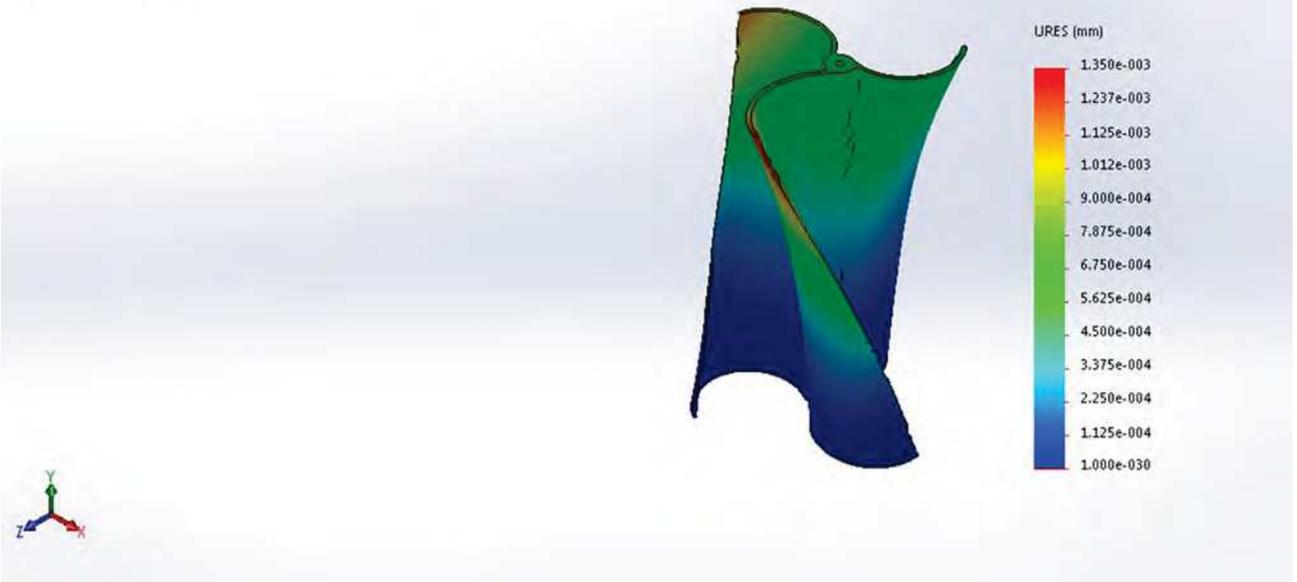


Figura 9. Grafico de desplazamientos máximos de turbina de acero aleado.



Nombre del modelo: flow turbine 2
Nombre de estudio: Análisis estático 1 (Predeterminado-)
Tipo de resultado: Análisis estático tensión nodal Tensiones1
Escala de deformador: 26803.1

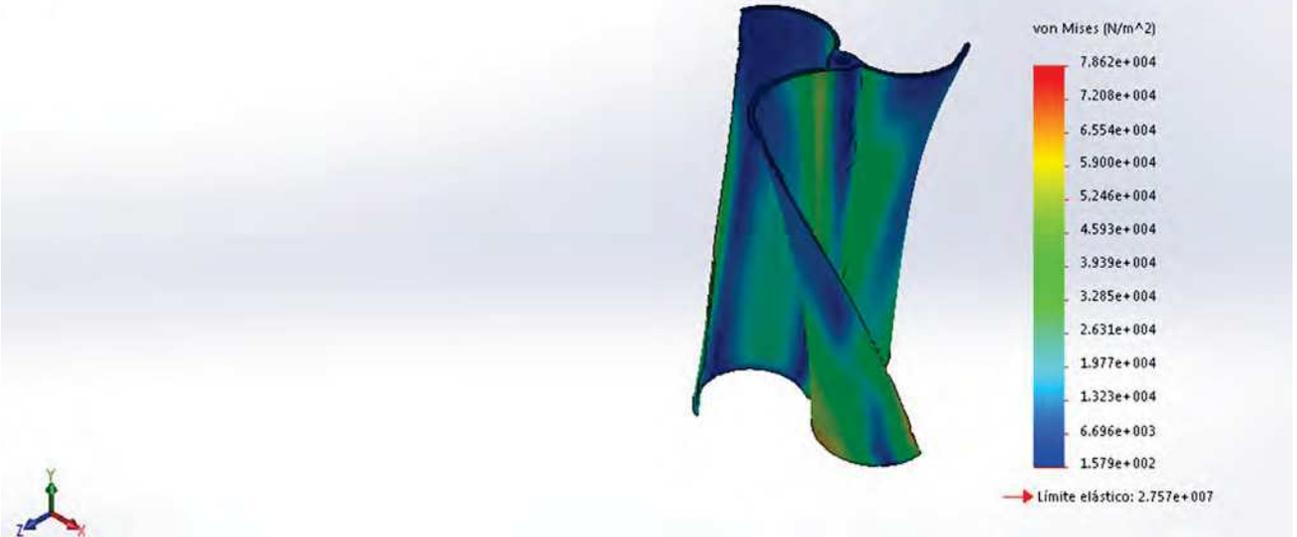


Figura 10. Gráfico de esfuerzos de turbina de aluminio T1060.

Nombre del modelo: flow turbine 2
Nombre de estudio: Análisis estático 1 (Predeterminado-)
Tipo de resultado: Desplazamiento estático Desplazamientos1
Escala de deformador: 26803.1

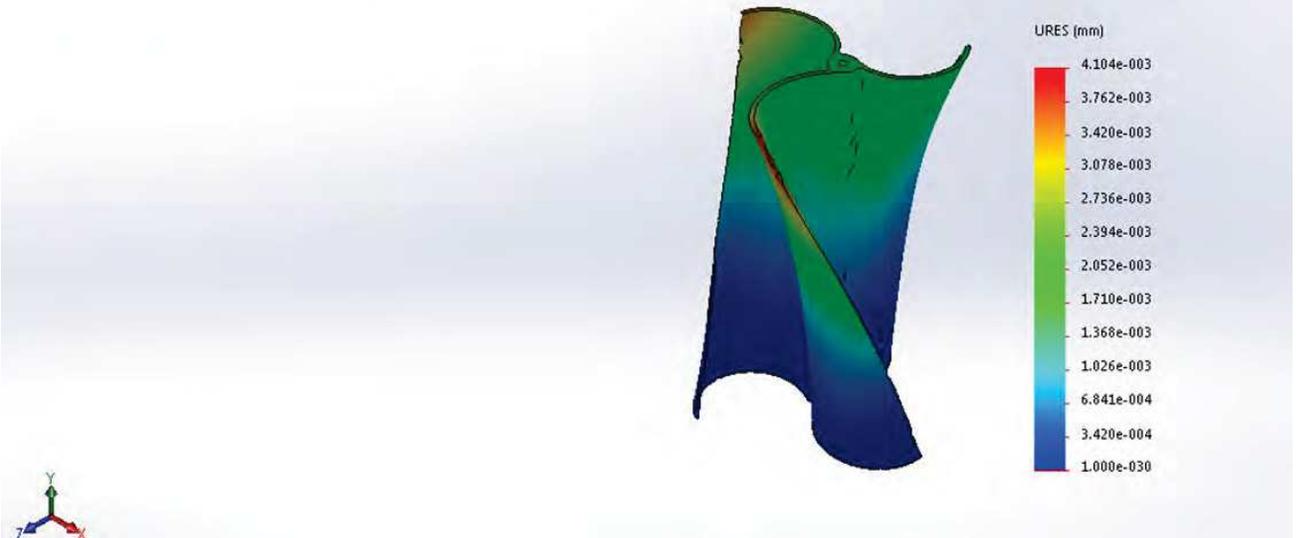


Figura 11. Gráfico de desplazamientos de turbina de aluminio T1060.



Nombre del modelo: flow turbine 2
Nombre de estudio: Análisis estático 1 (Predeterminado)
Tipo de resultado: Análisis estático tensión nodal Tensiones1
Escala de deformación: 779.231

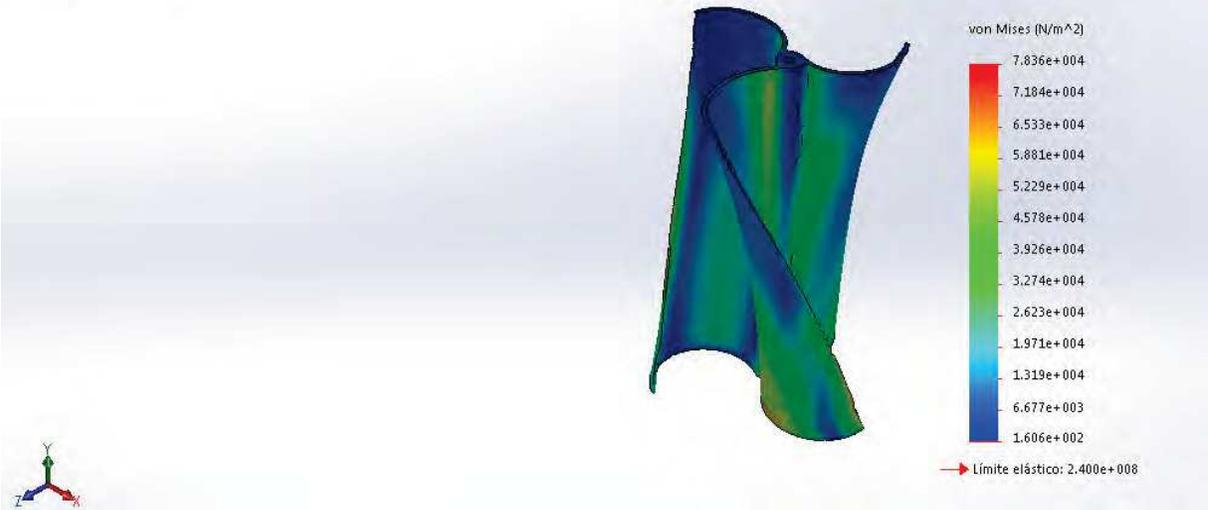


Figura 12. Grafico de esfuerzos de turbina de plástico.

Nombre del modelo: flow turbine 2
Nombre de estudio: Análisis estático 1 (Predeterminado)
Tipo de resultado: Desplazamiento estático Desplazamientos1
Escala de deformación: 779.231

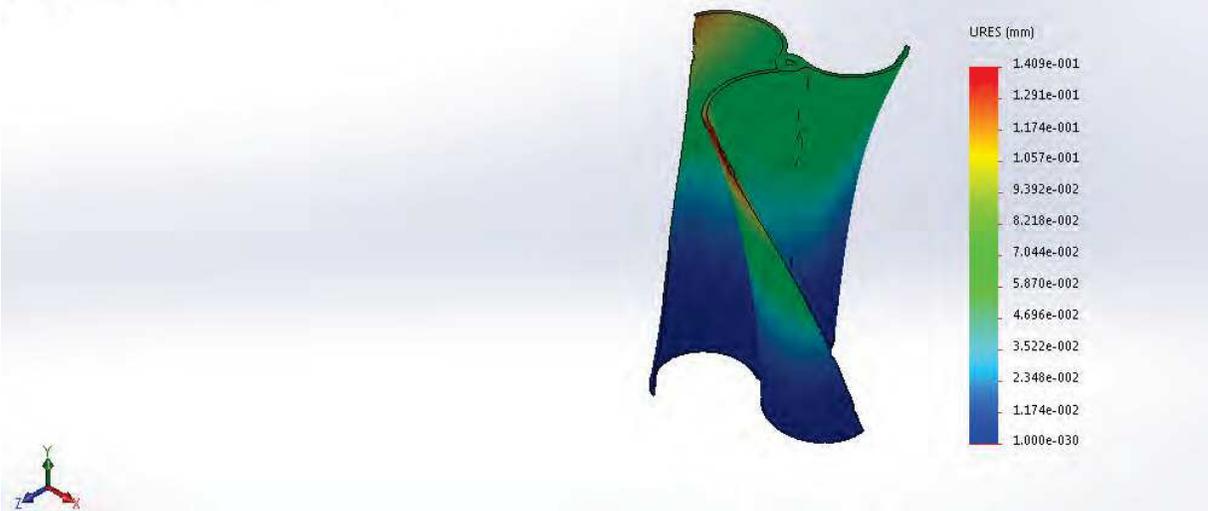


Figura 13. Grafico de desplazamientos de turbina de plástico.

Los esfuerzos máximos se presentan en las secciones centrales y bordes exteriores de la parte inferior de la turbina (ver Figura 14). Ahí en la parte central se presentan máximas presiones y por tanto máximos esfuerzos. En los bordes se presentan también máximas presiones debido a que el aire tiende a acelerarse en dichas regiones. La Figura 15 muestra máximas deformaciones en la región superior de la turbina.

Nombre del modelo: flow turbine 2
Nombre de estudio: Análisis estático 1 (Predeterminado-)
Tipo de resultado: Análisis estático tensión no dal Tensiones1
Escala de deformación: 1166.16

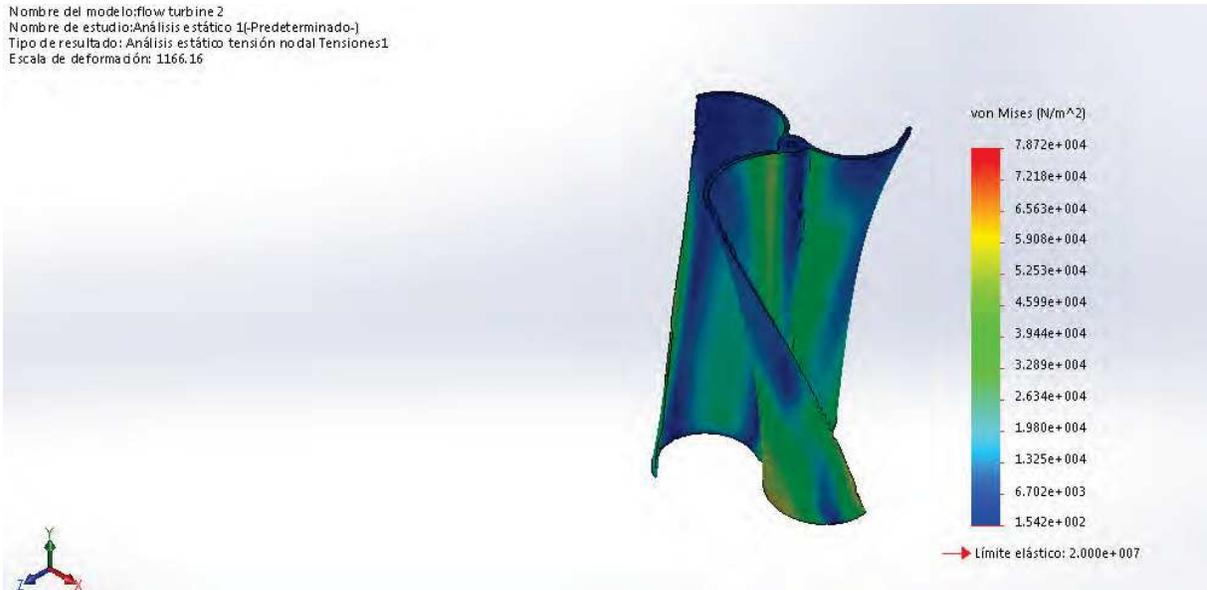


Figura 14. Grafico de esfuerzos de turbina de madera balsa.

Nombre del modelo: flow turbine 2
Nombre de estudio: Análisis estático 1 (Predeterminado-)
Tipo de resultado: Desplazamiento estático Desplazamientos1
Escala de deformación: 1166.16

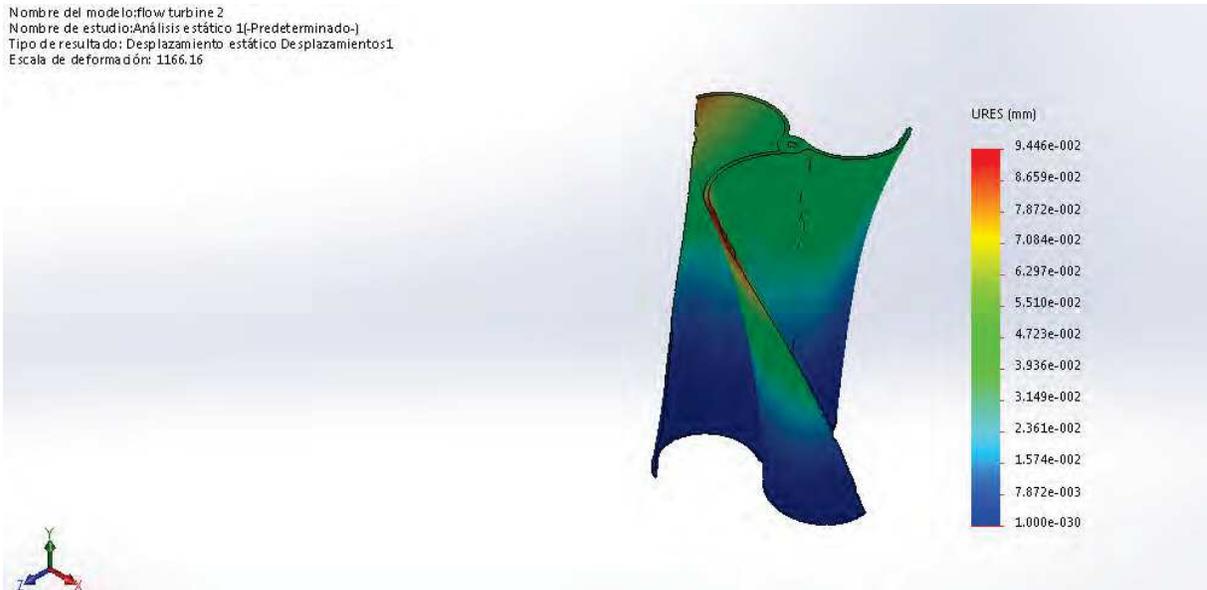


Figura 15. Grafico de desplazamientos de turbina de madera balsa.

5. Resultados

De acuerdo a los resultados de la simulación aerodinámica se obtuvieron las variaciones del torque y potencia desarrollados versus la velocidad del viento. En la Tabla 1 de muestran dichos resultados.

En cuanto al análisis de esfuerzos de la turbina se encontró que el aluminio T1060 presenta los menores esfuerzos máximos generados y resultó con el menor desplazamiento máximo en mm como se muestra en la Tabla 2.



Tabla 1. Potencia y torque de turbina

Velocidad del viento, m/s	Torque, N-m	Velocidad angular, rad/s	Potencia aerodinámica, Watts
5	3.55	15.15	53.79
10	14.13	30.30	428.18
15	31.01	45.45	1409.55

Tabla 2. Esfuerzos y desplazamientos máximos de turbina

Material	Esfuerzo máximo de VonMises, Pa	Desplazamiento máximo, mm
Madera balsa	78724.4	1.96E-05
Plástico	78630.3	3.14E-05
Aluminio T1060	78615.8	7.76E-07
Acero aleado	78706.8	2.78E-07

6. Conclusiones

Como conclusión se puede decir que el estudio sirve para analizar la configuración de la turbina eólica de eje vertical operando a varias velocidades del viento. Se puede ver que la turbina desarrolla una gran potencia aerodinámica a velocidades del viento normalmente encontradas en aplicaciones domésticas e industriales. Esta potencia aerodinámica va a ser contrarrestada por la fricción del viento y por la potencia restauradora del generador eléctrico. Es por eso que se necesita tener un control estricto de la velocidad de rotación de la turbina para obtener el máximo de potencia a diferentes velocidades del viento. También los gráficos de presión muestran las zonas más expuestas a altos esfuerzos en la estructura de la turbina. Y los gráficos de velocidad muestran la gran turbulencia creada por la interacción del viento y la estructura de la turbina. Esto puede llevar a vibraciones que se tienen que tomar en cuenta en el diseño de la misma. En este sentido se pretende realizar un análisis experimental de la turbina para medir las vibraciones, par y potencia desarrollados por la turbina de eje vertical.

Los análisis computacionales mostrados resultan muy útiles para evaluar materiales de la turbina en función de los esfuerzos máximos generados y los desplazamientos desarrollados. Desde el punto de vista de fabricación es lo que se tendría que analizar dadas las restricciones de maquinado y costos involucrados. Asimismo, se sugiere realizar un análisis de costos de fabricación así como la comparación contra otros diseños encontrados en el mercado.

Referencias

- [1] Fox B., Bryans L., Flynn D., Jenkins N., Milborrow D., O'Malley M., Wattson R., Anaya-Lara O., *"Wind Power Integration"*, IET Renewables Series 14, 2nd edition, 2014.
- [2] Alaimo A., Esposito A., Messineo A., Orlando C., Tumino D., *"3D CFD Analysis of a Vertical Axis Wind Turbine"*, Energies 2015, 8.
- [3] Salyers T., *"Experimental and Numerical Investigation of Aerodynamic Performance of Vertical-Axis Wind Turbines Performance with Various Blade Designs"*, Georgia Southern University, 2016.
- [4] Liu H., *"Linear Strength Panel Vortex Method for NACA 4412 Airfoil"*, IOP Conference Series: Materials Science and Engineering, 2018.
- [5] Torres M., *"Diseño de un Generador Eólico de Eje Vertical de Baja Potencia"*, Universidad Tecnológica de Pereira, 2015.



Diseño y construcción de un robot cuadrúpedo para la difusión de las Ingenierías en Electrónica y Mecatrónica

Santiago Espinosa Felipe, Barahona Ávalos Jorge Luis

Universidad Tecnológica de la Mixteca
Instituto de Electrónica y Mecatrónica
Km. 2.5 Carretera a Acatlima
Huajuapán de León, Oaxaca
fsantiag@mixteco.utm.mx, jbarahona@mixteco.utm.mx

Resumen

Una tarea fundamental de las universidades es la difusión de su oferta educativa para que los egresados del nivel medio superior conozcan un poco de las diferentes opciones posibles, sin embargo, actualmente fluye un exceso de información por Internet que dificulta captar la atención de una audiencia juvenil. Con el propósito de favorecer la difusión de las Ingenierías en Electrónica y Mecatrónica, en la Universidad Tecnológica de la Mixteca (UTM) se desarrolló un robot cuadrúpedo con 12 grados de libertad. El robot es capaz de caminar hacia adelante, atrás y girar a la izquierda o derecha, comandando los movimientos desde un dispositivo móvil con SO Android a través de una interfaz Bluetooth, además, el robot puede caminar y evitar obstáculos de forma autónoma. El diseño de las partes se hizo en Solid Works, su manufactura con una impresora 3D y una máquina CNC. Cada pata está compuesta por 3 eslabones con un servomotor por eslabón. Para el control de los movimientos se realizó la programación de una tarjeta Arduino Mega. La experiencia ha demostrado que cuando el robot se pone en operación genera mucho interés en los presentes, no sólo observan el desempeño del sistema, también cuestionan sobre los conocimientos que deben adquirir para crear sistemas similares y los semestres de la carrera en que obtendrán estos conocimientos.

Palabras clave: Android, Arduino, Cuadrúpedo, Difusión, Robot.

1. Introducción

La difusión es un trabajo importante que las universidades realizan para la captación de alumnos, existen diferentes estrategias para la misma, como visitas a instituciones, ferias educativas, casa abierta a la comunidad, etc. En todos los casos, el objetivo es la motivación de los aspirantes para que opten por alguna de las carreras que la institución oferta. Cuando se trata de difundir carreras del ámbito tecnológico, como las ingenierías en Electrónica o Mecatrónica, los alumnos requieren más que palabras, diapositivas o videos para enfocar su atención. Algunas instituciones utilizan robots comerciales con este propósito, un ejemplo de ello son los Nao [1], que son muy versátiles y se les pueden programar rutinas de exhibición muy atractivas, pero por su alto costo no conviene dedicar un robot de esta categoría sólo para difusión ya que se pone en riesgo su integridad. Además, el robot tiene una arquitectura compleja y poco transparente, tampoco es simple relacionar los cursos de una ingeniería con la operación y partes del robot.

Con la finalidad de apoyar la difusión de las Ingenierías en Electrónica y Mecatrónica, es conveniente mostrar a los aspirantes universitarios prototipos que integran y aplican las diferentes áreas del conocimiento de las carreras en cuestión, por ejemplo, el robot cuadrúpedo con 12 grados de libertad descrito en el presente ensayo. Se trata de un robot que recibe comandos desde un dispositivo móvil con SO Android, la comunicación del robot con el móvil es inalámbrica, a través de bluetooth, para ello,



el cuadrúpedo contiene un módulo receptor con número de serie HC-06 y fue necesario desarrollar la aplicación para el móvil empleando Applinventor [2].

El cuadrúpedo tiene dos modos de operación, un modo manual en el que responde a comandos recibidos por bluetooth mediante los cuales se le puede solicitar al robot que se flexione, avance, retroceda, gire a la izquierda o gire a la derecha. También se le puede solicitar que ingrese a un modo autónomo, en el que el cuadrúpedo va a avanzar continuamente hacia adelante y si encuentra un obstáculo lo va a evadir al girar de manera alternada a la izquierda o a la derecha, los obstáculos son detectados por medio de un sensor ultrasónico con número de serie HC-SR04.

El sistema completo es coordinado por una tarjeta Arduino Mega, la cual cuenta con un microcontrolador ATmega2560, se elige esta tarjeta por su bajo costo y porque cuenta con los recursos suficientes para el manejo de los servomotores, módulo bluetooth y sensor ultrasónico.

Existen diferentes diseños para los robots cuadrúpedos y se han desarrollado con distintos propósitos, por ejemplo, Bustos-Torres y colaboradores diseñaron un robot enfocado a una competencia denominada "RoboGames" [3], el cuadrúpedo participó en la categoría titulada "Mech Warfare", que es una competencia extrema entre robots equipados con cámaras y pistolas de aire, controlados a distancia por medio de una computadora, en el documento se hace un análisis estático y del robot en movimiento. Otro ejemplo es el trabajo desarrollado por Miret-Rubio [4], quien en su trabajo de fin de Master en Mecatrónica realizó el diseño e implementación de un cuadrúpedo con 3 GDL por articulación, con la finalidad de emular el movimiento de los animales de 4 patas, aplicando cinemática inversa y con los sensores adecuados, consigue que el robot pueda caminar en un terreno irregular. Un último ejemplo es el "Proyecto At-As" [5], un trabajo de fin de Grado en el que Prado-Navarrete describe el Diseño de un robot cuadrúpedo imprimible, fabricación e implementación de la locomoción, el documento describe las diferentes fases del proyecto, desde el diseño hasta la puesta en marcha, pasando por un análisis de la cinemática del robot.

2. Requerimientos del proyecto

Puesto que la tarea principal del cuadrúpedo es la difusión de las ingenierías en Electrónica y Mecatrónica, no hay restricción en las dimensiones o peso del robot, sólo se parte del hecho de que una vez terminado el robot debe ser visible desde cualquier punto de un aula o un auditorio pequeño, por ello, se determina que el cuadrúpedo debe ser relativamente grande y se considera una estructura de tamaño aproximado de 50 x 50 cm², con una altura entre 20 y 30 cm para que la locomoción se pueda apreciar a distancia, estas dimensiones también son adecuadas por el tamaño de los servomotores empleados, que son de la marca TowerPro con número de serie MG995R.

En los siguientes apartados se describen otros requerimientos relacionados con la estructura del robot, diseño de las partes, suministro de energía y tarjeta de control.

2.1 Estructura del cuadrúpedo.

Para el diseño del robot se propuso la estructura mostrada en la figura 1. En la figura 1(a) se puede ver la distribución de los 12 grados de libertad para permitir al robot diferentes tipos de movimiento: avance, retroceso, giros y flexiones. Cada grado de libertad será manejado por un servomotor porque no requiere de un movimiento mayor a 180° y porque en su estructura interna un servomotor incluye un control de posición retroalimentado manejado por una señal modulada en ancho de pulso (PWM) y una caja de engranes metálicos que garantizan un alto torque, en el MG995R el torque es de 10 kg/cm [6]. En la figura 1(b) se muestra la forma en que se colocarán 3 servomotores en cada una de las extremidades, los superiores hacen posible el avance del robot y los otros dos se deben coordinar para flexionar las extremidades.

El cuadrúpedo debe tener un peso ligero, para ello, se eligió el uso de un material basado en aluminio denominado dibond para las partes planas y plástico ABS a través de impresiones en 3D para

los enlaces o conexiones entre las partes planas. El dibond es un material compuesto a partir de dos capas de aluminio y una de polietileno, lo que le confiere estabilidad, peso ligero y alta resistencia, esto lo hace un candidato ideal para proyectos de robótica y similares.

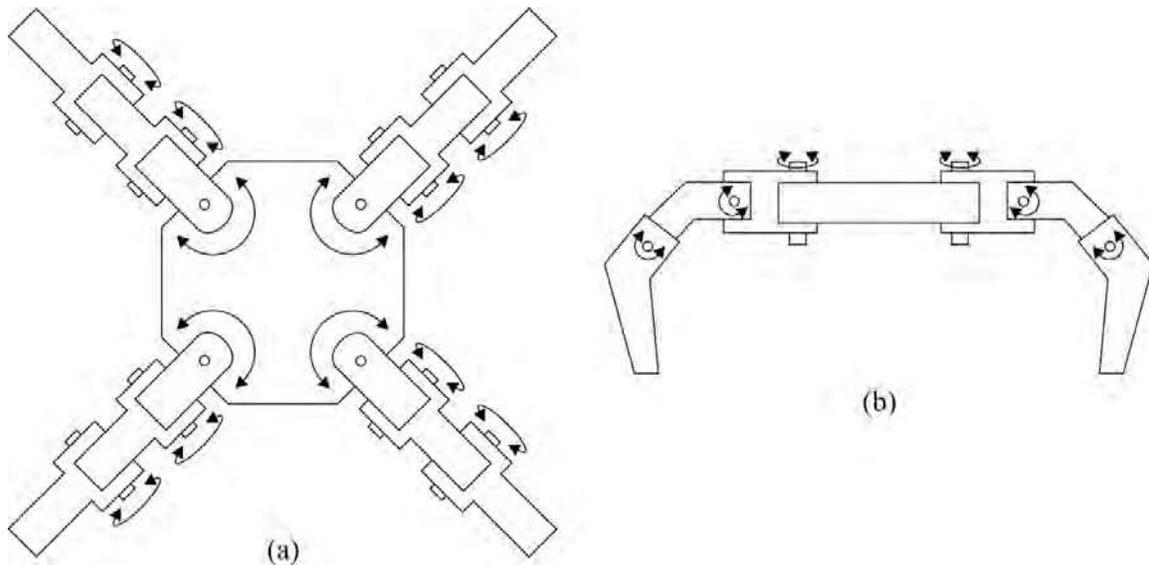


Figura 1. Propuesta para la estructura del cuadrúpedo.

2.2 Diseño de las partes

Se observan dos elementos principales en la estructura propuesta para el cuadrúpedo, el cuerpo y las patas, el diseño de ambas partes se realizó en Solid Works. En el cuerpo se debe alojar la batería y el circuito de control, en la figura 2 se muestra su diseño, en 2(a) se pueden ver las 11 piezas que lo conforman y en 2(b) se muestra su acoplamiento, la altura del cuerpo es de 33 mm y está determinada por el tamaño de los servomotores, con las tapas se crea una caja para ubicar la batería con los cables para la distribución de la energía.

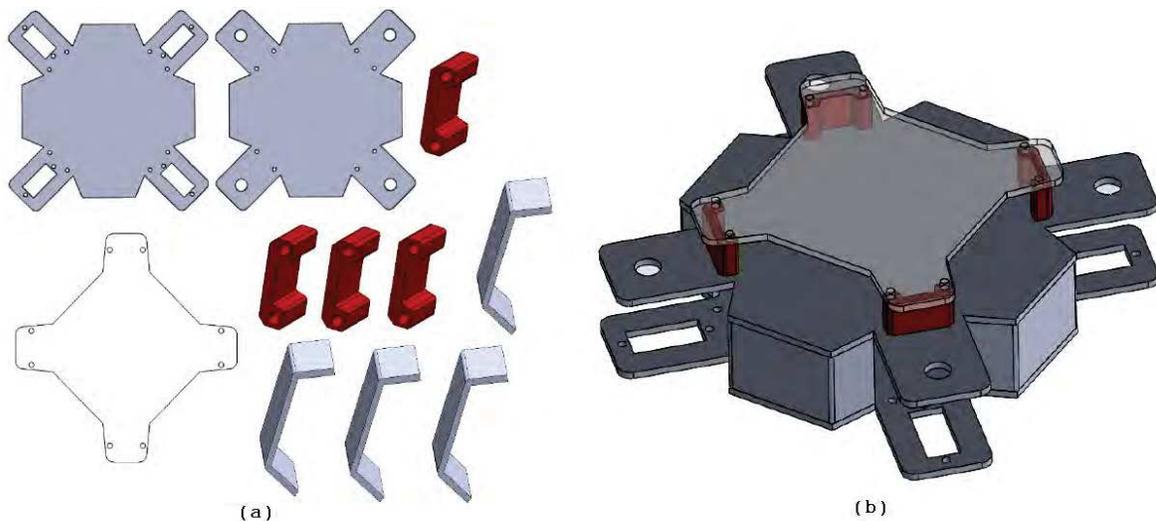


Figura 2. Cuerpo del cuadrúpedo.

En la figura 3 se muestra la tapa superior del cuerpo con sus dimensiones expresadas en milímetros. Con estas dimensiones se busca cumplir con el requerimiento de tamaño del robot, facilitar el libre movimiento de los servomotores y dejar espacio suficiente para ubicar la batería con sus circuitos

de distribución de energía. La tarjeta de control se ubicará sobre el cuerpo del cuadrúpedo, en una base de acrílico para evitar un corto circuito por el aluminio del dibond, con unos soportes se deja un espacio para los cables y conectores en donde se conectarán los servomotores. La tapa inferior tiene las mismas dimensiones que la superior, su diseño cambia ligeramente porque se requiere de un rectángulo en lugar de un círculo, necesario para el soporte de los servomotores.

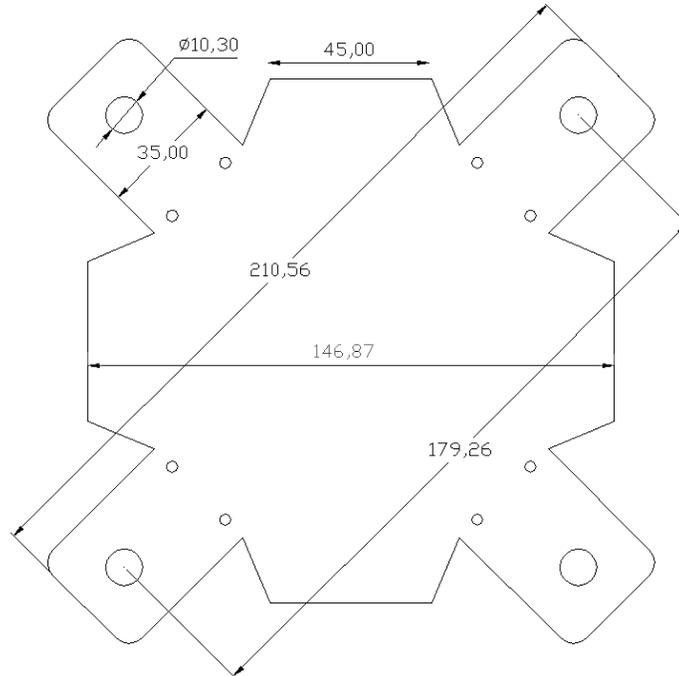


Figura 3. Dimensiones de las tapas del cuerpo del cuadrúpedo.

Las patas o extremidades son los otros elementos importantes en el cuadrúpedo, para su diseño se considera que cada pata tendrá dos servomotores para que el cuadrúpedo se pueda flexionar de arriba hacia abajo. En la figura 4 se exhibe una de las patas, en 4(a) se pueden ver las 8 partes que la conforman y en 4(b) se muestra la forma en que las partes se acoplan para integrar la extremidad. En la figura 5 se muestran las dimensiones en milímetros de las partes laterales de cada pata que soportan a los dos servomotores. El ancho de las patas también queda determinado por el tamaño de los servomotores, su ajuste debe ser preciso para permitir su movilidad.

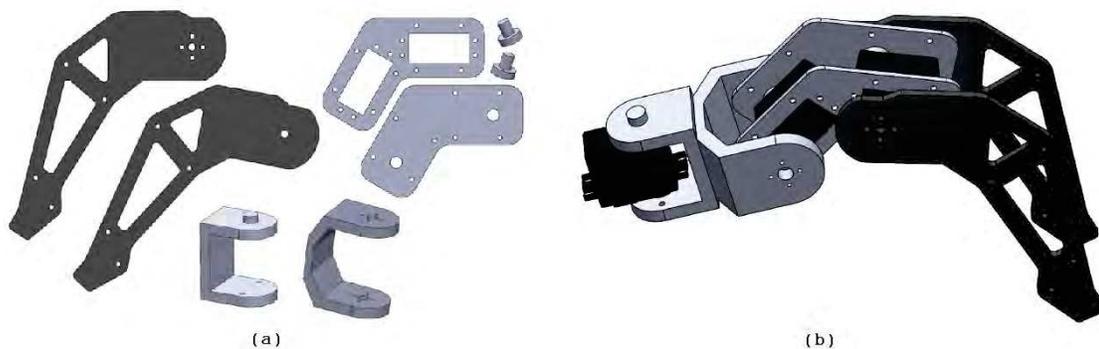


Figura 4. Patas del cuadrúpedo.

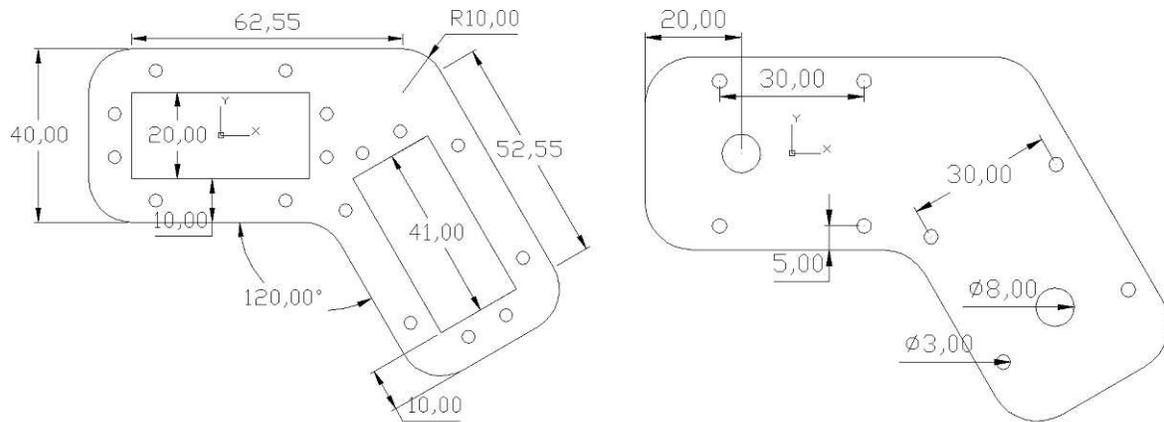


Figura 5. Soporte para los servomotores de las extremidades.

La pieza que une al cuerpo con cada pata se tuvo que separar en dos partes, en el diseño inicial se consideró como una sola, pero al momento de hacer la impresión en 3D no se implementó adecuadamente porque no contaba con un lado plano suficientemente grande que sirviera de soporte. Por ello, el diseño se separó en dos partes y después se unieron por medio de pegamento de alta resistencia.

La unión del cuerpo con las extremidades se muestra en la figura 6 con las dimensiones faltantes en las patas. Sumando las dimensiones se observa que el robot cumple con las especificaciones del tamaño planteadas como requisito inicial.

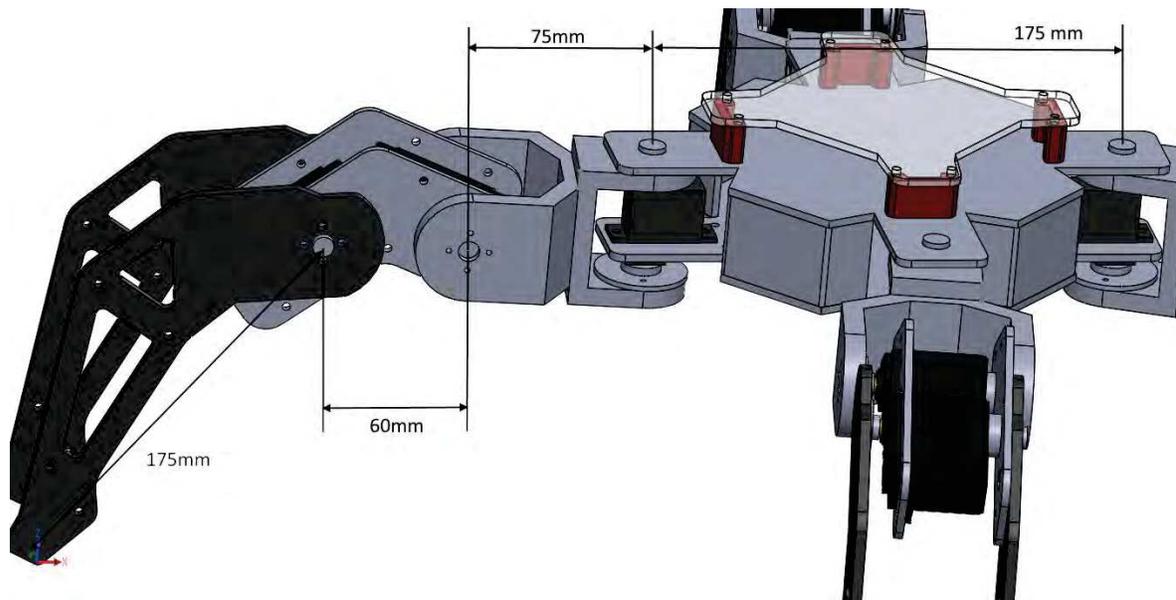


Figura 6. Unión del cuerpo con las extremidades.

2.3 Suministro de energía

Uno de los aspectos más importantes en un robot es el suministro de energía, se debe seleccionar una batería con la capacidad suficiente para el manejo de 12 servomotores. De acuerdo con el fabricante, un servomotor MG995R tiene un consumo de 170 mA en una operación sin carga, lo que conlleva a un consumo de 2.04 A con los 12 motores energizados [6]. Además, se debe considerar la inclusión de una tarjeta de control y algunos periféricos, como el módulo bluetooth y el sensor

ultrasónico. La batería debe permitir que el robot opere en forma independiente por lo menos durante 30 minutos, tiempo suficiente para una exhibición.

Durante esta fase de revisión de requerimientos se consideraron dos alternativas para el suministro de energía: Emplear una batería del tipo Lipo (Litio-Polimero) de 7.4 V con una capacidad de 1800 mA/h o una batería NiCa (Nikel-Cadmio) de 7.2 V y 2000 mA/h. Se optó por la batería LiPo porque tiene dos ventajas importantes, un peso de 110 gramos contra 210 gramos de la NiCa, con un peso menor se reduce el torque exigido a los servomotores, además de que el tiempo de carga para una LiPo es mucho menor que para una NiCa, en menos de 1 hora alcanza su carga máxima mientras que la NiCa requiere entre 6 y 8 horas.

La batería proporciona el voltaje para los diferentes dispositivos electrónicos, la alimentación de la etapa lógica (tarjeta de control, módulo bluetooth y sensor ultrasónico) debe ser independiente para evitar una mala operación debida a las transiciones ocurridas en el arranque de los servomotores. La etapa lógica requiere de un voltaje de alimentación de 5 V, éste se consigue con un regulador lineal con número de serie 7805, un regulador lineal es conveniente por su bajo precio y porque la etapa lógica demanda muy poca corriente.

Por otro lado, los servomotores requieren de un voltaje entre 4.8 V y 6.6 V [6], con un consumo aproximado de corriente de 2 A. Para garantizar la corriente requerida se utilizó un convertidor de CD a CD tipo reductor (buck) de la marca Pololu con número de serie D24V50F5, se trata de un regulador conmutado por lo que tiene una eficiencia entre el 85 % y 95 % en un tamaño reducido, las pérdidas en un regulador lineal son, por lo general, 5 a 10 veces superiores que en uno conmutado. El convertidor D24V50F5 proporciona un voltaje regulado de 5 V a la salida y puede suministrar hasta 5 A, con un voltaje de entrada en el rango de 6 V a 38 V [7]. En la figura 7 se muestra el circuito para el suministro de energía a partir del voltaje de la batería LiPo.

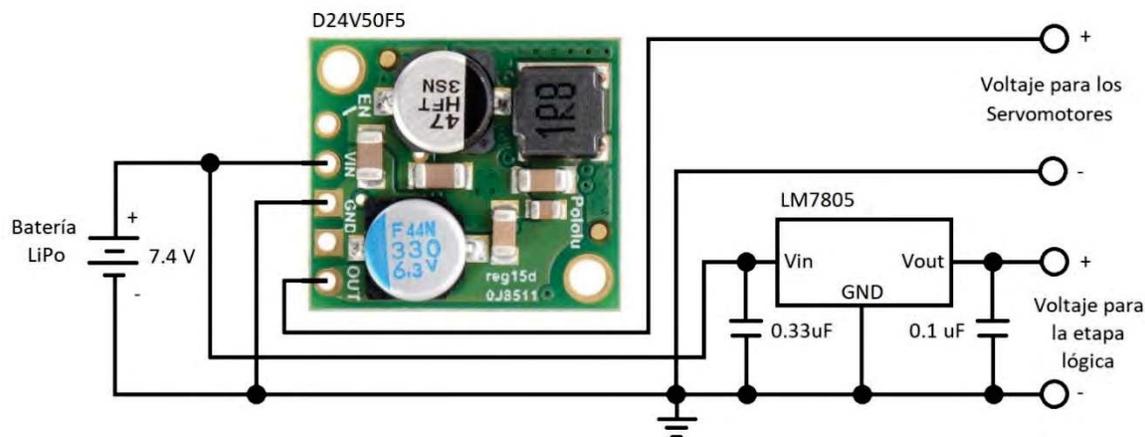


Figura 7. Diagrama de conexiones para el suministro de energía.

2.4 Tarjeta de Control

La tarjeta de control es el elemento central del robot cuadrúpedo, sus tareas principales son: Recepción de comandos por medio de bluetooth, detección de obstáculos a través del sensor ultrasónico y generación de señales PWM para la manipulación de los servomotores. El resultado de estas tareas se ve reflejado en la coordinación de los eslabones del robot para determinar su movimiento. Cada eslabón es manejado por un servomotor y requiere de una señal PWM con un periodo de 20 ms (frecuencia de 50 Hz) para la ubicación de su eje en la posición deseada. En la figura 8 se describe el comportamiento de un servomotor ilustrando 3 posiciones diferentes de su eje, la relación entre el ancho de pulso y la ubicación del eje de salida es lineal, la circuitería interna y un sistema de engranes en cada servo son los responsables de este comportamiento.

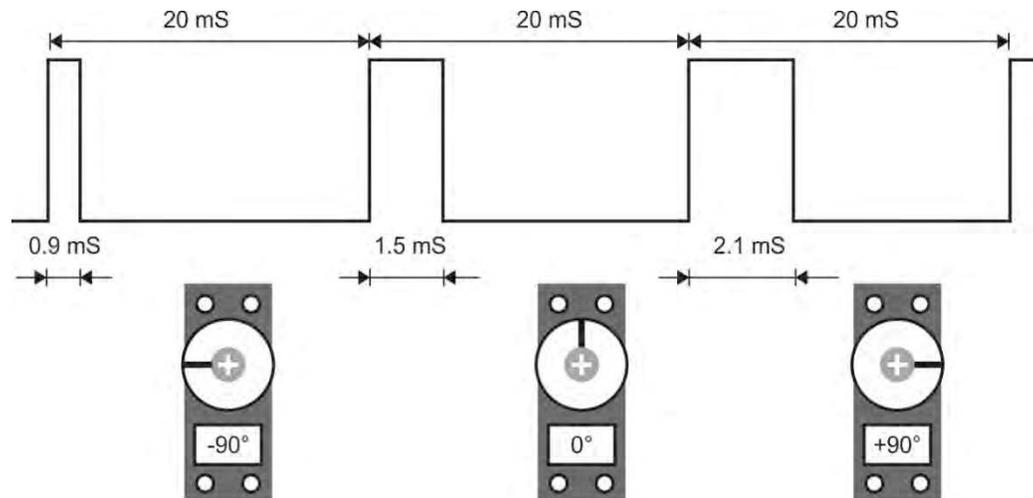


Figura 8. Comportamiento de un servomotor.

Por lo tanto, la tarjeta de control que coordine al cuadrúpedo debe contar con la capacidad de generar por lo menos 12 señales PWM, además de dejar puertos disponibles para la conexión de los otros elementos de la etapa lógica. Para cubrir estos requerimientos se optó por el uso de una tarjeta Arduino Mega, la tarjeta está basada en el CI ATmega2560 que es un microcontrolador de 8 bits perteneciente a la familia AVR. Entre sus principales características la tarjeta contiene 54 entradas/salidas digitales (15 pueden ser usadas como salidas PWM), 16 entradas analógicas, 4 puertos seriales (UART), un cristal oscilador de 16 MHz, un puerto USB para la programación y alimentación, botón de reset y conectores de expansión [8].

En la figura 9 se muestra el diagrama de conexiones del microcontrolador ATmega2560 incluido en la tarjeta Arduino Mega con los diferentes elementos del cuadrúpedo. Se utilizan 12 salidas PWM para el manejo de los servomotores, así como dos de los cuatro puertos seriales disponibles, uno para la programación de la tarjeta y otro para la conexión del módulo bluetooth, que es el medio por donde se reciben los comandos del usuario. Adicionalmente, se utilizan dos terminales digitales para la conexión del sensor ultrasónico.

Las tarjetas Arduino facilitan el desarrollo de proyectos al ocultar las características del microcontrolador (MCU) que les da soporte, para ello, las terminales del MCU se renombran en cada tarjeta con números de acuerdo a su ubicación, de manera que en los programas de Arduino se hace referencia al número y no a la terminal de cada puerto. En la Tabla 1 se listan las terminales empleadas en el proyecto, con su ubicación en la tarjeta Arduino y en el microcontrolador ATmega2560.

3. Manufactura y ensamble

En la figura 10 se ilustra parte del proceso de manufactura y ensamble, se puede ver a la máquina CNC mientras está cortando, algunas piezas ya cortadas y avances en el ensamble del cuadrúpedo. En la medida en que se fueron colocando los servomotores también se atornillaron las partes para mantener una estructura estable. Dado que los servomotores sólo tienen la libertad de mover su eje en un rango de -90° a $+90^\circ$, se buscó una posición inicial en la que el servomotor estuviera lo más cercano a su punto medio (0°) para contar con el mismo margen de movimiento en ambas direcciones.

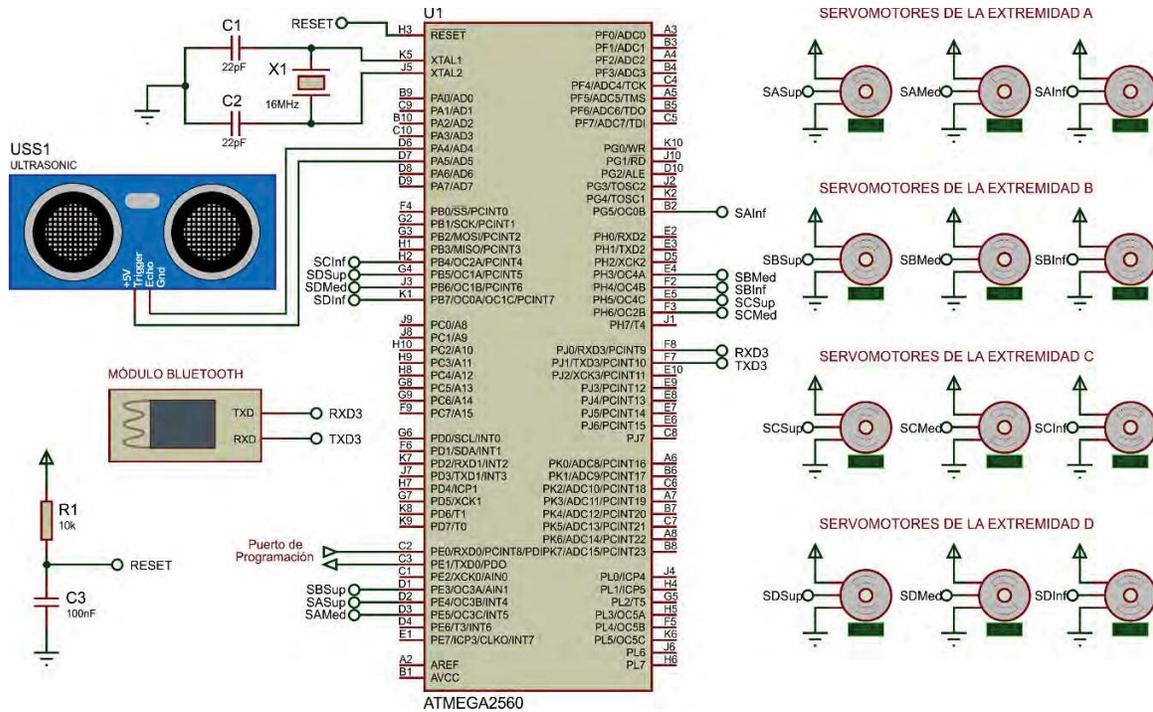


Figura 9. Diagrama eléctrico del robot cuadrúpedo.

Tabla 1. Terminales de la tarjeta Arduino Mega empleadas en el robot cuadrúpedo.

Terminal		Uso en el robot cuadrúpedo
Arduino	MCU	
0 y 1	PE0 y PE1	Puerto Serie 0, para la programación de la tarjeta
2	PE4	Servomotor superior de la extremidad A
3	PE5	Servomotor medio de la extremidad A
4	PG5	Servomotor inferior de la extremidad A
5	PE3	Servomotor superior de la extremidad B
6	PH3	Servomotor medio de la extremidad B
7	PH4	Servomotor inferior de la extremidad B
8	PH5	Servomotor superior de la extremidad C
9	PH6	Servomotor medio de la extremidad C
10	PB4	Servomotor inferior de la extremidad C
11	PB5	Servomotor superior de la extremidad D
12	PB6	Servomotor medio de la extremidad D
13	PB7	Servomotor inferior de la extremidad D
14 y 15	PJ0 y PJ1	Puerto Serie 3, para el módulo bluetooth
26	PA4	Eco del sensor ultrasónico
27	PA5	Disparo del sensor ultrasónico



Figura 10. Ensamble del cuadrúpedo.

En forma práctica no es posible conseguir una posición exacta en el punto medio en todos los ejes, esto no afecta porque los eslabones del cuadrúpedo no requieren de movimientos de 90° , por software se define la posición inicial y a partir de ella se realizan los desplazamientos necesarios.

En la figura 11 se puede ver al cuadrúpedo ensamblado, la batería queda dentro del cuerpo del robot y la tarjeta Arduino Mega sobre el mismo. También fue necesario el diseño e impresión en 3D de una base para la ubicación del sensor ultrasónico y para el módulo receptor por bluetooth.

4. Diseño de Programas

El cuadrúpedo requiere de dos elementos de software, un programa para la tarjeta Arduino Mega y una aplicación para un dispositivo móvil con SO Android que sirve como interfaz de usuario. Los dos programas se enlazan a través de bluetooth, por este medio se envía un comando cuando el usuario presiona uno de los botones de la aplicación del móvil, el programa de la tarjeta Arduino recibe e identifica al comando para darle respuesta. En los siguientes apartados se proporciona más información de ambos programas.

4.1 Programa para la tarjeta Arduino

El diagrama de flujo de la figura 12 ilustra el comportamiento del programa para la tarjeta Arduino, se observa que el cuadrúpedo tiene dos modos de operación, el usuario puede cambiar el modo a partir del envío de un comando que el microcontrolador recibe en su puerto serie, un comando básicamente es un carácter ASCII preestablecido.

Después de que el cuadrúpedo se energiza es llevado a una posición inicial e inicia en el modo de operación manual descrito en la figura 13, en este modo el programa recibe comandos que identifica para que el cuadrúpedo realice el movimiento correspondiente. Los movimientos se realizan a través de funciones que se deben completar para que quede en una posición que le permita iniciar con la misma o una secuencia diferente. El ciclo repetitivo del programa principal hace que el movimiento sea continuo hasta que se reciba un comando que corresponda con un movimiento diferente o al comando 'E' para detener el movimiento.



Figura 11. Cuadrúpedo ensamblado.

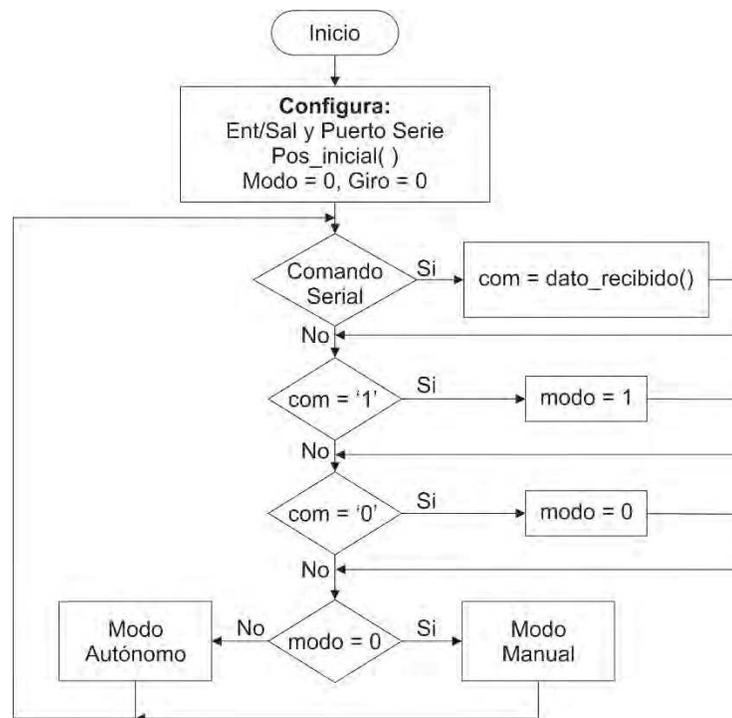


Figura 12. Programa principal en la Arduino Mega.

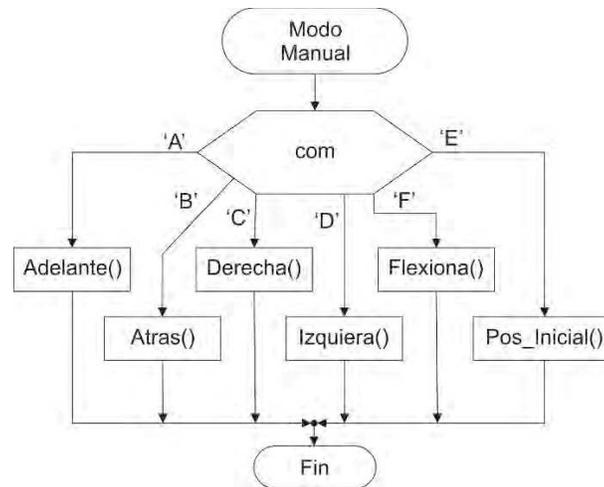


Figura 13. Modo de operación manual del cuadrúpedo.

La operación en modo autónomo hace que el cuadrúpedo avance de manera continua hacia adelante y en caso de detectar un obstáculo, por medio del sensor ultrasónico, el cuadrúpedo hará una rutina evasiva, girando a alguno de los lados, la dirección del giro se va alternando con el apoyo de una variable, un giro es a la derecha y el próximo a la izquierda, en la figura 14 se describe el comportamiento del modo autónomo para la operación del cuadrúpedo.

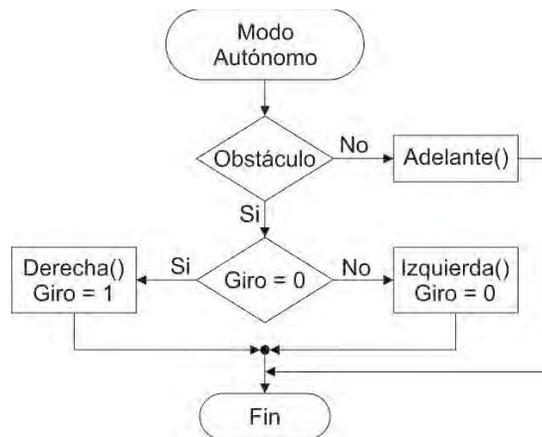


Figura 14. Modo autónomo para la operación del cuadrúpedo.

En cada función se coordina la posición de los servomotores, dejando intervalos de tiempo para su establecimiento, considerando que tienen una velocidad de operación de 60 grados en 200 ms [6]. Las funciones de movimiento se obtuvieron de manera práctica, primero se determinó la posición inicial y a partir de ahí se realizaron desplazamientos en los servomotores evaluando con pequeños incrementos, aplicando la estrategia de ajustar, programar y evaluar el desempeño, repetidamente hasta alcanzar resultados satisfactorios.

Como un ejemplo, en la figura 15 se muestra el código de la función *adelante()*, se observa que se realizan movimientos cruzados, primero las patas A y C, luego las patas B y D, y finalmente se deja a los servomotores en la posición inicial. Las funciones de Arduino para el manejo de los servomotores utilizan un número de 8 bits para determinar su posición, es decir, para establecer el ancho de pulso de una salida PWM, se debe escribir un número en el intervalo entre 0 y 255.

```
// Movimiento hacia Adelante
void adelante()
{
  moverAC();
  moverBD();
  pos_inicial();
  delay(tim);
}

void moverAC()
{
  servoA2.write(A2i + 20);
  servoC2.write(C2i + 20);
  servoA1.write(A1i + 15);
  servoC1.write(C1i - 15);
  delay(tim);
  servoA2.write(A2i + 10);
  servoC2.write(C2i + 10);
  delay(tim);
}

void moverBD()
{
  servoD2.write(D2i + 20);
  servoB2.write(B2i + 20);
  servoD1.write(D1i - 15);
  servoB1.write(B1i + 15);
  delay(tim);
  servoD2.write(D2i + 10);
  servoB2.write(B2i + 10);
  delay(tim);
}
```

Figura 15. Código para mover al cuadrúpedo hacia adelante.

4.2 Aplicación para el envío de comandos

La aplicación para el dispositivo móvil se realizó empleando el ambiente de programación visual denominado App Inventor, una herramienta muy intuitiva desarrollada por el MIT con la que se pueden construir aplicaciones completamente funcionales para teléfonos inteligentes y tabletas con sistema operativo Android [2].

Un programa en App Inventor se realiza en dos etapas, primero debe definirse la interfaz de usuario con los elementos o recursos requeridos por la aplicación y posteriormente se establecen las acciones a realizar cuando ocurre un evento en un elemento de la interfaz o en un recurso de la aplicación.

La App que se desarrolló sólo requiere de un recurso para la conectividad a través de bluetooth y 8 botones para el envío de comandos al cuadrúpedo, aunque también es necesaria una lista de selección para identificar los dispositivos bluetooth disponibles. En la figura 16(a) se muestra el diseño de la interfaz de usuario en la herramienta de App Inventor y la figura 16(b) presenta una pantalla de la App en ejecución en un teléfono inteligente.

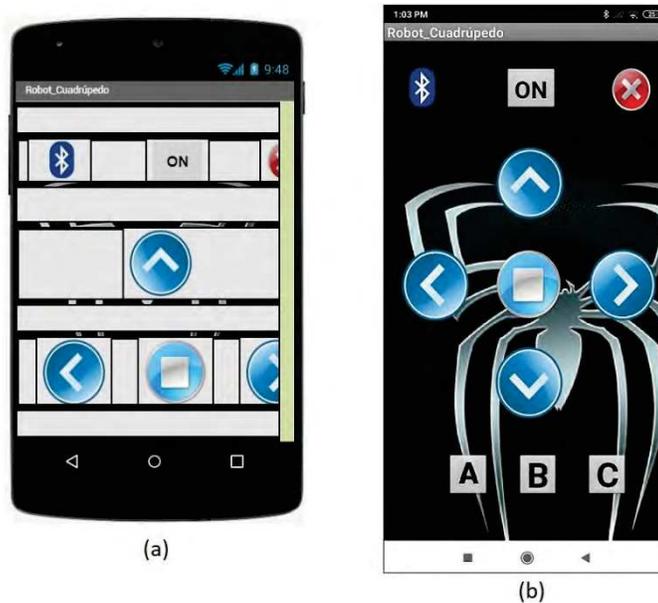


Figura 16. Interfaz de la App para la manipulación del cuadrúpedo.

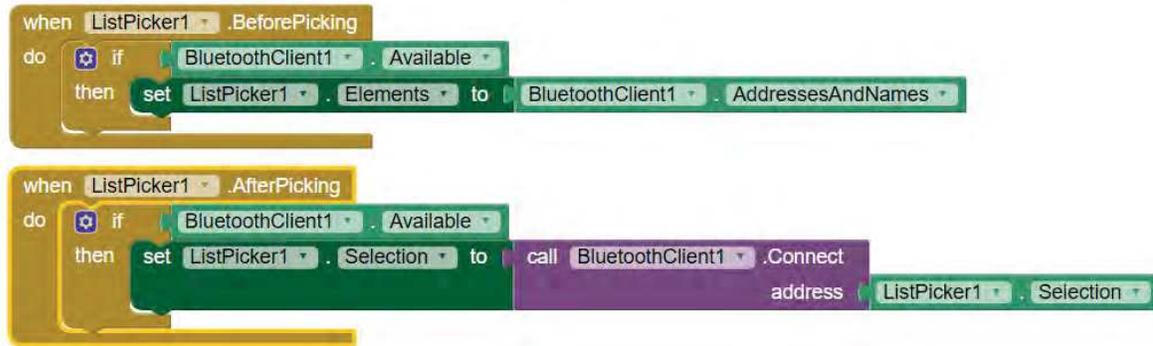


Figura 17. Acciones para activar la conexión por bluetooth con el cuadrúpedo.

En la segunda etapa del desarrollo de una aplicación se determinan las acciones a realizar cuando ocurren eventos en los elementos de la interfaz, para ello, la herramienta de App Inventor proporciona un conjunto de bloques de programación propios para cada elemento, así como bloques genéricos de programación, como las estructuras de decisión. En lo que respecta a la conectividad por bluetooth, la lista de selección es quien genera los eventos, en la figura 17 se describe cómo se realizan estas acciones, se distinguen dos momentos, previo a la elección del usuario se obtiene la lista de dispositivos vinculados y posterior a la elección, la App se conecta con el dispositivo elegido, en ambos casos, una estructura condicional garantiza la presencia del servicio de bluetooth.

Para la atención de cada botón también se deben especificar las acciones correspondientes, a manera de ejemplo, en la figura 18 se muestran las tareas a realizar cuando son presionados los botones 1 y 2, básicamente se envían caracteres ASCII por la interfaz bluetooth, garantizando previamente la presencia de la conexión por bluetooth. La atención a los otros botones se realiza con segmentos de código muy similares.

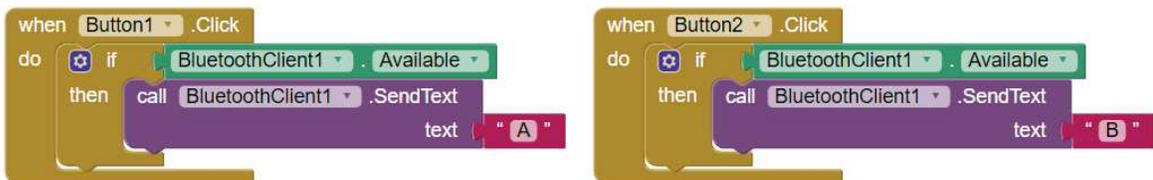


Figura 18. Atención a los botones 1 y 2 de la interfaz de la aplicación.

5. Resultados

El desempeño del cuadrúpedo es favorable y cumple con el propósito para el cual se elaboró, que es la difusión de las Ingenierías en Electrónica y Mecatrónica de la UTM. El peso total del cuadrúpedo es de 1.5 kg, aunque no es muy alto si es considerable al realizar los movimientos. En la tabla 2 se muestra el costo de los elementos principales, para estimar el costo total del cuadrúpedo, las impresiones en 3D y la manufactura en dibond se realizaron en los talleres de la institución, pero se solicitó el precio a los negocios relacionados para aproximar el costo total del proyecto en pesos mexicanos.

Como parte de los resultados, en la figura 19 se muestra la exhibición del cuadrúpedo en la Secundaria General “Telésforo Mendoza”, de la ciudad de Huajuapán de León, a los alumnos de 3er año se les impartió una plática sobre robótica, para motivarlos y sembrar la idea de estudiar alguna de las ingenierías que se imparten en la UTM.

Tabla 2. Relación de costos de los elementos del cuadrúpedo.

Parte	Cantidad	Costo unitario	Suma
Servomotor MG995R	12	\$ 120.00	\$ 1 440.00
Batería LIPO 1800 mAh	1	\$ 450.00	\$ 450.00
Regulador De Voltaje D24v50f5 5v, 5a	1	\$ 400.00	\$ 400.00
Regulador de voltaje LM7805	1	\$ 5.00	\$ 5.00
Tarjeta Arduino Mega 2560	1	\$ 250.00	\$ 250.00
Sensor ultrasónico HC-SR04	1	\$ 27.00	\$ 27.00
Módulo bluetooth HC-06	1	\$ 74.00	\$ 74.00
Terminal con 6 tornillos	4	\$12.00	\$ 48.00
Hoja de dibond	1/8	\$ 2 400.00	\$ 300.00
Impresiones 3D para conectar las patas	8	\$ 70.00	\$ 560.00
Impresiones 3D complementarias	9	\$ 40.00	\$ 360.00
Total:			\$ 3 914.00



Figura 19. Exhibición del cuadrúpedo ante alumnos de 3er año de secundaria.

El cuadrúpedo también fue presentado con éxito en la expo-robótica realizada en la ciudad de Tehuacán, Puebla, como parte de las actividades del 10° aniversario de la Universidad Tecnológica de Tehuacán (UTT), el 13 de septiembre de 2019 (figura 20). Actualmente está disponible en el Instituto de Electrónica y Mecatrónica para su exhibición ante las diferentes visitas de estudiantes interesados en informarse sobre las Ingenierías en Electrónica y Mecatrónica.



Figura 20. Exhibición del cuadrúpedo en una expo-robótica realizada por la UTT.

6. Conclusiones

El desarrollo del cuadrúpedo dejó una amplia experiencia y satisfacción de carácter técnico y personal. En lo técnico, es interesante observar el desempeño de un sistema mecatrónico en el que se coordinan 12 servomotores para proporcionar el movimiento del robot, comandados desde una tarjeta arduino, cuyo comportamiento es determinado por medio de un programa. En lo personal, el cuadrúpedo se ha exhibido en diferentes audiencias, su estructura y operación captan la atención de la gente, con lo que se cumple el propósito de contar con un robot para la difusión de las Ingenierías en Electrónica y Mecatrónica, a un costo relativamente bajo.

El peso del cuadrúpedo (1.5 kg) dificultó la elaboración del algoritmo de la caminata, se intentó emplear la cinemática inversa pero la estructura del robot dificulta contar con mediciones exactas para el establecimiento del algoritmo, los resultados se han ido mejorando conforme se va adquiriendo más experiencia en el desempeño real del cuadrúpedo.

Aunque el cuadrúpedo fue realizado con propósitos de difusión, la estructura mecánica queda disponible para ser manipulada con otras plataformas para sistemas embebidos, como tarjetas basadas en arreglos de compuertas programables (FPGA) o tarjetas basadas en procesadores, como la Raspberry Pi.

Referencias

- [1] *Robot NAO – Aliverobots*, <https://www.aliverobots.com/nao/>, 2015, última visita en septiembre de 2020.
- [2] *MIT APP Inventor | Explore MIT APP Inventor*, <https://www.appinventor.mit.edu/>, 2019, última visita en septiembre de 2020.
- [3] Bustos D., Torres J. C. y Turrubiartes A. I. (2011). *Robot Cuadrúpedo Diseño, Construcción y Control*, 10º Congreso Nacional de Mecatrónica, Asociación Mexicana de Mecatrónica A.C., páginas 145-150, Puerto Vallarta, Jalisco.
- [4] Miret-Rubio J. D. (2017). *Diseño e implementación de un robot cuadrúpedo 3GDL con microcontrolador de 32bits STM*, Tesis de maestría, Universidad Politécnica de Valencia.
- [5] Prado-Navarrete A. (2012). *“PROYECTO AT-AS” Diseño de un robot cuadrúpedo imprimible, fabricación e implementación de la locomoción*; Trabajo de tesis, Universidad Carlos III de Madrid.



- [6] *MG995 | TowerPro* (2018), <https://www.towerpro.com.tw/product/mg995/>, 2015, última visita en septiembre de 2020.
- [7] *Pololu 5V, 5A Step-Down Voltage Regulator D24V50F5*, <https://www.pololu.com/product/2851>, 2015, última visita en septiembre de 2020.
- [8] *Arduino Mega 2560 Rev3*, <https://store.arduino.cc/usa/mega-2560-r3>, 2011, última visita en septiembre de 2020.



Verticallium: Solución Integral para el Cultivo Vertical de Ajo Usando Compost Aplicando IoT

Castillo Javier¹, Grisales Andres Felipe², Martinez Ortega Sara Virginia³, Ortega Solarte Leidy Dayanna⁴, Ramirez Preciado Leider⁵

Universidad Santiago de Cali, Ingeniería Electrónica
Grupo de Investigación en Ingeniería Electrónica Industrial y Ambiental - GIEIAM
Email: [1javier.castillo00, 2andres.grisales00, 3sara.martinez01, 4leidy.ortega03, 5leider.ramirez00]@usc.edu.co
Santiago de Cali, Colombia

Resumen

La agricultura vertical ha tenido gran apogeo en los países de mayor desarrollo, sin embargo no ha sido posible llevarlo a mayores escalas debido a que cambios menores en el entorno logran producir un gran impacto en la productividad y la calidad de la actividad agrícola. En este documento se presenta el proceso de desarrollo del proyecto Verticallium, el cual tiene como propósito permitir el cultivo de ajo automatizado en zonas urbanas aplicando compost como material nutritivo y añadiendo un sistema de monitoreo para realizar un seguimiento de las condiciones físicas del cultivo. En este sistema se emplearán diferentes sensores para detectar estas condiciones y esa información será llevada al sitio web oficial del proyecto haciendo uso del internet de las cosas (IoT). Por otro lado el sistema proporciona la función remota donde los usuarios pueden variar las condiciones de cultivo para encender/apagar los diferentes actuadores del proceso. Los elementos electrónicos serán alimentados por un sistema de paneles solares para asegurar el auto sostenimiento del proyecto promoviendo las energías renovables.

Palabras clave: Compost, Cultivo Orgánico, Energías Renovables, Huerta Vertical, IoT

1. Introducción

El Índice Global de Seguridad Alimentaria, es una valoración integral que se basa en la posibilidad del acceso de las personas a la comida e incluye temas como que los productores no pasen dificultades para cosechar, y que en las grandes ciudades sus habitantes puedan acceder a ella. En este estudio Colombia ocupa el puesto 43 [1] de 113, evidenciándose un bajísimo porcentaje de inversión pública en temas de Investigación y Desarrollo agrícola, lo cual se ve traducido en procesos productivos ineficientes, donde la intervención humana forma parte activa de éstos. Sin embargo, la ubicación geográfica del país lo deja muy bien parado con una calificación significativa en temas de diversidad alimentaria con buena disponibilidad y volúmenes constantes.

El momento coyuntural que está viviendo el mundo debido al coronavirus (SARS-CoV 2) ha hecho que los sistemas alimentarios se transformen, impactando de manera directa a todos los eslabones de la cadena de suministro, iniciando en los sistemas productivos hasta la composición dietaria de las personas. El confinamiento ha afectado en mayor proporción a la población urbana en situación de pobreza, ya que no solo tienen dificultades para movilizarse y abastecerse de alimento, sino que además no han podido desempeñar sus ocupaciones en el ámbito laboral.

Las Escuelas Rurales Alternativas (ERA) han tomado gran vocería para fomentar la creación de huertas familiares (en gran medida son orgánicas) que puedan sustentar dietas saludables, aún cuando los cultivos básicos se ven afectados por múltiples contrariedades, incluido el cambio climático. Estos huertos han promovido la creación de redes comunitarias donde las personas intercambian sus productos y además generan un intercambio de conocimiento. “En estas plataformas



se desarrollan proyectos productivos articulando todos los factores necesarios para hacerlos económicamente viables y sostenibles: tierra, conocimiento, capital, trabajo y mercados” [3].

Actualmente, en el mundo hay un movimiento creciente por el consumo de alimentos orgánicos, debido a que las personas están continuamente preguntándose por el tipo y procedencia de alimentos que consumen para mejorar su bienestar [4]. Colombia siendo país agrícola, le ha tomado tiempo hacer la transición a alimentos orgánicos por falta de apoyos gubernamentales, aún así, se ha dado una reducción en el uso de pesticidas y químicos en los alimentos, pero el 95 % de los alimentos orgánicos son exportados [5], por lo tanto, el 5 % que queda es de un alto costo para la comunidad [4].

El ajo es un producto que hace parte de la gastronomía colombiana y que tiene múltiples beneficios para la salud por lo que tiene una alta demanda en los hogares colombianos, pero así como su demanda crece, ha crecido su importación, lo que pone en desventaja a los cultivadores nacionales, el cual no ha tenido crecimiento a la producción de hace una década (364 hectáreas) [7]. El ajo colombiano se enfrenta con la alta importación del ajo chino [6]. Por lo tanto los productores nacionales han tenido que mejorar en la producción y la innovación, para poder competir y mantener la producción, aunque la producción de una tonelada de ajo colombiano puede costar el doble que la importación del ajo chino; pero el ajo colombiano es de mejor calidad porque el ajo chino es modificado genéticamente para esterilizarlo.

El desarrollo de este proyecto tiene como objetivo general desarrollar un proceso controlado y autosostenible que permita el cultivo de ajo orgánico en los hogares de la ciudad de Santiago de Cali (Colombia) con el fin de maximizar la cantidad de plantas optimizando el espacio.

2. Contextualización del Cultivo de Ajo en Colombia

El ajo (*Allium sativum*) es una hortaliza que tiene grandes cualidades alimenticias medicinales, debido a que contiene 4,7% de proteína y una gran cantidad de minerales, además de ser antiséptico, depurativo, estimulante del apetito y es activador de las funciones digestivas, por lo tanto es un ingrediente importante en las cocinas de muchos países. En Colombia, las zonas productoras de ajo son Boyacá, Cundinamarca, Nariño, Cauca y el Valle del Cauca [7].

Los ajos se pueden clasificar según el color de túnica o por sus requerimientos de fotoperíodo. Colombia tiene 2 variedades de ajo según el color de su túnica: Blanco o Rojo. El Blanco es el más común, más sensible al ataque de enfermedades foliares, pero mejor para almacenamiento y tienen una hoja relativamente ancha. El ajo blanco presenta unas clasificaciones:

- Blanco Criollo (Chocontano): Presenta 12 dientes aunque puede variar de 10 a 20. Es altamente aromático pero susceptible a enfermedades como *Alternaria* y nemátodos.
- Blanco Pastuso: Da de 20 a 30 dientes, de formas irregulares con una túnica delgada. Es susceptible a enfermedades como *Sclerotium* y *Alternaria*.
- Blanco Ancho o Gigante: Presenta de 4 a 10 dientes, puede llegar a pesar 30 gramos.

El conocido ajo Rojo puede tener túnicas rosadas o moradas. Este ajo es menos resistente al almacenamiento. Dentro de su clasificación, se encuentran 2 tipos:

- Rojo Criollo: Tiene la misma forma, número de dientes y aroma que el ajo blanco criollo, aunque es mucho más resistente a las enfermedades, pero como su túnica es delgada no resiste el almacenamiento.
- Rosado Peruano: Tiene bulbos grandes y uniformes aunque su aroma es menos penetrante que el ajo rojo criollo. Este tipo de ajo presenta una alta tolerancia al encebollamiento.

Según el requerimiento de fotoperíodo, en Colombia está adaptado mejor por las condiciones del país el de día corto que es de 11 a 12 horas de luz/día, aunque los dientes son pequeños y menos



uniformes, cumplen mejor las condiciones de calidad. Los que son de día largo, a temperaturas altas y más horas de luz/día presentan bulbos grandes y uniformes.

Según el Ministerio de Agricultura y desarrollo rural, en Colombia se sembraron 375 hectáreas de ajo, con una producción de 4300 toneladas, presentando un rendimiento del 11.5% ton/ha. Los departamentos que más producción tuvieron fueron Santander, Cundinamarca y Boyacá con 234 ha, 70 ha y 35 ha, respectivamente. Aún así la importación del ajo desde países como Estados Unidos, India y China, desplaza la comercialización del ajo colombiano en el mercado y ha hecho muy difícil para los cultivadores colombianos competir; por lo que se han visto caídas en la producción, por ejemplo en 2014 la productividad promedio fue de 18,2 toneladas por hectárea y para 2017 se redujo a 14,1 toneladas por hectárea[6]. El ajo es un cultivo de alta demanda insatisfecha dentro de Colombia, y la producción de una tonelada de ajo puede costar el doble que la importación del ajo chino, el cual es un ajo que viene modificado genéticamente por medio de radiación para que no pueda reproducirse [6].

Debido a la situación actual de la pandemia debido al Covid-19, se ha dejado de importar materias primas y material importado de China, cerca de diez mil contenedores no ha podido llegar al puerto de Buenaventura; y se ha dado una reducción del ajo en el mercado lo que ha hecho crecer su precio en un 500% [8].

El consumo de alimentos orgánicos ha tenido un crecimiento del 80% desde el 2014 a la fecha, debido a que las personas buscan cada vez productos más sanos[4], alejados de químicos, conservantes o productos industrializados [5], además que intentan consumir local, lo que ha hecho que la comercialización del ajo colombiano se mantenga en el mercado a pesar de toda la competencia que lo afecta [6]. Aunque algo que afecta los precios es que el 95% de los productos orgánicos producidos en Colombia son exportados y la producción que queda para las mesas colombianas se encarece al rededor de un 10%. Además hay barreras como licencias y certificaciones de los productos orgánicos que ha hecho una alza en los precios de producción que desestimula a los agricultores y empresarios.

3. Necesidades Agroecológicas del Ajo

Colombia tiene buenas zonas productoras de ajo, como lo son el departamento de Boyacá, Cundinamarca, Nariño, Cauca y Valle del Cauca; debido a que en Colombia, el ajo se adapta a los 1700 a 3000 m.s.n.m., siempre que las temperaturas sean menores a 20°C durante las primeras etapas de crecimiento [6]. El clima ideal es seco, debido a que la humedad favorece la pudrición del bulbo y otro tipos de enfermedades; y su temperatura óptima es de 12 a 18 °C.

Para un buen cultivo del ajo es necesario a parte de la temperatura tener en cuenta el tipo de suelo, los nutrientes necesarios, el tipo de riego y ciertos procesos a la hora de la cosecha y postcosecha[7].

- Suelo: Deben ser suelos sueltos y con buen contenido de materia orgánica que permita un buen drenaje y su estructura retenga bien la humedad. Los suelos franco-arcillosos permiten un buen desarrollo del bulbo, en cambio los suelos pesados dificultan la formación de las cabezas y favorecen la pudrición. Se recomienda que el pH del suelo esté entre 5.5 y 6.8, ácidos a neutros porque la planta no tolera la alcalinidad.
- Nutrientes: Las necesidades de nutrientes se suelen determinar por el análisis del suelo pero hay algunos parámetros generales como la aplicación de materia orgánica que mejora las condiciones del suelo, aunque se debe tener bien descompuesta porque si no incide en las pudriciones del bulbo. El fósforo (P) es un nutriente importante en la formación y tamaño de los bulbos, además para un buen desarrollo debe haber una buena cantidad de potasio (K). El nitrógeno (N) es importante para el área fotosintética pero su exceso ocasiona sobretamamiento



o encebollamiento. La proporción adecuada debe ser N:P:K de 1:3:1. El Boro, Zinc y Magnesio pueden ser aplicados pero en menores proporciones.

- El mejor momento para aplicar estos nutrientes son en la siembra y cuando comienza la formación del bulbo que se da aproximadamente a las 6 semanas. Luego de este periodo el ajo no responde.
- Riego: No se debe dejar en ningún momento deficiente de agua, sobretodo en las etapas de la formación del bulbo. Se recomienda modo aspersión, aunque es más beneficioso la irrigación directa al suelo para disminuir problemas foliares. Faltando 15 a 20 días antes de la cosecha se suspende el riego para dejar secar el suelo porque sino se da la posibilidad de que el bulbo se pudra, además de que pierde color, firmeza y calidad.
- Cosecha: Se da a los 5 o 6 meses luego de la siembra. Debe coincidir con la época de verano por lo que el terreno debe estar seco y los tallos deben tener un color rojizo y se han doblado. Las plantas se deben arrancar a mano y se dejan secar de 3-10 días. El bulbo se protege con las hojas y tallos para evitar una quemazón del sol [7]. En el caso de que haya lluvias, se debe hacer bajo techo.
- Pos-Cosecha: Mientras se secan los bulbos, se deben voltear diariamente. Luego se clasifican según el mercado o si se utilizarán como semilla. Se pueden almacenar por varios meses a 0°C y 65% de humedad. Para los mercados se puede remover los tallos de 2-3 cm de altura y las raíces a 1-2 cm, o se pueden amarrar en atados dejando tanto tallos como raíces. Luego se empacan en cajas de plástico o madera, o en sacos de fique o polipropileno que permite la aireación. Se pueden vender inmediatamente o almacenar. Si se utilizaran como semillas, se deja en reposo de 2 a 3 meses en climas fríos y de 1 a 2 meses en climas cálidos. Sin el reposo el porcentaje de germinación es bajo y tardío, también se dan bulbos desuniformes o muchos dientes se pudren.

4. Agricultura Vertical

Las huertas urbanas han nacido como una necesidad a la crisis alimentaria y contribución al medio ambiente en las ciudades, se estima que 800 millones de personas en el mundo están involucradas y con esto se suplen hasta el 20 % de las necesidades alimentarias [9]. En Santiago de Cali, el barrio San Antonio fue reconocido por el gobierno de Francia como Eco-barrio de Latinoamérica porque se realizaron 80 huertas urbanas, ochenta composteras domesticas y dos pacas digestoras, ademas crecieron los mercados orgánicos [9].

Para entender la relevancia de los cultivos urbanos es necesario hacer una revisión histórica, donde se pueda evidenciar su origen, su evolución y sus potencialidades. Es destacable que la agricultura urbana ha tenido su apogeo en tiempos de crisis tanto económicas como energéticas, que han llevado a su implementación para asegurar el abastecimiento y la seguridad alimentaria.

Sin embargo, esta práctica data del año 600 A.C. [10] en la antigua Mesopotamia, en donde Los Jardines Colgantes de Babilonia eran los protagonistas de esta técnica. La monumental construcción en las alturas contaba con grandes terrazas ajardinadas y fueron construidas durante el reinado de Nabucodonosor II. Su complejo sistema hidráulico tomaba el agua del río Éufrates y la subía por columnas huecas a través de importantes norias movidas por animales o esclavos, esto permitía el riego, la humidificación y el embellecimiento del lugar [11] y lo convirtió en la única maravilla del mundo antiguo que contó con cuestiones ecológicas y orgánicas [12] haciendo un mayor aprovechamiento de los recursos. A pesar de que los jardines tenían en esencia un objetivo ornamental se encontraban plantaciones de palmeras y árboles frutales, como el dátil y los cocos. Por otro lado, en la antigua China en el siglo I A.C. se hacen avances tecnológicos con fines agrícolas, tal como fue el uso del martinete hidráulico [13]. Su principal función era la trilla y el pulido de granos, actividad que anteriormente era manual. Después en el siglo I D.C. emplean la bomba de cadena para el suministro de agua por las tuberías urbanas y sobre todo para hacer el riego en los cultivos.



Avanzando en la línea del tiempo se encuentra que a lo largo de la edad media en Europa surgen importantes innovaciones tecnológicas que aportan algunos elementos positivos al trabajo de los campesinos. Se destaca el cambio del buey por el caballo como animal de tiro que deja como resultado el desarrollo de la herradura y la collera, elementos que permitían al caballo tirar mayores cargas fácilmente. En zonas fértiles se introdujo la rotación de cultivos de tres hojas, lo que reducía significativamente la necesidad de barbecho, aumentando la producción y haciendo que fuera más diversificada aumentando a su vez la productividad, ya que este cambio en las rotaciones permitió la fijación de nitrógeno [14]. En esta época fueron mejorados los molinos de agua que habían sido introducidos por los romanos y junto a los molinos de viento se utilizaron para moler granos, cortar madera y producir textiles [15].

Durante la revolución industrial las ciudades principales crecieron con gran afán para albergar la llegada de los trabajadores provenientes de las áreas rurales. Este desplazamiento pone en evidencia la insuficiencia de producción agrícola que logre abastecer la demanda en las ciudades. Desde este momento los huertos aparecen como una herramienta de primera mano de las estrategias de subsistencia en momentos de crisis. Ahora bien, en la primera mitad del siglo XX la historia de estos huertos está ligada a las grandes guerras, tiempo durante el cual las ciudades tuvieron que adaptarse a la falta de medios e introducir nuevos procesos productivos para abastecerse de bienes de primera necesidad. “Durante la Segunda Guerra Mundial comienza a usarse esta forma de cultivo en las ciudades y se llegan a consumir hasta un 40 % de alimentos procedentes de los huertos urbanos (War Gardens). Se habían convertido en indispensables ya que durante las dos guerras muchos países europeos no podían depender de las importaciones y había que asegurarse el alimento” [16].

Esta tendencia ha crecido a gran escala que hoy por hoy es posible generar los ambientes propicios para el cultivo de diferentes plantas y vegetales incluso sin ser típicos de esta región. Tal es el caso de Japón, que a pesar de ser un país tan pequeño y con escasez de tierra cultivable, ha logrado crear unos huertos automatizados que permiten cultivar sin tierra y aún así tener cultivos de alta calidad para consumo humano y además garantiza una producción durante todo el año. Esta técnica de cultivo sin tierra se conoce como agricultura hidropónica y ha revolucionado la agricultura con el desarrollo de unas “películas de polímeros transparentes hechas a base de un hidrogel permeable, que ayudan a almacenar líquidos y nutrientes, donde las plantas pueden crecer sin necesidad de tierra. Además de permitir que la fruta y verdura crezca en cualquier entorno, la técnica consume un 90 % menos de agua que la agricultura tradicional y no utiliza pesticidas, ya que los poros del polímero bloquean el paso de virus y bacterias” [17].

En los múltiples avances de la agricultura se ha llegado al desarrollo de sistemas verticales, estos permiten tener un aprovechamiento de los espacios, respetando el medio ambiente y promoviendo una cultura del consumo de alimentos orgánicos. Una tendencia en estos sistemas verticales es la inclusión del internet de las cosas (IoT). Gracias a esta tecnología es posible hacer un monitoreo de las condiciones agrícolas en aplicaciones web. Es una forma más conveniente de realizar un seguimiento del rendimiento agrícola vertical y contribuir a la investigación y el desarrollo en el campo agrícola con análisis de datos de casos reales. Con la ayuda del sistema, podría formarse un entorno confiable para las actividades agrícolas. En consecuencia, los cultivos se cultivan en un entorno controlado y eliminan los peligros causados por condiciones climáticas extremas, como sequías e inundaciones [18, 9].

El reto de la agricultura urbana y sus huertas urbanas, es aprovechar simultáneamente materiales de re-uso y reciclaje disponibles en los hogares, ofreciendo soluciones de mínimo costo e integración de subsistemas de producción de abono, recolección de agua lluvias y conservación de semillas que permitan la auto sostenibilidad de la huerta [10]; debido a que el 55 % de las hortalizas que se producen en América Latina se pierden antes de llegar a los mercados o se desperdician en manos de los consumidores, según la Organización de las Naciones Unidas para la Alimentación y la Agricultura (FAO) [10].



5. Procesos de Compost

5.1 Aprovechamiento de los Residuos Orgánicos en Colombia.

El censo DANE del 2018 reveló que un 77.1% de la población colombiana se concentra en cabeceras municipales [22], y entre las cuatro ciudades principales (Bogotá, Cali, Medellín y Barranquilla) se generaba el 37.7% de residuos sólidos a nivel nacional, siendo el relleno sanitario la disposición final mayormente efectuada. Por tal motivo se esperan programas y proyectos de aprovechamiento de los residuos sólidos orgánicos, un claro ejemplo es el de “Huila sin Basuras” donde cabe recalcar que los residuos orgánicos aprovechables ingresan a la planta de compostaje, y una vez se encuentra madurada se procede a moler y generar así fertilizantes, donde un porcentaje se comercializa con los agricultores y ganaderos del sector y otro se complementa con una adición de minerales para su venta [23].

5.2 Conceptos Básicos del Compostaje

Según [24] y [25] se deben identificar los siguientes conceptos como claves a la hora de hablar de compostaje, toda vez que un correcto uso de ellos puede evitar una equívoca interpretación y logra de esta forma profundizar en temas que tal vez investigadores alejados del área de la agricultura no manejen.

ABONADO: Acción o proceso cuya finalidad es hacer que la tierra sea fértil o productiva. Aplicación de fertilizante, ya sea sintético o natural.

AERÓBICO: Proceso que ocurre en presencia de oxígeno. Para que un compost funcione con éxito se debe proporcionar suficiente oxígeno para que mantenga el proceso aeróbico.

APROVECHAMIENTO DE LOS RESIDUOS: Conjunto de acciones cuyo objetivo es recuperar el valor económico de los residuos mediante su reutilización, remanufactura, rediseño, reciclado y recuperación de materiales secundados o de energía.

COMPOSTAJE: Es un proceso biológico de reciclaje completo de la materia orgánica mediante el cual ésta es sometida a fermentación en estado sólido, controlada (aerobia) con el fin de obtener un producto estable para ser utilizado en la agricultura.

COMPOST: También llamado abono orgánico es el producto resultante del proceso de compostaje.

DISPOSICIÓN FINAL: La acción de depositar o confinar permanentemente residuos sólidos en sitios o instalaciones cuyas características prevean afectaciones a la salud de la población y a los ecosistemas y sus elementos.

FERMENTACIÓN: Transformación de compuestos orgánicos en compuestos más simples y/o inorgánicos por la acción de microorganismos.

HUMIFICACIÓN: Es el proceso de formación de ácidos húmicos y fúlvicos, a partir de la materia orgánica mineralizada.

HUMUS: Materia orgánica descompuesta, amorfa y de color marrón oscuro de los suelos, que ha perdido todo indicio de la estructura y la composición de la materia vegetal y animal a partir de la que se originó. Por tanto, el término humus se refiere a cualquier materia orgánica que ha alcanzado la estabilidad y que se utiliza en la agricultura para enmendar el suelo. El producto de la lombriz suele llamarse equivocadamente humus, cuando en realidad debe llamarse vermicompost.

LAVADO O LIXIVIACIÓN DE NITRATOS: Cuando el agua entra en contacto con fertilizantes nitrogenados o con estiércol, puede disolver los nitratos y otros componentes solubles del estiércol y transportarlos disueltos en su seno cuando se infiltra en el suelo y desciende hasta las aguas subterráneas.

MACROORGANISMOS: Organismos vivos que pueden ser observados a simple vista (arañas, lombrices, roedores, hormigas, escarabajos...). También se denomina mesofauna.

MICROORGANISMOS MESÓFILOS: Grupo de bacterias, y hongos (levaduras u hongos filamentosos) que pueden vivir, trabajar y multiplicarse durante el compostaje entre los rangos de temperatura de 30°C a 40°C.

NITRÓGENO: elemento indispensable para las plantas que puede estar en forma orgánica (proteínas y compuestos orgánicos), o inorgánica (nitrato o amonio).



ORGÁNICO: Pertenece o derivado de los organismos vivos. Que pertenece a los compuestos químicos que contienen carbono.

RELACIÓN C:N: cantidad de carbono con respecto a la cantidad de nitrógeno que tiene un material.

RESIDUOS SÓLIDOS: En función de la actividad en que son producidos, se clasifican en agropecuarios (agrícolas y ganaderos), forestales, mineros, industriales y urbanos. A excepción de los mineros, los demás poseen numerosos aspectos comunes desde el punto de vista de la recuperación y reciclaje.

VERMICOMPOST: También conocido como lombricultura o vermicultura, es el cultivo de la lombriz para la descomposición de materia orgánica.

5.3 Beneficios del Compost

Román [25] clasifica los beneficios del compost en tres divisiones, mejora de las propiedades físicas, mejora de las propiedades químicas y mejora de la actividad biológica, las cuales se describen a continuación.

Mejora las propiedades físicas:

- Facilitando el manejo del suelo para las labores de arado o siembra.
- Aumentando la capacidad de retención de la humedad del suelo.
- Reduciendo el riesgo de erosión.
- Ayudando a regular la temperatura del suelo (temperatura edáfica).
- Reduciendo la evaporación del agua y regulando la humedad.

Mejora las propiedades químicas:

- Aportando macronutrientes, como N, P, K y micronutrientes.
- Mejorando la capacidad de intercambio de cationes.

Mejora la actividad biológica:

- Aportando organismos (como bacterias y hongos) capaces de transformar los materiales insolubles del suelo en nutrientes para las plantas y degradar sustancias nocivas.
- Mejorando las condiciones del suelo y aportando carbono para mantener la biodiversidad de la micro y macrofauna (lombrices).

5.4 Proceso de Compostaje

Es un proceso biológico y seguro en el cual actúan agua, aire, microorganismos y en algunas ocasiones macroorganismos, para transformar materia de origen animal y/o vegetal mediante un proceso de descomposición en abono orgánico o compost, generando también calor, vapor de agua, dióxido de carbono y más microorganismos. En el proceso se deben tener en cuenta múltiples factores para generar un compost de alta calidad, a continuación se explicarán las dos formas de compostaje mayormente utilizadas, la común y el vermicompost, con las condiciones necesarias para su producción.

5.4.1 Compost

Este proceso sucede en presencia de microorganismos, los cuales aprovechan el carbono C y el nitrógeno N de la materia orgánica para generar su propia biomasa. Al descomponerlos se generan diferentes niveles de temperatura a lo largo del tiempo por tal motivo se reconocen cuatro fases en el proceso de compostaje de acuerdo a esos niveles [24], [25].

Fase Mesófila: El proceso comienza a temperatura ambiente y en pocos días aumenta la temperatura hasta los 45° debido a los microorganismos quienes han empezado a descomponer la



materia orgánica. En esta fase el pH puede disminuir hasta 4.0 o 4.5 debido a la aparición de ácidos orgánicos. Puede tener una duración de entre 2 y 8 días.

Fase Termófila o de Higienización: La temperatura pasa los 45° hasta los 60°, mueren los microorganismos mesófilos y las malas hierbas. Comienza la descomposición de los termófilos y composiciones más complejas de C, estos microorganismos transforman el N en amoníaco por lo que el pH aumenta. Puede tener una duración de días o meses ya que influye el material orgánico inicial, la condiciones climáticas del lugar y otros factores.

Fase de Enfriamiento o Mesófila II: La temperatura disminuye hasta llegar a la temperatura de 40° o 45°, se consume el material orgánico y algunos compuestos de C restantes. Aparecen algunos hongos indicadores de esta fase, los cuales descomponen la celulosa del suelo y la temperatura disminuye hasta la temperatura ambiente, en el proceso el pH disminuye aunque es un poco alcalino. Puede tener una duración de varias semanas.

Fase de Maduración: Es un periodo de tiempo en el cual se realizan los últimos procesos de degradación y se disminuye la actividad metabólica, toda la fase ocurre a temperatura ambiente y puede durar casi un mes o hasta más.

Finalmente, las condiciones que afectan la supervivencia y reproducción de los microorganismos son: temperatura, humedad, pH, oxigenación y la relación C:N [24], [25].

Temperatura: Este factor tiene mucha variabilidad durante el proceso puesto que es dependiente de la fases del proceso. Sin embargo se sugiere que las temperaturas estén en el rango de 35° a 70°.

Humedad: El rango ideal se encuentra entre el 45% y 60% pues si es demasiado baja la actividad de los microorganismos se ralentiza, y si es muy alta no habrá suficiente aireación.

pH: Al igual que la temperatura, el pH varía en cada fase, también depende de los materiales orgánicos utilizados. El rango ideal es entre 4.5 y 8.5.

Oxígeno: La presencia de oxígeno es totalmente necesaria en el proceso pues este debe ser aerobio, además de que los microorganismos necesitan aireación, el nivel óptimo es del 10%.

Relación C:N: La necesidad de carbono y nitrógeno varía durante las fases, siendo el rango óptimo una relación de 15:1 hasta 35:1, la relación C:N disminuye 2/3 con el tiempo porque parte del carbono se pierde como CO₂ durante las fases.

5.4.2 Vermicompost

En este proceso, además de los microorganismos, también se involucran las lombrices produciendo biomasa y humus (abono orgánico) de alta calidad, estas lombrices se crían de diversas maneras para la producción de compost, como en torres o baldes, se alimentan de material orgánico y biodegradable igual que los microorganismos. Las lombrices por sí mismas al estar en constante movimiento airean el suelo, otorgando niveles óptimos de oxígeno para el proceso de compostaje.

El humus es un producto estable que brinda muchos más nutrientes a las plantas, ya que contienen cinco veces más nitratos que el suelo, 11 veces más potasio y, lo que es más importante, 7 veces más fósforo intercambiable y 3 veces más magnesio intercambiable [24]. El humus líquido es otro producto resultante del vermicompost, se puede definir como el lixiviado proveniente de la disolución de los nutrientes de la composta en agua.

La lombriz roja californiana (*Eisenia foetida*) comercialmente es la que más se emplea pues tiene una gran habilidad en la digestión de material orgánico, de forma que cada 24 horas consume alimento correspondiente a su propio peso al día. Para la cría de lombrices se deben cumplir



fundamentalmente 5 parámetros: Suficiente sustrato, una humedad del 80%, un pH de 6.5 a 7.5, una temperatura de 18°C a 24°C y baja luminosidad [25].

La vermicompostera puede ser realizada con estibas de madera, contenedores o canastas, existen diferentes opciones, en cuanto a tamaño y calidad, se requiere ser abierto para obtener mejores resultados. Mayormente se utilizan pisos de canastas o baldes apilados uno sobre el otro con perforaciones para la circulación del aire, separando los cultivos de lombrices, obteniendo un humus estable pues se descomponen los residuos de manera uniforme [26].

Los sustratos más recomendables para la alimentación de la lombriz son los residuos orgánicos crudos, es decir, todos los residuos de frutas y verduras, si se quiere estos residuos se pueden someter a un precompostaje, facilitando el proceso de absorción de nutrientes por parte de las lombrices. El proceso puede durar de 3 a 4 meses, posteriormente el humus estará listo para su recolección y secado, mientras que las lombrices entrarán a un nuevo proceso de vermicompostaje [26].

De acuerdo con el SIPAF [27], algunas de las ventajas del lombricompost frente al compost vegetal son: Demanda de menos mano de obra que el compost vegetal. Reducción de costos del proceso en el lombricompost debido a que no se requiere de maquinaria y mano de obra para realizar los volteos al material, ya que las lombrices se encargan de mover los residuos y airear la mezcla. En el lombricompostaje no se pierden nutrientes durante un proceso prolongado, como sí puede suceder con los procesos de compostaje. El abono resultante tiene mejor calidad que el compost, debido a la gran cantidad de microorganismos benéficos que aporta la lombriz. De la lombricultura se obtienen abono sólido, líquido y pie de cría, que son tres productos para uso doméstico o comercialización.

6. Generación de Conceptos

A continuación, se mostrará la caracterización de las especificaciones preliminares para el diseño y puesta en marcha del proyecto Verticalium. Aspectos como optimización de terrenos, sensórica, autonomía y monitoreo remoto fueron considerados relevantes en la generación de conceptos.

6.1 Requerimientos de Diseño

Para el desarrollo de este proyecto se han levantado unos requerimientos y necesidades iniciales por parte del cliente los cuales se han listado en la tabla 1.

Tabla 1. Especificaciones Preliminares

#	Especificación	Importancia	UND	Valor Marginal	Valor Ideal
1	Macronutrientes obtenidos de manera natural	5			
2	Sistema de control y monitoreo remoto	4			
3	Estructura que permita el cultivo de 24 plántulas	4	und	>20	24



Tabla 1 (cont.). Especificaciones Preliminares

#	Especificación	Importancia	UND	Valor Marginal	Valor Ideal
4	Exposición a luz artificial controlada	5	lm	>200	300
5	Autosostenible-autónomo	5	W	>10	15
6	Trabajo Continuo	5	hr	24	>12
7	Bajo Costo	5	\$USD	<1000	<600

6.2 Descomposición Funcional

En el desarrollo de esta actividad se puede descomponer el problema partiendo de las necesidades y especificaciones preliminares. Este proceso de descomposición tiene el fin de hacer una abstracción de la idea de la identidad de los elementos constitutivos del producto y así tener una representación macroscópica del proyecto y de su función global. Para esto se ha elaborado un esquema jerárquico modular basados en la metodología de desarrollo de sistemas *Top-Down*.

Esta metodología permite partir de lo más general a lo más detallado, es un proceso análogo que permite ver el concepto general y a partir de allí comenzar a detallar los subsistemas involucrados. En la Figura 1 se observa que la prioridad es tener como salida del sistema una cosecha óptima y que sea posible hacer el monitoreo de las variables que intervienen en el proceso.

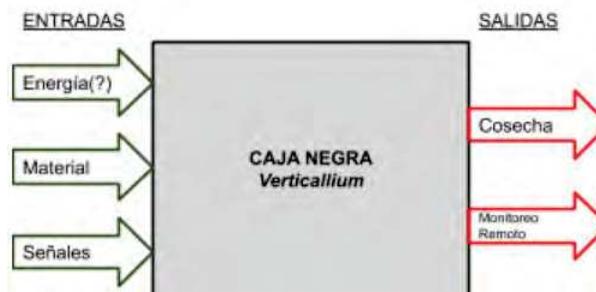


Figura 1 Caja Negra-Descripción de entradas y salidas

La metodología implementada para el desarrollo de este proyecto se describe en la Figura 1 a partir de la descomposición funcional a través de bloques. Aquí se observa un proceso modular para la construcción e integración del sistema.

Las diferentes etapas se conforman por un sistema de adquisición de datos que tiene el propósito de medir las variables físicas del cultivo como son Temperatura, Humedad, pH y Luminosidad. Una vez centralizada esta información se procede con el envío de la misma a un servidor en la nube.

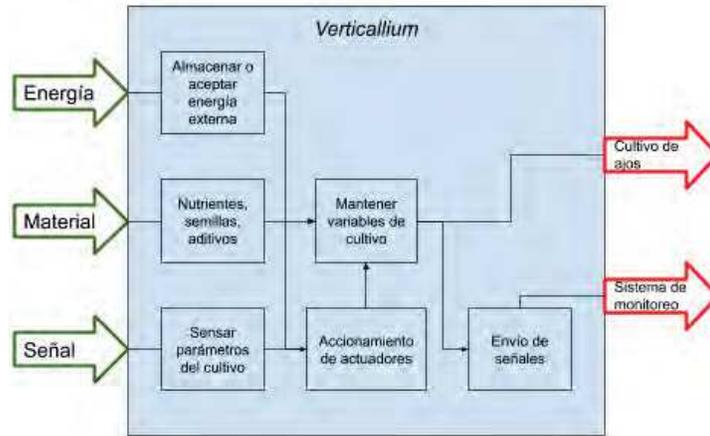


Figura 2. Caja Transparente-Descomposición de bloques funcionales

6.3 Generación de Concepto

Cada bloque del diagrama de bloques funcionales puede ser resuelto, solucionado o llevado a cabo por diferentes dispositivos, por eso surgen los conceptos para cada bloque, y se crea una lista que está compuesta por todos los posibles dispositivos que dan solución al requerimiento. Inicialmente en esta “lluvia de conceptos” se toman todos los existentes sin discriminación alguna, lo que puede llevar a soluciones altamente ingeniosas, exitosas, tecnológicas e innovadoras. El mismo proceso de selección de conceptos se encarga de eliminar soluciones no viables y encontrar la más óptima para el diseño. La principal ventaja de tener esta generación de conceptos en una metodología de diseño concurrente se encuentra en que es posible retomar conceptos en el momento que se considere pertinente sin una gran alteración en el proceso de desarrollo del producto. Sin embargo, a continuación la figura 3 ilustra la arquitectura preliminar que cumple con todos los requerimientos de diseño.

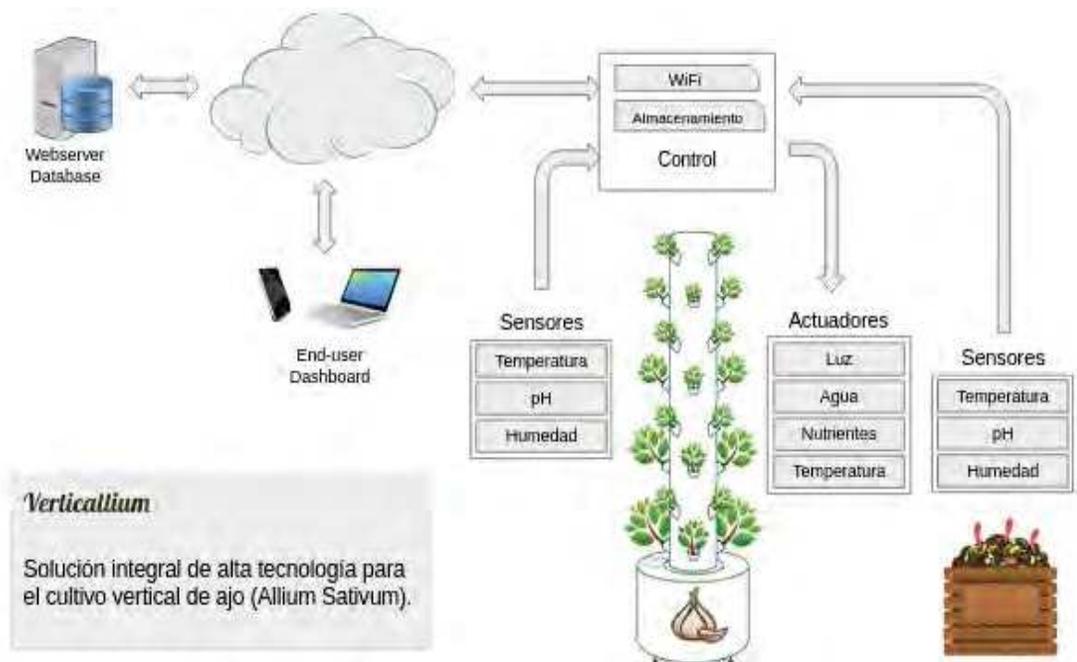


Figura 3. Arquitectura preliminar



6.4 Diseño de Detalle

El sistema de control embebido del proyecto usa una tarjeta de desarrollo basada en el microcontrolador ESP32. El proceso de adquisición de datos se ha planteado según los requerimientos anteriormente listados, para ello se han conectado cuatro tipos de sensores que permiten medir las variables de luminosidad, temperatura, humedad y el pH en la tierra de cultivo. Estas variables son medidas con los sensores BH1750, DS18B20, YL100 y el sensor GravityAnalog pHmetro respectivamente. Cada uno de estos valores obtenidos son llevados a un dashboard en tiempo real que permite monitorear las variables y el comportamiento de los actuadores, para así mantener las condiciones óptimas para el cultivo.

6.4.1 Adquisición de datos

El sistema de adquisición de datos de la huerta vertical fue compuesto con la implementación de diferentes sensores que permiten la medición en tiempo real de las variables de temperatura, humedad y luminosidad en la tierra de cultivo y sus señales fueron centralizadas en un microcontrolador ESP32 tal y como se ilustra en la Figura 4.

1. *Sensor BH1750*: Este sensor permite hacer mediciones de flujo lumínico de forma digital, posee un conversor ADC de 16 bits con una salida digital I2C. A través de esta se entrega la intensidad lumínica directa en unidades de Lux (lumen/m²). Se decide implementar este tipo de dispositivo con el objetivo de medir la incidencia de las luces LED que posee el sistema y así poder variarlas según la etapa del proceso que se esté realizando.

2. *Sensor DS18B20*: Este sensor es un dispositivo versátil que permite medir la temperatura en ambientes de alta humedad, incluso dentro del agua. Su rango de temperaturas está entre -55°C y 125°C y tiene un margen de error en medición desde $\pm 0,5^\circ\text{C}$ hasta $\pm 2^\circ\text{C}$. Este será el encargado de dar a conocer la temperatura de la tierra de cultivo con el fin de determinar la necesidad de agua o mermar la intensidad de luz.

3. *Sensor YL100*: Este sensor permite detectar la humedad del suelo, cuando el módulo demuestra déficit de humedad de la tierra de cultivo genera un nivel alto y viceversa. Gracias a este dispositivo es posible automatizar el riego del cultivo siguiendo los parámetros del proceso.

4. *Sensor GravityAnalog pHmetro*: Este medidor de pH analógico es el encargado de reflejar la acidez o alcalinidad en la solución del agua y el humus líquido en el reservorio. Sin embargo al querer medir una variable química se debe tener en cuenta que es un proceso alterno que requiere una supervisión y esta medición no puede ser continua con los pHmetros convencionales, pero el dispositivo seleccionado tiene la ventaja de que si se puede dejar sumergido en la solución a medir. Al estar en contacto con la solución y al no presentar una variación tan drástica la calibración se hará una vez al año durante los dos primeros años de funcionamiento y después de esto será cada 6 meses [28].

6.4.2 Sistema de energía solar

Debido al interés por disminuir los consumos energéticos del proyecto potencializando su sostenibilidad se ha implementado un sistema de energía solar. Este sistema alimenta todos los elementos eléctricos y electrónicos. El sistema de energía solar se ha basado en la integración de un panel fotovoltaico, un regulador y un rack de baterías (Figura 5). Dado que el sistema es totalmente electrónico y los elementos como actuadores no son de gran potencia no será necesario un inversor logrando que los costos sean menores.

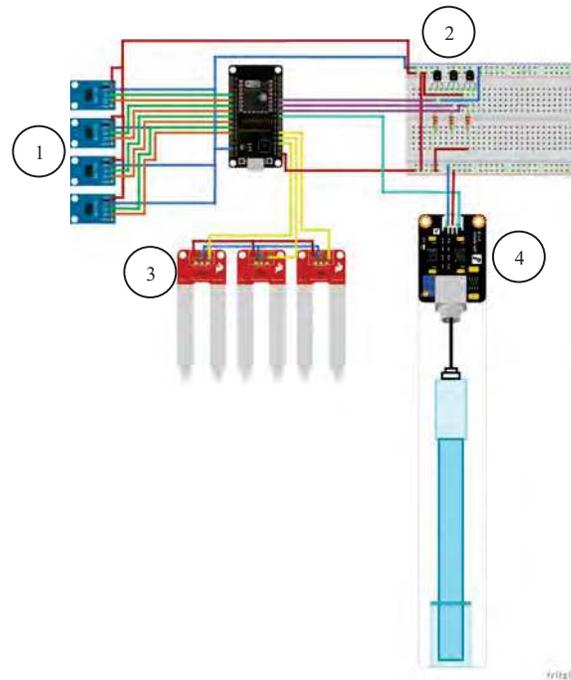


Figura 4. Diseño esquemático del sistema electrónico

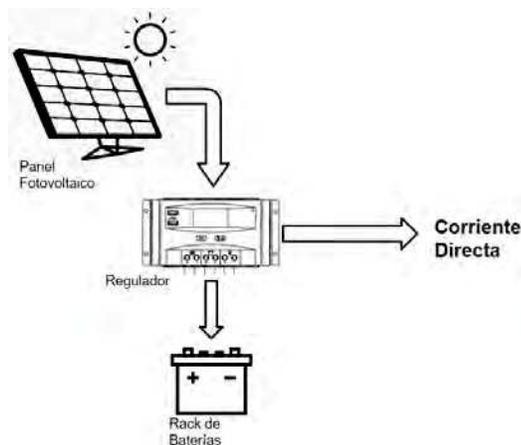


Figura 5. Modelo esquemático del sistema de energía solar

6.4.3 Seguidor solar de dos ejes

Además con el fin de tener un mayor aprovechamiento de la incidencia de la luz del sol se agregó un seguidor solar de dos ejes. De esta manera, se ha creado un sistema independiente que no tenga interferencia directa en el proceso de cultivo, y para ello se seleccionó un Arduino Uno, al cual se conectarán cuatro sensores tipo LDR y dos servomotores encargados del posicionamiento activo en tiempo real del panel solar como se observa en la Figura 6. El mayor beneficio de estos sistemas de seguimiento solar es que ofrece un incremento en la producción de la electricidad, dado que puede presentar un aumento en el rendimiento del sistema. Si la instalación se realiza en zonas donde la posición del sol varía de manera arbitraria en meses de invierno y verano un sistema de seguimiento solar de dos ejes maximizaría la producción solar y así generar suficiente potencia para alimentar la huerta.

Este sistema se mueve en dos ejes, entre los cuatro ejes cardinales alineados norte-sur y este-oeste, de esta manera se logra maximizar la producción de energía durante todo el año, variando la orientación según la estación y siguiendo la posición solar de forma más precisa.

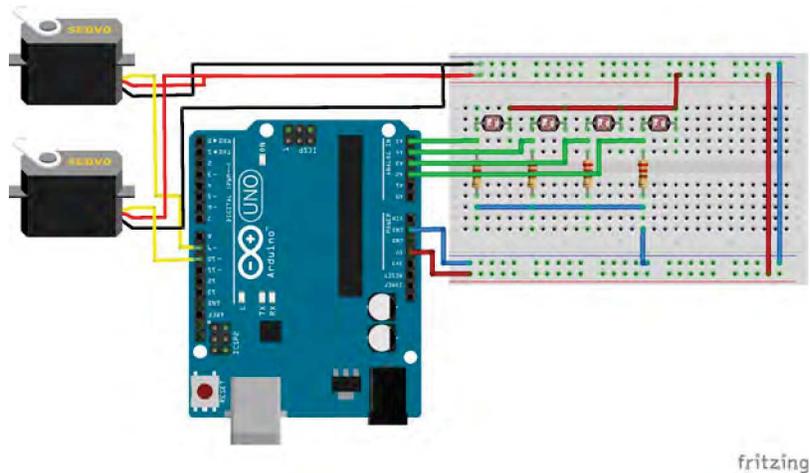


Figura 6. Diseño esquemático del seguidor solar

6.4.4 Estructura vertical

La estructura de la huerta vertical (Figura 7) se escogió del tipo jaula por su facilidad de armado y por su capacidad de apertura de la luz, con este tipo de estructura es posible lograr que la iluminación tenga mayor cobertura a través de las macetas. El material de la jaula es en tubería de PVC con una altura de 180 cm y una base de 70X70 cm. En las caras internas se han sujetado 4 lámparas LED encargadas de simular las condiciones de iluminación, dando cumplimiento al requerimiento de exposición a luz artificial controlada. Aunado a esto se han apilado seis macetas tipo trébol para cultivar un total de 24 plantas.



Figura 7. Modelo 3D de huerta vertical

En el área central de la huerta vertical se ha incluido un canal con múltiples orificios que es el encargado de suplir las necesidades nutritivas del cultivo, a través de este se inyecta el vermicompost maduro el cual libera la concentración de nutrientes en cada uno de los niveles de la huerta.

A un lado de este canal central se ha ubicado una manguera de riego, la cual tiene como función principal mantener la humedad ideal de la tierra de cultivo haciendo recircular una solución de



agua reposada en el recipiente base con aditivo de humus líquido, generado por la filtración del agua de riego en el vermicompost y de las macetas. El sistema de riego también es controlado y automatizado y cumple su función dependiendo de los valores sensados en función de la humedad.

7. Implementación

7.1 Sistema de control

El sistema al ser un proceso bidireccional permitirá la toma de acciones de forma embebida y de manera remota. Para esto se centraliza la información captada por los sensores del sistema y se agrupa en una trama de datos que es enviada a la nube (Figura 8) haciendo uso del Internet de las cosas (IoT). Esta información es procesada localmente en caso de tener que tomar acciones correctivas en la posible desviación de las condiciones ideales del proceso (Figura 9). Además, esta información es proyectada en un dashboard que permite tener una interacción para cargar unos nuevos parámetros en el sistema. Esta parametrización dependerá de la etapa de cultivo y de esta forma tener condiciones ideales que se ajusten a estas.

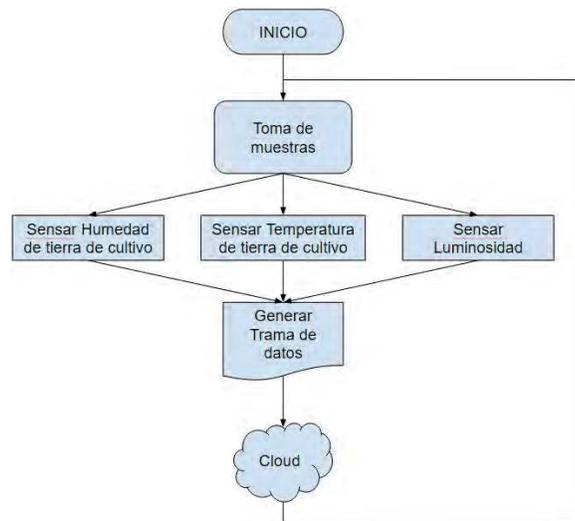


Figura 8. Diagrama de bloques del microcontrolador para envío de datos

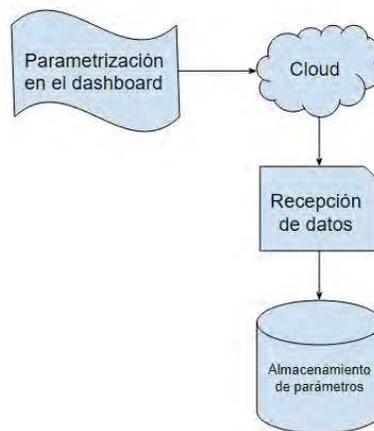


Figura 9. Diagrama de bloques del microcontrolador para recepción de datos

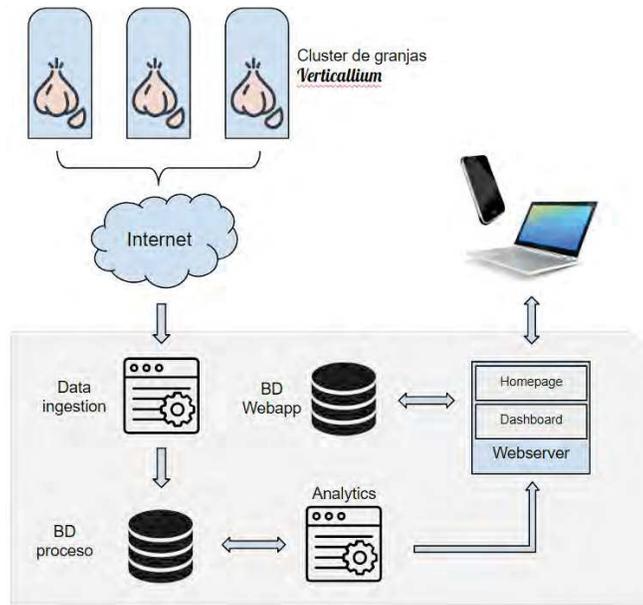


Figura 10. Arquitectura web del sistema

El sitio web ha sido desarrollado en un template de wordpress y se le han añadido una serie de plugins que permiten captar e interpretar la información obtenida de las variables físicas del proyecto. Además es posible tener una interacción desde este sitio para cargar nuevos parámetros de control y es posible acceder a él por medio de la url (<http://verticallium.tech/>)

7.2 Vermicompostadora

Se implementó el método de vermicompostaje para obtener un abono orgánico de alta calidad. La construcción de la vermicompostadora se realizó en tres niveles como se observa en la Figura 11, el primer nivel y el más bajo es donde se recolecta el humus líquido, el segundo nivel es para obtener la primera producción de compost, y por último, el tercer nivel es para una futura producción.

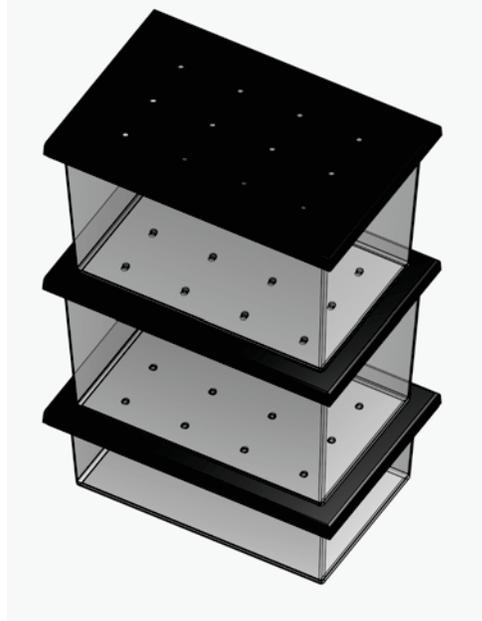


Figura 11. Modelo 3D de Vermicompostadora



El contenido máximo de el segundo y tercer nivel es de 11L, además de esto ambos tienen perforaciones en sus bases, como también las tapas, para drenar el humus líquido hasta el primer nivel, que tiene una capacidad de 9L.

El vermicompostaje se realizó bajo cubierta, pues la lombriz roja californiana requiere un ambiente fresco y parcialmente oscuro. Para la primera producción de compost el sustrato o alimento para las lombrices fueron utilizados los residuos vegetales crudos de la cocina, la frecuencia de alimentación fue de tres días y la cantidad fue de aproximadamente 1lb cada vez, en ese periodo también se regó con agua el suelo para mantener una humedad relativamente cercana a la ideal.

El humus líquido obtenido luego del proceso de riego no es abundante, pues el suelo absorbe gran cantidad del agua y la retiene por un largo periodo de tiempo, debido a esto no es necesario regar constantemente, el humus líquido se adiciona a este riesgo y estimula el proceso de abonado, de igual manera se utilizó en pequeñas cantidades para el brote de los ajos, adecuándolos a este abono desde sus primeras fases de crecimiento.

8. Conclusiones

Una vez realizado el proceso de desarrollo de *Verticallium* se puede concluir que los sistemas de monitoreo de huertas verticales urbanas basados en *IoT* pueden llegar a ser una estrategia para reducir la intervención del hombre, proporcionando sistemas autónomos de procesos productivos inteligentes.

Con la implementación de estas huertas verticales urbanas se puede evidenciar la optimización de espacios, toda vez que no se necesita de grandes terrenos para hacer una producción controlada de diferentes vegetales. Además, se puede hacer un aprovechamiento de los residuos sólidos orgánicos en los hogares evitando el aumento de las basuras y a su vez reduciendo el uso de aguas. Esto hace que sean respetuosos con el medioambiente y se asegura un cultivo de calidad rico en nutrientes, produciendo así sistemas autosostenibles para producción de alimento de consumo humano.

Como trabajos futuros, se propone incluir a la huerta vertical un sistema de monitoreo con cámaras y el uso de *Machine Learning* para hacer un control de calidad de los cultivos supervisando de manera remota la aparición de plagas y agentes patógenos.

Referencias

- [1] Economist Intelligence Unit. “*Global food security Index - Colombia food security*”. The Economist Group: 2019.
- [2] Prager, S. Béné C. “*Colombia y la COVID-19: ¿Crisis u oportunidad?*”, CIAT Comunicaciones. 2020
- [3] Escuelas Rurales Alternativas (ERA), Colombia. Proyecto de PASO COLOMBIA.
- [4] “*Alimentos orgánicos, un negocio para sacarle provecho en Colombia*”, Portafolio: 2018.
- [5] “*¿Por qué no despega la producción de alimentos orgánicos en Colombia?*”, Revista Dinero: 2018.
- [6] Vega, J. “*Los productores de ajo locales buscan frenar las importaciones de este producto*”, Fedeorganicos: 2018.
- [7] Tisnés, F; De la Rotta, C; “*El cultivo del Ajo*”, Federación Nacional de Cafeteros de Colombia, Cuarta Edición.
- [8] “*No llegan barcos a Buenaventura desde China, hay escasez hasta de ajos*”, Caracol Radio: 2020.
- [9] Shien, Y. Audah, L. “*Vertical Farming Monitoring System Using the Internet of Things (IoT)*”, AIP Conference Proceedings, 2017.
- [10] Martínez, C. “*Maravillas del Mundo III: Los Jardines Colgantes*”, El blog de César: 201
- [11] Giobellina, B. “*Infraestructuras verdes : desde el territorio a la cubierta habitable : Serie : innovaciones para la sostenibilidad en vivienda, ciudad y territorio : para el caso de Córdoba*”.



- Editorial de la Facultad de Arquitectura, Urbanismo y Diseño de la Universidad Nacional de Córdoba: 2020. ISBN 978-987-4415-79-0
- [12] Morales, D. “*La historia secreta y completa de los Jardines Colgantes de Babilonia, la maravilla del mundo antiguo*”, Bioguía: 2019
- [13] Needham, J. “*Science and Civilization in China: Volume 4, Physics and Physical Technology, Part 2, Mechanical Engineering*”. Caves Books 1986.
- [14] Ruiz, S. “*De la agricultura arcaica al agronegocio y los modelos asociativos. Su impacto social*” Journal of Agriculture and Environmental Sciences: 2015. ISSN 2334-2404
- [15] Newman P. “*Daily life in the middle ages*” McFarland: 2001. ISBN 0786450525
- [16] Moran, N. Aja, A. “*Historia de los huertos urbanos. De los huertos para pobres a los programas de agricultura urbana ecológica*” Universidad Politécnica de Madrid.
- [17] Kamata, F. “*Cómo Japón está revolucionando la agricultura sin tierra ni trabajadores*”. BBC News. 2019
- [18] Sarkar, A. Majumder, M. “*Opportunities and Challenges in Sustainability of Vertical Eco-Farming: A Review*”. Journal of Advanced Agricultural Technologies, 2015
- [19] Escobar, H., Pinzón, H., & Parra, M. “*Producción de semilla garantizada de Ajo (Primera edición)*” Fundación Universidad de Bogotá Jorge Tadeo Lozano, 2012
- [20] Burba L. “*Recomendaciones prácticas para el cultivo de ajos*”. INTAEEA Santa Cruz, 2011.
- [21] Quintero, J. “*El cultivo del ajo*”. Ministerio de Agricultura, Alimentación y Medio Ambiente. España ISBN 84-341-0354-0
- [22] DANE, “*Censo Nacional de Población y Vivienda*”. Gobierno de Colombia, 2018.
- [23] Superintendencia de Servicios Públicos Domiciliarios, “*Informe de Disposición Final de Residuos Sólidos*”. República de Colombia, 2018.
- [24] G. J. Henao y L. M. Z. Márquez, “*Aprovechamiento de los residuos sólidos orgánicos en Colombia*”, Posgrado, Universidad de Antioquia, Colombia, 2008.
- [25] P. Román, M. M. Martínez, y Alberto Pantoja, “*Manual de compostaje del agricultor. Experiencias en América Latina*”. Organización de las Naciones Unidas para la Alimentación y la Agricultura, 2013.
- [26] L. A. S. Villada y J. A. A. Torres, “*Manual de Aprovechamiento de Residuos orgánicos a través de Sistemas de Compostaje y Lombricultura en el Valle de Aburrá*”, Primera edición. Medellín: Área Metropolitana del Valle de Aburrá, 2013.
- [27] Grupo de Investigación Sistemas Integrados de Producción Agrícola y Forestal (SIPAF), “*Guía técnica para el aprovechamiento de residuos orgánicos a través de metodologías de compostaje y lombricultura*”. Alcaldía Mayor de Bogotá Secretaría Distrital de Hábitat, 2016.
- [28] Consumer Grade pH Probe. Atlas Scientific Environmental Robotics (<https://atlas-scientific.com/files/consumer-grade-pH-probe.pdf>)



Diseño e implementación de reguladores saturados con ganancias variables para robots manipuladores

Sánchez García Brandon¹, Rodríguez Hernández Ernesto¹, Reyes Cortés Fernando¹, Al-Hadithi B. M.^{2,3} y Félix Beltrán Olga¹

Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla (¹); Centre for Automation and Robotics UPM-CSIC, Intelligent Control Group, Universidad Politécnica de Madrid (²); Department of Electrical, Electronics Control Engineering and Applied Physics, Higher Technical School of Industrial Design and Engineering, Universidad Politécnica de Madrid (³)
Email: brandon.sanchez@alumno.buap.mx

Resumen

En este trabajo se presenta una familia extensa de reguladores de tipo saturado con una gama grande de ganancias variables para robots manipuladores. La propuesta considera un conjunto amplio de esquemas de control formados por funciones hiperbólicas y trigonométricas inversas; a su vez, se incluye a una familia grande de ganancias proporcional y derivativa variables con sintonía automática, compuestas por funciones continuas dependientes del error de posición y la velocidad de movimiento del robot manipulador. Presentamos la propuesta de una función estricta de Lyapunov para realizar la demostración de la estabilidad asintótica y global del punto de equilibrio de la ecuación en lazo cerrado. Finalmente, para evaluar el desempeño y funcionalidad de los reguladores saturados con ganancias variables, exponemos algunos resultados vía simulación de algunos casos particulares de la familia de esquemas de control y ganancias variables para un robot de dos grados de libertad. También, se añade un análisis comparativo con dos reguladores bien conocidos en la literatura como son el proporcional derivativo (PD) y el tangente hiperbólico.

Palabras clave: Regulador saturado, Ganancias Variables, Sintonía automática, Estabilidad de Lyapunov, Robot manipulador.

1. Introducción

Los robots manipuladores son sistemas mecatrónicos que han tomado un papel importante en el desarrollo de nuestra sociedad actual. Son parte indispensable de nuestra producción en masa de la mayoría de los artículos que consumimos, construcción y ensamblaje de automóviles, aplicaciones de pintura, troquelado y corte láser, tratamientos de forja en acero y manufactura de hierro colado, asistencia especializada y rehabilitación fisioterapéutica, entre otros. Sin embargo, para que dicho robot realice fielmente algunas de las aplicaciones antes nombradas, es clave contar con algoritmos de control de alto desempeño. Por esta razón, el desarrollo de nuevos esquemas de control se ha convertido en una actividad científica importante y así aumentar las posibilidades de resolver problemas con robots donde la mano de obra del ser humano ya es limitada [1] [2].

Para indicarle al manipulador ciertas actividades, es indispensable abordar el problema de control de trayectoria. Este consiste en encontrar los pares aplicados por los servomotores (actuadores) para cada articulación del robot y este sea capaz de seguir fielmente una referencia deseada variante en el tiempo. El problema de regulación (control de posición) es un caso particular del control de trayectoria, donde se requiere que el extremo final del robot manipulador llegue a una posición articular deseada constante en el tiempo, desde cualquier condición inicial. El objetivo se centra en diseñar un regulador tal que, el error de posición y la velocidad de movimiento del robot manipulador, tiendan asintóticamente



al punto de equilibrio de la ecuación en lazo cerrado. Cuando se es posible realizar el control de posición de robots manipuladores, se pueden desarrollar aplicaciones de movimiento. Una de las más conocidas en la industria es el control punto a punto, donde la idea interpolar puntos de una trayectoria deseada por medio de varios ciclos de regulación donde, la posición deseada toma el papel de condición inicial y se genera una nueva posición deseada en estado transitorio para repetir el control de posición durante toda la trayectoria deseada [3].

El regulador proporcional-derivativo (PD) y el proporcional-integral-derivativo (PID) son los esquemas de control más simples para resolver el problema de control de posición. El control PD nos garantiza la estabilidad asintótica y global; mientras que el PID sólo puede garantizar dicha estabilidad asintótica de forma local [4]. Por otra parte, debido a ser reguladores no acotados, son capaces de proporcionar cualquier tipo de torque solicitado; por lo tanto, la sintonía de las ganancias es clave para no saturar los límites físicos inherentes del robot manipulador [5]. Entonces, una de las restricciones que se deben tomar en cuenta, es que los robots manipuladores contienen una fuente de par limitada por medio de los servomotores del mismo para que trabajen en su región lineal. A nivel práctico, esto representa una dificultad importante para el diseño de nuevos de controladores. Para dar solución a esto, se han presentado trabajos como en [6]; un esquema de control difuso con torques acotados. Otro enfoque se presenta en [7], con dos esquemas de control con estabilización acotada y global con entradas saturadas.

La técnica de moldeo de energía presentada por Takegaki y Arimoto en 1981, han abierto una puerta para que diversos investigadores propongan nuevos algoritmos de control que resuelven el problema de regulación y los problemas prácticos de los actuadores. Esto es gracias a que su naturaleza matemática es conveniente. Entre los que resaltan está [8] donde se presenta un regulador para robots con articulaciones elásticas; posteriormente se presenta un controlador con torques acotados para robots manipuladores en el trabajo de [9]. Años después, destaca la propuesta de [10], con un controlador del tipo raíz cuadrada para robots manipuladores. Finalmente, en trabajos más recientes, se presenta en [1] una familia extensa de esquemas de control del tipo hiperbólicos para robots manipuladores, cuyo resultado producen reguladores del tipo acotados; se demuestra la estabilidad asintótica y global del punto de equilibrio mediante una función estricta de Lyapunov.

Sin embargo, los esquemas que se diseñan a partir de la técnica de moldeo de energía, tienen sus ganancias constantes; sintonizar estas ganancias es complicado, no hay muchos métodos con rigor científico y por lo tanto el proceso de sintonización se realiza de manera empírica, perjudicando al algoritmo de control con una mala sintonía. Los controladores auto sintonizables existen de hace años, con diferentes concepciones. En el año 2000, se presentó en [11] la estimación de los parámetros de las ganancias variables por medio de algoritmos de seguimiento rápido y suave estado estable. Más tarde, en [12] se quiere obtener la sintonía automática de las ganancias del PD usando el auto-aprendizaje desde el contexto del Growing Multi-Expert Networks (GMN). Otros enfoques de sintonía de las ganancias del regulador PD, es por medio de esquemas lógica difusa [13], sintonía usando un algoritmo de control híbrido inteligente [14], en [15] se implementan dos controladores inteligentes para la sintonía automática y proponer un PD del tipo no lineal para robots manipuladores, como el trabajo de [16]. También, resaltamos el trabajo mostrado en [17] se diseña un PD con ganancias variables dependientes del error de posición articular del robot manipulador, con sintonía automática.

De igual manera, se ha abordado la problemática de la sintonía automática del controlador PID. El trabajo presentado en [18] se muestra una estructura PID con modos deslizantes para el control de trayectoria de robots manipuladores, donde se desconoce la dinámica de manera total o parcial del robot manipulador. En [19] también se presenta un PID basado en lógica difusa auto sintonizable con ganancias no lineales. Más tarde, otro PID con lógica difusa auto sintonizable para un robot manipulador con eslabón flexible es mostrado en [20]. Siguiendo esta temática se presentó en [21] un control PID demostrando estabilidad con sintonía automática por medio de lógica difusa. En [22] el regulador PID presentado, se demuestra estabilidad asintótica local sintonizando las ganancias, también por lógica difusa.



Otra perspectiva, es la que se aborda en [23], presentando dos nuevos controladores de ganancia variable; el primer controlador es el modelo difuso Takagai-Sugeno como una planta lineal, en el segundo esquema de control se proponen varios subsistemas lineales que son localmente estabilizados. El método de control por redes neuronales basado en ganancias variables fue propuesto en [24], donde las ganancias variables se adaptan para un sistema tele-operado en aplicaciones de rehabilitación, asegurando la estabilidad y suavidad en la respuesta del robot esclavo y eliminando las perturbaciones que puede generar el paciente. En [25] se propone un nuevo controlador de entrada estrictamente pasivo programado por ganancia, considerando el control de manipuladores robóticos de enlace flexible.

Desde el contexto de la técnica de moldeo de energía, en [26] se propone un nuevo regulador para robots manipuladores, basado en la función seno hiperbólico con reglas de sintonía para las ganancias variables; se propone una función estricta de Lyapunov para la demostración de la estabilidad asintótica y global del punto de equilibrio. En [2] se presenta una familia grande de reguladores saturados tipo hiperbólicos para robots manipuladores; la propuesta considera a la ganancia proporcional constante y a la ganancia derivativa variable con sintonía automática definida en función del error de posición, velocidad de movimiento y una inyección de amortiguamiento. Finalmente, en [27] se presenta un regulador del tipo seno hiperbólico para robots manipuladores, demostrando la estabilidad asintótica del punto de equilibrio por medio de una función estricta de Lyapunov; la sintonía de las ganancias es obtenido mediante Particle Swarm Optimisation, este esquema está inspirado en el comportamiento social de animales tales como los pájaros o las hormigas.

Sin embargo, los trabajos reportados hasta ahora no proponen a un conjunto amplio de esquemas de control saturados con una gama extensa de ganancias variables con sintonía automática; las cuales están compuestas por funciones continuas dependientes del error de posición y la velocidad de movimiento del robot. La motivación de este trabajo es resolver la problemática de regulación tomando en cuenta las limitantes prácticas de los robots manipuladores por medio de regulación saturada a partir de la técnica de moldeo de energía y la sintonía automática de las ganancias variables. Siendo un tema abierto para la comunidad científica en robótica. Al mismo tiempo, mediante la propuesta de una función estricta de Lyapunov, es posible llevar a cabo la demostración de estabilidad asintótica y global del punto de equilibrio de la ecuación en lazo cerrado; formada por el esquema de control propuesto y el modelo dinámico de un robot manipulador con articulaciones rígidas de n grados de libertad. Presentamos el procedimiento para el diseño de las matrices de ganancia variables; es decir, definir la estructura matemática de los casos particulares de la familia extensa de ganancias variables. Continuamos evaluando el funcionamiento y desempeño de nuestra propuesta por medio del ambiente de simulación que nos ofrece **MATLAB** para un robot de dos grados de libertad; realizamos un análisis comparativo con tres reguladores conocidos como son el PD y el tangente hiperbólico. Posteriormente, usando la norma \mathcal{L}_2 medimos el desempeño de cada uno de los esquemas anteriormente analizados.

Este artículo está organizado de la siguiente manera. En la Sección 2 presentamos el modelo dinámico de un robot de n grados de libertad con articulaciones rígidas y algunas propiedades matemáticas que este cumple. La sección 3 es usada para presentar a los reguladores saturados con ganancias variables más una compensación de gravedad, mostramos a la familia extensa de ganancias variables y realizamos la demostración de estabilidad asintótica y global del punto de equilibrio de la ecuación en lazo cerrado. En la Sección 4, presentamos los casos particulares de la familia de ganancias variables y describimos el procedimiento para su diseño. La Sección 5 tiene el objetivo de presentar los resultados vía simulación del desempeño de nuestro regulador con ayuda de un análisis comparativo con otros reguladores conocidos en la literatura. Finalizamos con la Sección 6, dedicada a exponer las conclusiones obtenidas en este trabajo.

2. Dinámica del robot manipulador

Se ha considerado el modelo matemático (1) de un robot rígido de n grados de libertad, en cadena abierta y con todas sus articulaciones rotacionales:



$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\dot{\mathbf{q}}, \mathbf{f}_e) = \boldsymbol{\tau} \quad (1)$$

siendo: $\mathbf{q} \in \mathbb{R}^n$ es el vector de coordenadas generalizadas o posiciones articulares, $\dot{\mathbf{q}} \in \mathbb{R}^n$ es el vector de velocidades articulares, $\ddot{\mathbf{q}} \in \mathbb{R}^n$ es el vector de aceleraciones articulares, $M(\mathbf{q}) \in \mathbb{R}^{n \times n}$ es la matriz de inercia, simétrica y definida positiva, $C(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ es la matriz de fuerzas centrípetas y de Coriolis, $\mathbf{f}(\dot{\mathbf{q}}, \mathbf{f}_e) \in \mathbb{R}^n$ es el vector de fuerzas o pares gravitacionales y $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ es el vector de pares de fricción que incluye la fricción viscosa, de Coulomb y estática f_{e_i} de la i -ésima articulación del robot manipulador, por último $\boldsymbol{\tau} \in \mathbb{R}^n$ es el vector de pares o torques aplicados. El modelo dinámico del robot manipulador cumple con las siguientes propiedades:

Propiedad 1.

La matriz de fuerzas centrípetas y de Coriolis $C(\mathbf{q}, \dot{\mathbf{q}})$ y la derivada temporal de la matriz de inercia $\dot{M}(\mathbf{q})$ satisfacen lo siguiente (2) y (3):

$$\dot{\mathbf{q}}^T \left[\frac{1}{2} \dot{M}(\mathbf{q}) - C(\mathbf{q}, \dot{\mathbf{q}}) \right] \dot{\mathbf{q}}, \quad \forall \mathbf{q}, \dot{\mathbf{q}} \in \mathbb{R}^n \quad (2)$$

$$\dot{M}(\mathbf{q}) = C(\mathbf{q}, \dot{\mathbf{q}}) + C(\mathbf{q}, \dot{\mathbf{q}})^T \quad (3)$$

Propiedad 2.

Para todos los robots que sólo contienen articulaciones rotacionales, entonces se cumple:

$$\|C(\mathbf{q}, \mathbf{y})\mathbf{z}\| \leq k_{ro} \|\mathbf{y}\| \|\mathbf{z}\| \quad (4)$$

Donde $k_{ro} > 0 \forall \mathbf{y}, \mathbf{z} \in \mathbb{R}^n$.

Propiedad 3.

Sean las constantes positivas k_{pl} y k_{pu} las cotas inferiores y superiores de la integral (5), respectivamente; en el intervalo cerrado: $k_{pu} \geq k_{pi}(\tilde{q}_i) \geq k_{pl} \quad \forall \tilde{q}_i \in \mathbb{R}$ para $i = 1, 2, 3, \dots, n$.

$$\frac{1}{2} k_{pl} \|\tilde{\mathbf{q}}\|^2 \leq \int_0^{\tilde{\mathbf{q}}} \boldsymbol{\sigma}^T K_p(\boldsymbol{\sigma}) d\boldsymbol{\sigma} \leq \frac{1}{2} k_{pu} \|\tilde{\mathbf{q}}\|^2 \quad (5)$$

3. Reguladores saturados con ganancias variables

Se presentan los siguientes reguladores saturados con una familia extensa de ganancias variables dependientes del error de posición y la velocidad de movimiento del robot manipulador más una compensación de gravedad:

$$\boldsymbol{\tau} = K_p(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) \begin{bmatrix} \frac{\tanh(\varphi \tilde{q}_1)}{\sqrt{1 + \arctan(\varphi \tilde{q}_1)}} \\ \frac{\tanh(\varphi \tilde{q}_2)}{\sqrt{1 + \arctan(\varphi \tilde{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \tilde{q}_n)}{\sqrt{1 + \arctan(\varphi \tilde{q}_n)}} \end{bmatrix} - K_v(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) \begin{bmatrix} \frac{\tanh(\varphi \dot{q}_1)}{\sqrt{1 + \arctan(\varphi \dot{q}_1)}} \\ \frac{\tanh(\varphi \dot{q}_2)}{\sqrt{1 + \arctan(\varphi \dot{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \dot{q}_n)}{\sqrt{1 + \arctan(\varphi \dot{q}_n)}} \end{bmatrix} + \mathbf{g}(\mathbf{q}) \quad (6)$$

Donde $\tilde{\mathbf{q}} = \mathbf{q}_d - \mathbf{q}$, siendo $\mathbf{q}_d \in \mathbb{R}^n$ el vector de posición deseadas y $\tilde{\mathbf{q}} \in \mathbb{R}^n$ el vector de error de posición y $\varphi \in \mathbb{N}$ es un entero positivo. Las matrices de ganancias variables $K_p(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ y $K_v(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$, son diagonales con las entradas $k_{pi}(\tilde{q}_i, \dot{q}_i)$ y $k_{vi}(\tilde{q}_i, \dot{q}_i)$, respectivamente. El vector de funciones que forman al esquema de control (6), cumplen lo siguiente:

$$\left\| \begin{bmatrix} \frac{\tanh(\varphi \tilde{q}_1)}{\sqrt{1 + \arctan(\varphi \tilde{q}_1)}} \\ \frac{\tanh(\varphi \tilde{q}_2)}{\sqrt{1 + \arctan(\varphi \tilde{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \tilde{q}_n)}{\sqrt{1 + \arctan(\varphi \tilde{q}_n)}} \end{bmatrix} \right\| \leq \frac{\sqrt{2}}{\sqrt{(\pi + 2)}} \sqrt{n} \rho_{Max}, \forall \mathbf{x} \in \mathbb{R}^n \quad (7)$$

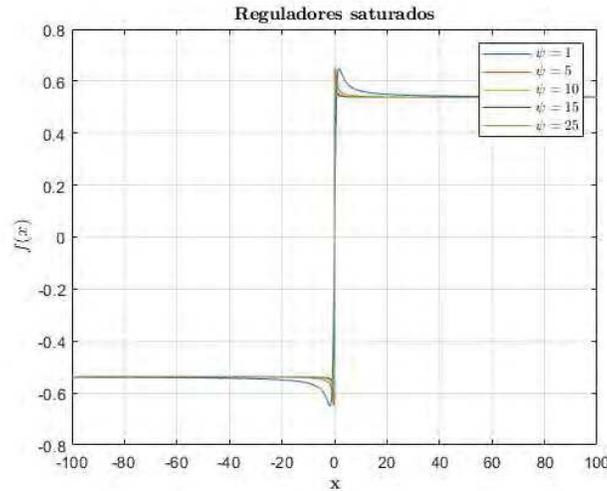


Figura 1. Familia de reguladores saturados.

En la Figura 1, se muestra la familia de reguladores saturados que tienen la propiedad de satisfacer (7), donde la constante ρ_{Max} representa el valor numérico de la cota superior máximo (valor pico máximo) y el cual depende de φ . De manera que aumenta este valor el rizo del regulador disminuye y el pico más alto aumenta con una pendiente más vertical. Este comportamiento es aprovechado para impulsar al error de posición del robot hacia cero y obtener energía para inyectar amortiguamiento y disminuir sobretiro en estado transitorio.

3.1 Familia extensa de ganancias variables

El trabajo [28] presentado a la comunidad científica, se volvió fuente de inspiración para diseñar una familia extensa de ganancias variables con sintonía automática (8) y (9), proporcional y derivativa; dependientes del error de posición y la velocidad de movimiento del robot manipulador.

$$K_p(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) = \begin{bmatrix} k_{p1}(\tilde{q}_1, \dot{q}_1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & k_{pn}(\tilde{q}_n, \dot{q}_n) \end{bmatrix} \quad (8)$$

$$K_v(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) = \begin{bmatrix} k_{v1}(\tilde{q}_1, \dot{q}_1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & k_{vn}(\tilde{q}_n, \dot{q}_n) \end{bmatrix} \quad (9)$$



Siendo las entradas $k_{pi}(\tilde{q}_i, \dot{q}_i)$ y $k_{vi}(\tilde{q}_i, \dot{q}_i)$ de la matrices de ganancia proporcional y derivativa variables, respectivamente y con la estructura matemática:

$$k_{pi}(\tilde{q}_i, \dot{q}_i) = \hat{k}_{pi}(1 - \beta e^{-\beta f_{pi}(\tilde{q}_i, \dot{q}_i)}) \quad (10)$$

$$k_{vi}(\tilde{q}_i, \dot{q}_i) = \hat{k}_{vi}(1 - \beta e^{-\beta f_{vi}(\tilde{q}_i, \dot{q}_i)}) \quad (11)$$

Las funciones $f_{pi}(\tilde{q}_i, \dot{q}_i)$ y $f_{vi}(\tilde{q}_i, \dot{q}_i) \in \mathbb{R}^2 \rightarrow \mathbb{R}_+$ y $f_{pi}(\tilde{q}_i, \dot{q}_i) = f_{vi}(\tilde{q}_i, \dot{q}_i) = 0 \Leftrightarrow \tilde{q}_i = \dot{q}_i = 0$. Los parámetros $\hat{k}_{pi} > 0$, $\hat{k}_{vi} > 0$ y $\beta \in [0,1]$. Entonces, bajo las condiciones anteriores, observe que: $k_{pi}(\tilde{q}_i, \dot{q}_i) > 0$ y $k_{vi}(\tilde{q}_i, \dot{q}_i) > 0$ y por lo tanto $K_p(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) > 0$, $K_v(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) > 0$.

A partir del modelo dinámico del robot manipulador (1) y nuestro regulador propuesto (6) se construye la ecuación en lazo cerrado (12), la cual es una ecuación diferencial autónoma no lineal.

$$\frac{d}{dt} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} -\dot{\mathbf{q}} \\ M(\mathbf{q})^{-1} \left[K_p(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) \begin{bmatrix} \frac{\tanh(\varphi \tilde{q}_1)}{\sqrt{1 + \arctan(\varphi \tilde{q}_1)}} \\ \frac{\tanh(\varphi \tilde{q}_2)}{\sqrt{1 + \arctan(\varphi \tilde{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \tilde{q}_n)}{\sqrt{1 + \arctan(\varphi \tilde{q}_n)}} \end{bmatrix} - K_v(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) \begin{bmatrix} \frac{\tanh(\varphi \dot{q}_1)}{\sqrt{1 + \arctan(\varphi \dot{q}_1)}} \\ \frac{\tanh(\varphi \dot{q}_2)}{\sqrt{1 + \arctan(\varphi \dot{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \dot{q}_n)}{\sqrt{1 + \arctan(\varphi \dot{q}_n)}} \end{bmatrix} - B\dot{\mathbf{q}} - C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \right] \end{bmatrix} \quad (12)$$

Nótese que: $\frac{d}{dt} \tilde{\mathbf{q}} = (-I)\dot{\mathbf{q}} = \mathbf{0} \in \mathbb{R}^n \Leftrightarrow \dot{\mathbf{q}} = \mathbf{0}$, siendo $I \in \mathbb{R}^{n \times n}$ la matriz identidad. También, $\frac{d}{dt} \dot{\mathbf{q}} = \mathbf{0} \in \mathbb{R}^n \Leftrightarrow \tilde{\mathbf{q}} = \mathbf{0}$ y $\dot{\mathbf{q}} = \mathbf{0}$. Por lo tanto, queda demostrado que el origen de la ecuación en lazo cerrado es el único punto de equilibrio.

3.2 Función estricta de Lyapunov

Para demostrar la estabilidad asintótica y global del punto de equilibrio de la ecuación en lazo cerrado (12) desde el contexto del método directo de Lyapunov, presentamos la siguiente función de energía estricta:

$$V(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T M(\mathbf{q}) \dot{\mathbf{q}} + \int_0^{\tilde{\mathbf{q}}} \begin{bmatrix} \frac{\tanh(\varphi \sigma_1)}{\sqrt{1 + \arctan(\varphi \sigma_1)}} \\ \frac{\tanh(\varphi \sigma_2)}{\sqrt{1 + \arctan(\varphi \sigma_2)}} \\ \vdots \\ \frac{\tanh(\varphi \sigma_n)}{\sqrt{1 + \arctan(\varphi \sigma_n)}} \end{bmatrix}^T K_p(\boldsymbol{\sigma}, \dot{\mathbf{q}}) d\boldsymbol{\sigma} - \boldsymbol{\epsilon} \tilde{\mathbf{q}}^T M(\mathbf{q}) \dot{\mathbf{q}} \quad (13)$$



$$\int_0^{\tilde{q}} \begin{bmatrix} \frac{\tanh(\varphi\sigma_1)}{\sqrt{1+\arctan(\varphi\sigma_1)}} \\ \frac{\tanh(\varphi\sigma_2)}{\sqrt{1+\arctan(\varphi\sigma_2)}} \\ \vdots \\ \frac{\tanh(\varphi\sigma_n)}{\sqrt{1+\arctan(\varphi\sigma_n)}} \end{bmatrix}^T K_p(\sigma, \dot{q}) d\sigma = \sum_{i=0}^n \frac{\tanh(\sigma_i)}{\sqrt{1+\arctan(\sigma_i)}} k_{pi}(\tilde{q}_i, \dot{q}_i) d\sigma_i \quad (14)$$

Por notación, la integral se describe en (14) y ϵ se define de la siguiente manera:

$$\epsilon(\|\tilde{q}\|) = \frac{\epsilon_0}{1 + \|\tilde{q}\|} \quad (15)$$

Donde ϵ_0 es una constante positiva y ϵ es lo suficientemente pequeño para satisfacer lo siguiente:

$$\frac{\sqrt{k_{pl}\lambda_{\min}\{M(q)\}}}{\lambda_{\max}\{M(q)\}} > \epsilon > 0 \quad (16)$$

Las cotas k_{pl} y k_{pu} son constantes y $k_{pi}(\tilde{q}_i, \dot{q}_i) > 0$ y $k_{vi}(\tilde{q}_i, \dot{q}_i) > 0$, siendo $K_p(\tilde{q}, \dot{q})$ una matriz diagonal, por lo tanto (16) es una condición suficiente para que (13) sea una función definida positiva.

Demostración: Considerando lo siguiente:

$$\int_0^{\tilde{q}} \begin{bmatrix} \frac{\tanh(\varphi\sigma_1)}{\sqrt{1+\arctan(\varphi\sigma_1)}} \\ \frac{\tanh(\varphi\sigma_2)}{\sqrt{1+\arctan(\varphi\sigma_2)}} \\ \vdots \\ \frac{\tanh(\varphi\sigma_n)}{\sqrt{1+\arctan(\varphi\sigma_n)}} \end{bmatrix}^T K_p(\sigma, \dot{q}) d\sigma \geq \frac{1}{2} k_{pl} \left\| \begin{bmatrix} \frac{\tanh(\varphi\tilde{q}_1)}{\sqrt{1+\arctan(\varphi\tilde{q}_1)}} \\ \frac{\tanh(\varphi\tilde{q}_2)}{\sqrt{1+\arctan(\varphi\tilde{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi\tilde{q}_n)}{\sqrt{1+\arctan(\varphi\tilde{q}_n)}} \end{bmatrix} \right\|^2 \geq \frac{1}{2} k_{pl} \|\tilde{q}\|^2 \quad (17)$$

Entonces, la función estricta de Lyapunov está acotada de la siguiente manera:

$$V(\tilde{q}, \dot{q}) \geq \frac{1}{2} k_{pl} \|\tilde{q}\|^2 + \lambda_{\min}\{M(q)\} \|\tilde{q}\|^2 - \epsilon \lambda_{\max}\{M(q)\} \|\tilde{q}\| \|\dot{q}\| \quad (18)$$

$$V(\tilde{q}, \dot{q}) \geq \frac{1}{2} \begin{bmatrix} \|\tilde{q}\| \\ \|\dot{q}\| \end{bmatrix}^T P \begin{bmatrix} \|\tilde{q}\| \\ \|\dot{q}\| \end{bmatrix} \quad (19)$$

La matriz P se compone:

- $P(1,1) = k_{pl}$
- $P(1,2) = \epsilon \lambda_{\max}\{M(q)\}$
- $P(2,1) = P(1,2)$
- $P(2,2) = \lambda_{\min}\{M(q)\}$

Si ϵ es elegido, de manera que pueda satisfacer (16), entonces (13) es una función definida positiva.



3.3 Derivada con respecto al tiempo de la función estricta

Derivando con respecto al tiempo la función estricta de Lyapunov (13) y usando la regla de derivación de integrales de Leibniz, obtenemos:

$$\begin{aligned} \dot{V}(\tilde{q}, \dot{q}) = & \dot{q}^T \dot{M}(q) \dot{q} + \frac{1}{2} \dot{q}^T \dot{M}(q) \dot{q} + \begin{bmatrix} \frac{\tanh(\varphi \tilde{q}_1)}{\sqrt{1 + \arctan(\varphi \tilde{q}_1)}} \\ \frac{\tanh(\varphi \tilde{q}_2)}{\sqrt{1 + \arctan(\varphi \tilde{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \tilde{q}_n)}{\sqrt{1 + \arctan(\varphi \tilde{q}_n)}} \end{bmatrix}^T K_p(\tilde{q}, \dot{q}) \tilde{q} - \epsilon \dot{q}^T \dot{M}(q) \tilde{q} \\ & - \epsilon \dot{q}^T \dot{M}(q) \dot{q} - \epsilon \dot{q}^T M(q) \tilde{q} - \epsilon \dot{q}^T M(q) \dot{q} \end{aligned} \quad (20)$$

Tomando en cuenta las trayectorias de la ecuación en lazo cerrado (12) y la Propiedad 1 (anti simetría), la expresión anterior se reescribe como se puede ver en la expresión (21).

$$\begin{aligned} \dot{V}(\tilde{q}, \dot{q}) = & -\dot{q}^T K_v(\tilde{q}, \dot{q}) \begin{bmatrix} \frac{\tanh(\varphi \dot{q}_1)}{\sqrt{1 + \arctan(\varphi \dot{q}_1)}} \\ \frac{\tanh(\varphi \dot{q}_2)}{\sqrt{1 + \arctan(\varphi \dot{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \dot{q}_n)}{\sqrt{1 + \arctan(\varphi \dot{q}_n)}} \end{bmatrix} - \dot{q}^T B \dot{q} - \\ & - \epsilon \begin{bmatrix} \frac{\tanh(\varphi \tilde{q}_1)}{\sqrt{1 + \arctan(\varphi \tilde{q}_1)}} \\ \frac{\tanh(\varphi \tilde{q}_2)}{\sqrt{1 + \arctan(\varphi \tilde{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \tilde{q}_n)}{\sqrt{1 + \arctan(\varphi \tilde{q}_n)}} \end{bmatrix}^T K_p(\tilde{q}, \dot{q}) \tilde{q} + \epsilon \tilde{q}^T K_v(\tilde{q}, \dot{q}) \begin{bmatrix} \frac{\tanh(\varphi \dot{q}_1)}{\sqrt{1 + \arctan(\varphi \dot{q}_1)}} \\ \frac{\tanh(\varphi \dot{q}_2)}{\sqrt{1 + \arctan(\varphi \dot{q}_2)}} \\ \vdots \\ \frac{\tanh(\varphi \dot{q}_n)}{\sqrt{1 + \arctan(\varphi \dot{q}_n)}} \end{bmatrix} + \\ & + \epsilon \tilde{q}^T B \dot{q} + \epsilon \tilde{q}^T C(q, \dot{q})^T \dot{q} - \epsilon \tilde{q}^T \dot{M}(q) \dot{q} + \\ & + \dot{q}^T M(q) \dot{q} - \epsilon \dot{q}^T M(q) \dot{q} \end{aligned} \quad (21)$$

Usando la Propiedad 2, es posible realizar lo que se muestra en (22).

$$\begin{aligned} -\epsilon \tilde{q}^T C(q, \dot{q})^T \dot{q} = & |-\epsilon \dot{q}^T C(q, \dot{q}) \tilde{q}| \\ \leq & \epsilon_0 k_{r0} \|\dot{q}\|^2 \end{aligned} \quad (22)$$

Tomando en cuenta la estructura matemática (15) su derivada con respecto al tiempo es (23) y podemos acotar como se muestra en (24).

$$\dot{\epsilon} = \frac{\epsilon_0}{\|\tilde{q}\|(1 + \|\tilde{q}\|)^2} \tilde{q}^T \dot{q} \quad (23)$$

$$\dot{\epsilon} \dot{q}^T \dot{M}(q) \dot{q} = \frac{\epsilon_0}{\|\tilde{q}\|(1 + \|\tilde{q}\|)^2} \tilde{q}^T \dot{q} \dot{q}^T M(q) \dot{q} \leq \epsilon_0 \lambda_{Max}\{M(q)\} \|\dot{q}\|^2 \quad (24)$$

Y finalmente, la derivada de la función estricta de Lyapunov está acotada en su forma cuadrática como se muestra en (25):

$$\dot{V}(\tilde{q}, \dot{q}) \leq - \begin{bmatrix} \|\tilde{q}\| \\ \|\dot{q}\| \end{bmatrix}^T Q \begin{bmatrix} \|\tilde{q}\| \\ \|\dot{q}\| \end{bmatrix} \quad (25)$$



La matriz Q se compone:

- $Q(1,1) = \frac{\epsilon_0}{1+|\tilde{q}|} \lambda_{\min}\{K_v(\tilde{q}, \dot{q})\}$
- $Q(1,2) = \frac{\epsilon_0}{2} [\lambda_{\max}\{K_v(\tilde{q}, \dot{q})\} + \lambda_{\max}\{B\}]$
- $Q(2,1) = Q(1,2)$
- $Q(2,2) = \lambda_{\min}\{K_v(\tilde{q}, \dot{q})\} + \lambda_{\min}\{B\} - \epsilon_0 k_{ro} - 2 \epsilon_0 \lambda_{\max}\{M(q)\}$

Por lo tanto, las condiciones son:

$$\frac{a_1 + a_2}{[1 + |\tilde{q}|]b_1 + b_2} > \epsilon_0 \geq \epsilon \quad (26)$$

- $a_1 = 4 \lambda_{\min}\{K_v(\tilde{q}, \dot{q})\} \lambda_{\min}\{K_v(\tilde{q}, \dot{q})\}$
- $a_2 = \lambda_{\max}\{B\}$
- $b_1 = \lambda_{\max}\{K_v(\tilde{q}, \dot{q})\} + \lambda_{\max}\{B\}$
- $b_2 = 4 \lambda_{\min}\{K_v(\tilde{q}, \dot{q})\} [\epsilon_0 k_{ro} + 2 \lambda_{\max}\{M(q)\}]$

La elección de ϵ está en función de satisfacer (16) y (26) de manera simultánea, obtener su valor numérico no es necesario para la familia de algoritmos de control y sólo es necesario para propósitos de la demostración de estabilidad asintótica del punto de equilibrio del sistema en lazo cerrado en el espacio de estados. Dicho lo anterior, concluimos que el punto de equilibrio de la ecuación en lazo cerrado (11) es asintóticamente estable y global, desde el contexto de la teoría de estabilidad de Lyapunov.

3.3 Diseño de las matrices de ganancia

Con fines de evaluar a la familia grande de ganancias variables (7) y (8), proponemos los siguientes casos particulares de $f_{pi}(\tilde{q}_i, \dot{q}_i)$ y $f_{vi}(\tilde{q}_i, \dot{q}_i)$:

$$\begin{aligned} k_{pi}(\tilde{q}_i, \dot{q}_i) &= \hat{k}_{pi} \left(1 - \beta e^{-\beta \tanh(\tilde{q}_i^2 + \dot{q}_i^2)} \right) \\ k_{vi}(\tilde{q}_i, \dot{q}_i) &= \hat{k}_{vi} (1 - \beta e^{-\beta \tanh(\tilde{q}_i^2 + \dot{q}_i^2)}) \end{aligned} \quad (27)$$

$$\begin{aligned} k_{pi}(\tilde{q}_i, \dot{q}_i) &= \hat{k}_{pi} \left(1 - \beta e^{-\beta (\sinh(\tilde{q}_i)^2 + \cosh(\dot{q}_i)^2)} \right) \\ k_{vi}(\tilde{q}_i, \dot{q}_i) &= \hat{k}_{vi} (1 - \beta e^{-\beta (\sinh(\tilde{q}_i)^2 + \cosh(\dot{q}_i)^2)}) \end{aligned} \quad (28)$$

Siendo $\hat{k}_{pi} = \delta_i \sqrt{\tau_i^{Max}}$, $\hat{k}_{vi} = \gamma_i \sqrt{\tau_i^{Max} + 1}$, $\delta_i > 0$, $\gamma_i > 0$ y τ_i^{Max} es el par i -ésimo máximo aplicado por el servomotor del robot manipulador.

En la Figura 2 y la Figura 3, nos muestra el perfil de ganancia variable de los casos particulares (26) y (27), respectivamente. Nótese que, no difieren del todo. Esto nos permite elegir una gama grande de casos particulares $f_{pi}(\tilde{q}_i, \dot{q}_i)$ y $f_{vi}(\tilde{q}_i, \dot{q}_i)$ sin el riesgo de afectar la sintonía automática de las ganancias variables. La sintonía automática es una consecuencia de las propiedades de estabilidad asintótica y global del punto de equilibrio; todas las soluciones dentro del atractor llegan asintóticamente al punto de equilibrio. Dicho lo anterior, gráficamente se puede notar que, el error de posición y la velocidad de movimiento está cerca del punto de equilibrio, la sintonía automática se da de manera natural.

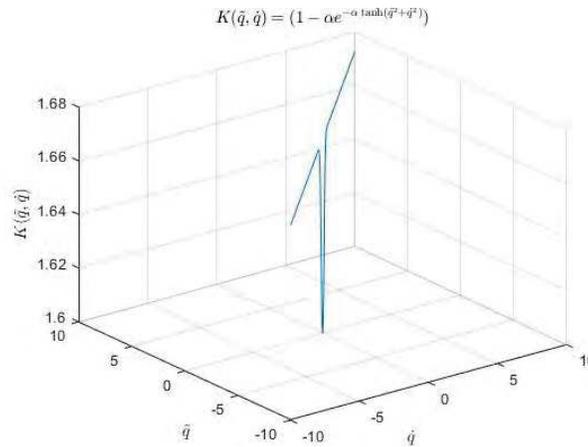


Figura 2. Perfil de primer caso particular de ganancia variables.

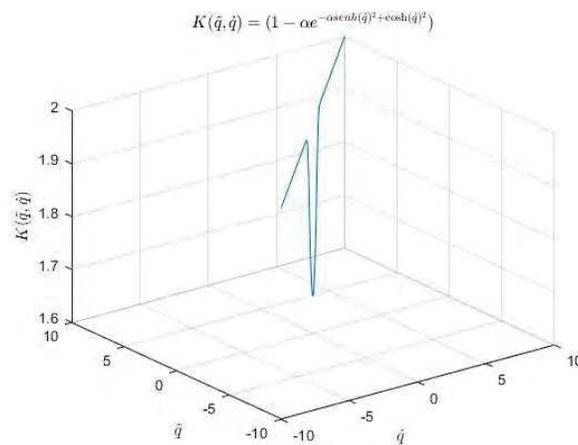


Figura 3. Perfil de segundo caso particular de ganancia variable.

4. Resultados

Con propósitos de evaluar a nuestro regulador acotado y a la familia de ganancias variables, realizamos la estrategia de regulación de un robot antropomórfico de dos grados de libertad a nivel simulación usando el software **MATLAB**. Este enfoque nos permite evaluar los fenómenos físicos del robot manipulador, sin necesidad de tenerlo construido físicamente y obtener resultados muy cercanos a los que se pueden esperar experimentalmente. La idea central es comparar su desempeño con reguladores conocidos en la literatura como son el PD y el tangente hiperbólico presentado en [29].

El experimento virtual consiste en llevar a las posiciones articulares del hombro y el codo del robot manipulador (ver Fig. 4) desde su posición de casa $[0,0]^T$ grados, hacia las posiciones deseadas $[45,90]^T$ grados. Usamos los reguladores PD y el tangente hiperbólico sintonizando sus ganancias de manera empírica. Para nuestra propuesta, consideramos dos casos particulares de la familia de reguladores con $\varphi = 5$ y $\varphi = 10$; también los dos casos particulares de ganancias variables (27) y (28). Los parámetros numéricos del robot de dos grados de libertad se han obtenido mediante los experimentos de identificación paramétrica presentados en [3]. Finalmente, tomamos en cuenta el torque máximo proporcionado por los actuadores del robot manipulador $\tau_1^{Max} = 150 Nm$ y $\tau_2^{Max} = 15 Nm$ del hombro y del codo, respectivamente.



Figura 4. Robot manipulador de 2 grados de libertad.

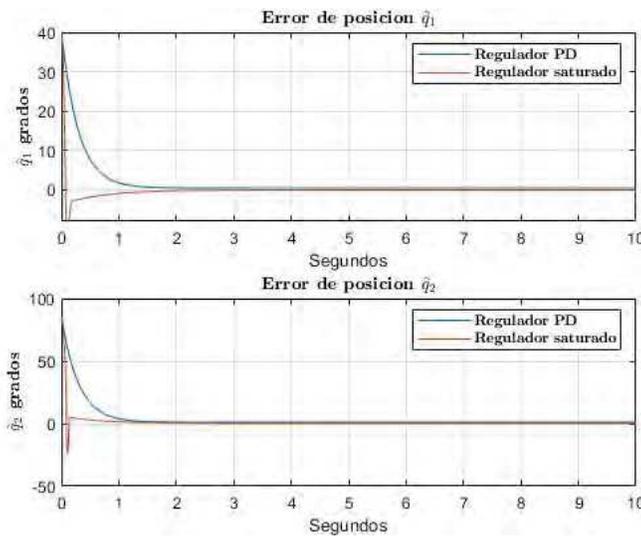


Figura 5. Error de posición del robot de dos grados de libertad.

En la Figura 5, se muestran los errores de posiciones del robot manipulador por medio del regulador PD y el regulador acotado con $\varphi = 5$ y el caso particular de ganancias variables (27). Como se puede observar, con ambos reguladores los errores de posición del robot manipulador llegan asintóticamente a cero. De igual manera, las velocidades de movimiento llegan asintóticamente a cero, tal y como se muestra en la Figura 6. Por medio de simulación, se ha podido mostrar que nuestra estrategia de control propuesta logra llevar al manipulador asintóticamente al punto de equilibrio de la ecuación en lazo cerrado. Realizando la comparativa entre el regulador PD y el regulador acotado; el PD lleva al robot desde las posiciones de casa hacia las posiciones deseadas, con una respuesta sobre amortiguada debido a la acción de control derivativa a través de la velocidad de movimiento, la cual inyecta amortiguamiento para eliminar sobre impulsos y una llegada suave a estado estacionario.

Por otro lado, nuestra propuesta también logra dar solución al problema de regulación; sin embargo, se tiene una respuesta sub amortiguada en estado transitorio y con una llegada más rápida y suave a estado estacionario. Esto es gracias a la acción de control de amortiguamiento por medio de la velocidad de movimiento. El pico máximo de la velocidad, es aprovechado por el esquema de control para la inyección de un freno mecánico al robot manipulador para llegar a las posiciones deseadas. Las ganancias variables, se sintonizan de manera automática por las propiedades de estabilidad asintótica del punto de equilibrio; es decir, se resuelve el problema de control de posición sin pasar por el proceso complicado y empírico de la sintonía de ganancias, como en el caso del regulador PD.

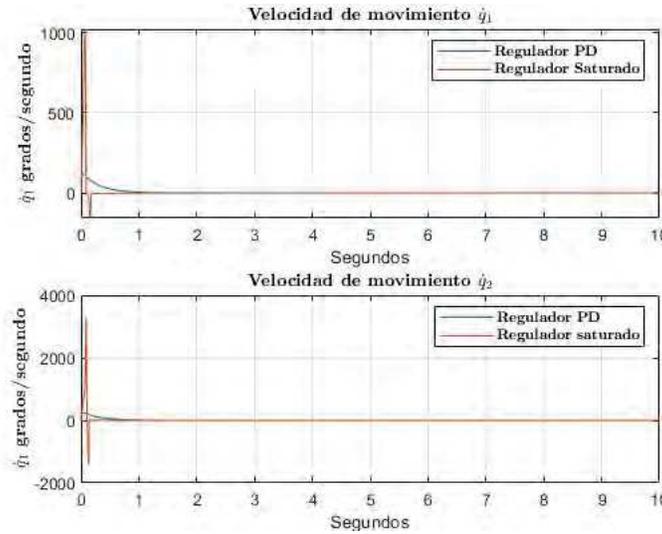


Figura 6. Velocidad de movimiento del robot de dos grados de libertad.

Para continuar, mantenemos el caso particular de la familia de reguladores saturados con $\varphi = 5$ modificando al segundo caso particular de ganancias variables (28). En la Figura 7, se muestran los errores de posición del robot manipulador, dándonos evidencia gráfica que se llega a cero. La velocidad de movimiento, también llega a cero de forma asintótica (ver. Figura 8). Es decir, se llega asintóticamente al punto de equilibrio de la ecuación en lazo cerrado. Lo anterior se cumple para ambos reguladores. Es importante resaltar que, a pesar de modificar la estructura de las ganancias variables, nuestra propuesta sigue llevando al robot manipulador hacia las posiciones deseadas. Incluso, se obtiene una mejora con respecto al caso particular anterior. Se llega más rápido a estado estacionario, con una respuesta sub amortiguada a la inyección de amortiguamiento del esquema de control, con respecto al regulador PD. La velocidad de movimiento, una vez más es aprovechada por la estructura propuesta para una llegada suave a estado estacionario. Lo resultados obtenidos en este escenario, nos pueden brindar una prueba importante que, la elección de $f_{pi}(\tilde{q}_i, \dot{q}_i)$ y $f_{vi}(\tilde{q}_i, \dot{q}_i)$ es clave para obtener mejoras en la respuesta del robot manipulador. En suma, se han abordado de manera eficiente el problema de regulación sin pasar por ningún proceso empírico y complejo de sintonía de las ganancias del regulador saturado.

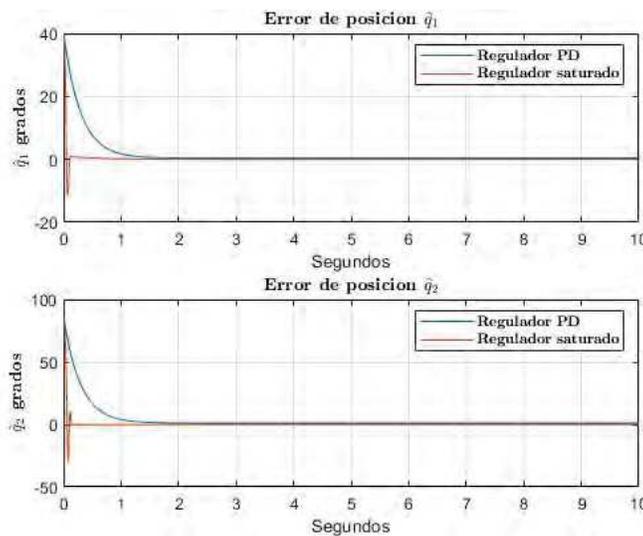


Figura 7. Error de posición del robot manipulador de dos grados de libertad.

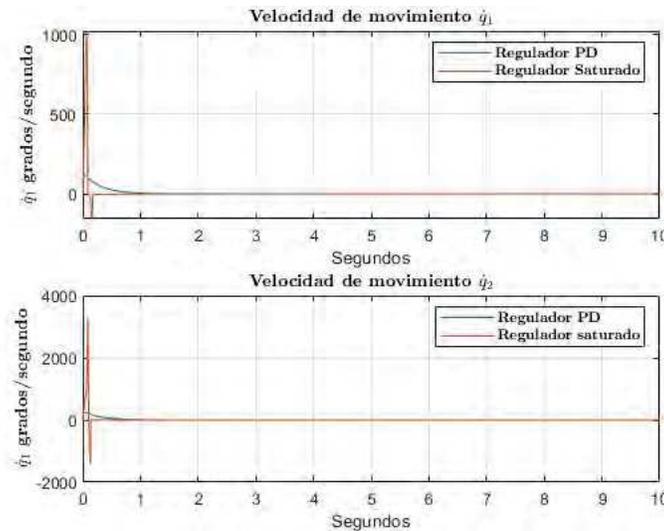


Figura 8. Velocidad de movimiento del robot de dos grados de libertad.

Con fines de evaluar de seguir evaluando a la familia de reguladores saturados con ganancias variables, consideramos que es importante realizar una comparación con algún regulador del tipo saturado. El tangente hiperbólico presentado en [1] es uno de los más conocidos que pertenecen a esta clasificación. Para esto, tomamos otro caso particular de la familia con $\varphi = 10$. Consideramos una vez más al caso particular de ganancias variables (27). Las ganancias del regulador hiperbólico, se han sintonizado a experiencia.

Bajo las condiciones anteriores, obtenemos las respuestas del error de posición y la velocidad de movimiento del robot manipulador. Ambos reguladores llevan al manipulador al punto de equilibrio de la ecuación en lazo cerrado asintóticamente, en base a lo obtenido en las Figuras 9 y 10.

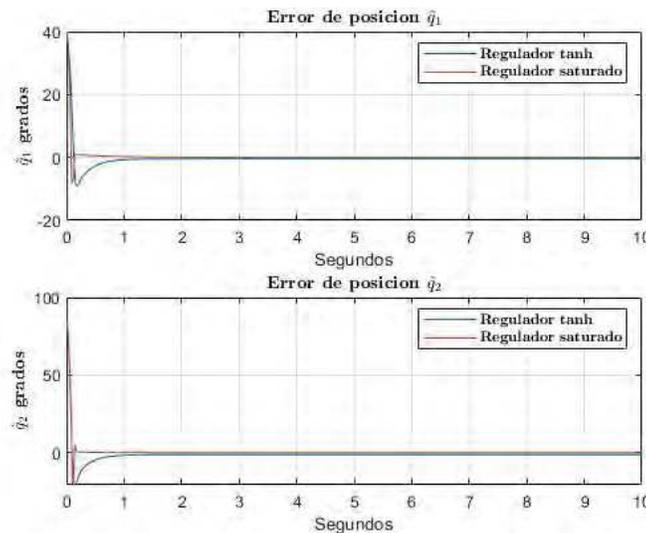


Figura 9. Error de posición del robot manipulador de dos grados de libertad.

Como podemos notar en la Figura 9, se pueden resaltar ciertas diferencias en la respuesta del error de posición articular. Nuestro caso particular de la familia de reguladores acotados, llega a estado estacionario en menos tiempo y con una respuesta sub amortiguada por medio de la acción de control de amortiguamiento por medio de la velocidad de movimiento (su comportamiento se muestra en la



Figura 10), la cual es aprovechada para la inyección de dicha energía. En términos del estado transitorio, el regulador saturado se mantiene en menos tiempo en dicha etapa, en comparación del tangente hiperbólico. Finalmente, para este escenario recalcar que se ha dado solución al problema de regulación con la sintonía de las ganancias variables de forma automática. Mientras que el tangente hiperbólico, ha pasado por un proceso de sintonía de ganancias complicado y empírico.

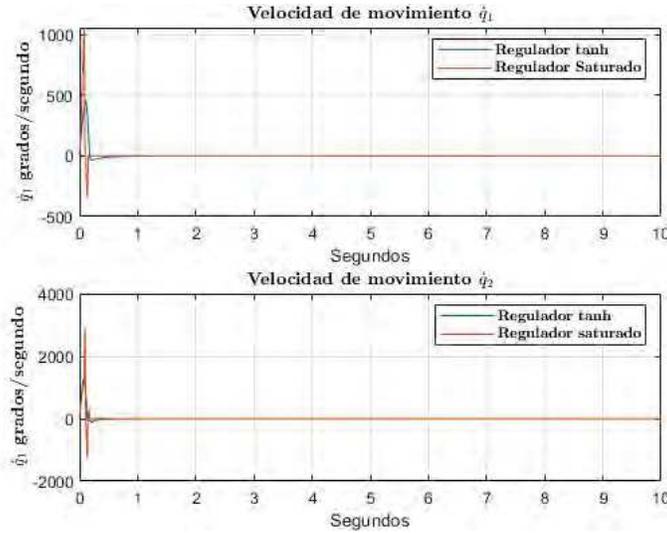


Figura 10. Velocidad de movimiento del robot de dos grados de libertad.

Para finalizar, modificamos una vez más el caso particular de ganancias variables. Probamos con la estructura (28) manteniendo a $\varphi = 10$. Dicho lo anterior, obtenemos los resultados del error de posición y de velocidad de movimiento del robot manipulador. Estas respuestas, las cuales se ilustran en la Figura 11 y la Figura 12, respectivamente; demuestran a nivel simulación que el caso particular de la familia de reguladores acotados, bajo las condiciones antes mencionadas, llevan al manipulador al punto de equilibrio de la ecuación en lazo cerrado. Lo anterior también se cumple para el regulador tangente hiperbólico.

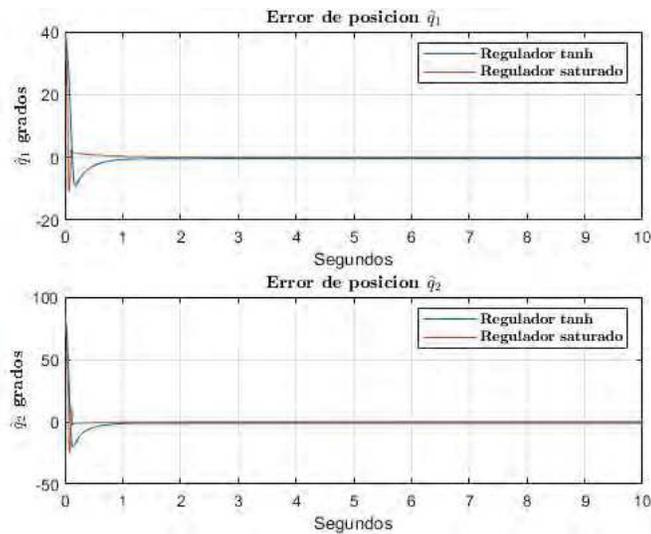


Figura 11. Error de posición del robot manipulador de dos grados de libertad.



Nuevamente, el cambio de caso particular de ganancias variables mejora la respuesta de los errores de posición del robot manipulador. La llegada a estado estacionario es rápida y suave, la respuesta en etapa transitoria es sub amortiguada y con muy poca duración para llegar a las posiciones deseadas debido a la acción de control de amortiguamiento, de acuerdo a lo mostrado a la Figura 1. Por último, las respuestas de las velocidades de movimiento son aprovechadas por nuestro regulador acotado para inyectar la energía necesaria para llevar a las posiciones a las posiciones deseadas. Este comportamiento se presenta en la Figura 12. La sintonía automática de las ganancias es parte fundamental para la obtención favorable de los resultados mostrados en este escenario; se soluciona el problema de control de posición y se sintonizan a las ganancias usando un enfoque científico.

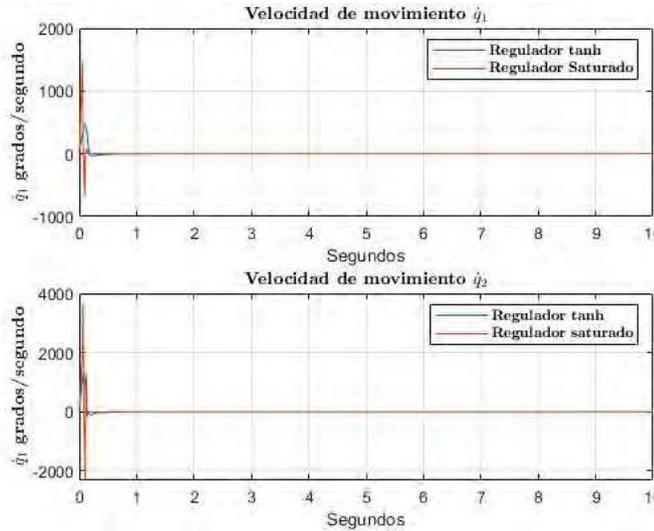


Figura 12. Velocidad de movimiento del robot de dos grados de libertad.

No obstante, no es posible decir contundentemente que nuestra propuesta tiene mejor desempeño en comparación de los reguladores PD y el tangente hiperbólico, en función de los resultados anteriormente presentados. Dar un veredicto es subjetivo y queda a criterio del lector. En este sentido, usamos a la norma \mathcal{L}_2 como herramienta para la medición del desempeño de los reguladores analizados en esta sección.

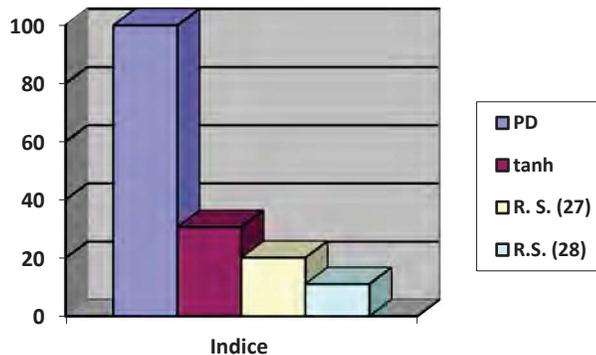


Figura 13. índice de desempeño de los reguladores.

En la Figura 13, se muestra el desempeño del regulador PD, el tangente hiperbólico, el regulador saturado con $\varphi = 5$ y los casos particulares de ganancias variables (27) y (28). El tangente hiperbólico tiene una mejora en su desempeño de 69.33 % en comparación con el regulador PD. Para nuestra propuesta, el regulador saturado con el caso particular (27) tiene mejora del 79.88 % con respecto al PD y del 10.55 % con respecto al tangente hiperbólico. Modificando el caso particular de ganancias



variables a (28), se tiene un mejor desempeño; un 88.93 % con respecto al PD y un 19.60 % en comparación al regulador hiperbólico. Desde este contexto, se ha mostrado que, este caso particular de la familia de reguladores saturados y de ganancias variables, tiene mejor desempeño de regulación que los otros dos reguladores en cuestión.

Para el siguiente escenario, consideramos ahora al regulador saturado con $\varphi = 10$ y los casos particulares (27) y (28) de ganancias variables. Para este escenario, nuestro regulador propuesto con ganancias variables (27) tiene una mejora de desempeño del 84.88% comparándolo con el PD y del 15.55% con respecto al hiperbólico. Realizando el cambio a (27) de ganancias variables, se tiene un mejor desempeño de 92.33% contra el PD y 20.01% comparado con el regulador tangente hiperbólico. Lo anterior se puede ver en la Figura 14. En suma, ambos casos particulares de la familia extensa de los reguladores aquí propuestos resuelven; el problema de regulación, la sintonía automática de las ganancias variables (presentando algunos casos particulares) y con un mejor índice de desempeño desde el contexto de la norma \mathcal{L}_2 .

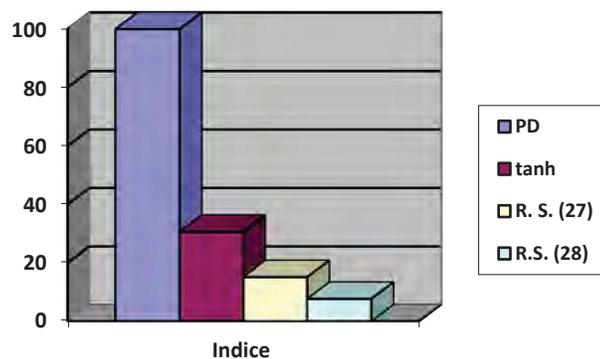


Figura 14. índice de desempeño de los reguladores.

5. Conclusiones

Para concluir, la propuesta de una familia de reguladores saturados con ganancias variables da solución al problema de regulación, genera un atractor con propiedades de estabilidad asintótica del punto de equilibrio de la ecuación en lazo cerrado, demostrando dicha afirmación mediante la teoría de estabilidad de Lyapunov. También, resaltamos la aportación de la familia de ganancias variables presentada en este trabajo; dándonos resultados positivos para la familia de reguladores saturados, de su posible acoplamiento a otros reguladores que existen en la literatura para dar solución a la sintonía de las ganancias desde un criterio científico y mejorar su desempeño de una manera importante. Sin embargo, elegir el caso particular en función del desempeño es queda abierto y dependerá de la propuesta del diseñador. Finalmente, obtener resultados a nivel simulación dan un respaldo importante a lo presentado en este trabajo. No obstante, los trabajos experimentales le aportan rigor científico. Esto nos motiva a presentar trabajar futuros con resultados muy similares para un robot manipulador de transmisión directa de tres grados de libertad.

Referencias

- [1] Reyes, F., Felix O., Cid, J., Alonso, G., "A Family of Hyperbolic-Type Control Schemes for Robot Manipulators", *Kybernetika*, Volume 55, pp. 561-585, (2019).
- [2] Limón, M. A., Reyes F., and Gonzales, E., "Saturated regulation with derivate variable gain for robot manipulators", *RIA Elsevier*, pp. 434-445, (2017).
- [3] Reyes, F., "Robótica Control de robots manipuladores", AlfaOmega, México, Primera edición, pp. 257, (2011).



- [4] Kelly, R. and Santibáñez, V., “Control of robot manipulators in joint space”, Springer, (2005).
- [5] Zavala, A., Santibáñez, V. “Output-feedback proporcional-integral-derivative-type control with multiple saturating structure for the global stabilization of robot manipulators with bounded inputs”. *International Journal Advanced Robotic Systems*, 6, pp. 1-12, (2016).
- [6] Santibáñez, V., Kelly, R., and Llama M. A., “A novel global asymptotic stable set-point fuzzy controller with bounded torques for robot manipulators”, *IEEE Trans. Fuzzy Systems* 13, 3, pp. 362-372, (2005).
- [7] Aguiñaga, E., Zavala, A., Santibáñez, V., and Reyes, F., “Global trajectory tracking through static feedback for robot manipulators with bounded inputs”, *IEEE Trans. Control Systems Technology*, 17, vol. 4, pp. 934-944, (2009).
- [8] Kelly R. and Santibáñez V. “Global regulation of elastic joint robots based on energy shaping”, *IEEE Trans. Automatic Control*, Vol.10, No 43, pp. 1887-1894, (1998).
- [9] Santibáñez V, Kelly R. and Reyes F. “A new set point controller with bounded torques for robot manipulators”, *IEEE Trans. Industrial Electronic*, Vol. 1, No 45, pp. 126-133, (1998).
- [10] Reyes, F., Cid, J., Limón, M. A., and Cervantes, M., *Square root-type control for robot manipulators*, *International Journal Advanced Robotic Systems*, No. 10, vol. 39, pp. 1-7, (2013).
- [11] Bai, E. W., and Huang Y. F., “Variable gain parameter estimation algorithms for fast tracking and smooth steady state”, *Automatica* 7, vol. 36, pp. 1001-1008, (2000).
- [12] Kiong, L. C., Mandava, R., Kiong W. E., and Rao, M., *A self-learning no-linear variable gain proportional-derivative (PD) controller in robot manipulators*, *Pertanika J. Sci. and Technol. Supplement*, vol. 12, No. 2, pp. 139-158, (2004).
- [13] Kelly R., Haber R, Haber-Guerra R and Reyes F. “Lyapunov Stable Control of Robot Manipulators: A Fuzzy Self-Tuning Procedure”, *Intelligent Automation and soft computing*, Vol 5, No. 4, pp. 313-326, (1999).
- [14] Zuo, Y., Wang, Y., Huang, L., and Li, C., “Intelligent Hybrid Control Strategy for Trajectory Tracking of robot manipulators”, *Journal of Control Science and Engineering*, vol. 2008, (2008).
- [15] Shang, W., Cong, Z., Xiang, Li and Long Jiang, S., Augmented, “Nolinear PD controller for a Redundantly Actuated Parallel Manipulator”, *Advanced Robotics*, vol. 23, pp. 1725-1742, (2009).
- [16] Kelly R. and Carelli R. “A class No linear PD-type Controllers for robot manipulators”, *Journal of Robotic Systems*, Vol. 13, pp. 793-802, (1996).
- [17] Salas., F., G., Santibáñez, V., and Llama, M. A., “Variable gains PD tracking control of robot manipulators: Stability analysis and simulations”, *World Automatic Congress (WAC)*, pp. 1-6, (2012).
- [18] Jafarov, E., Parlakci, M., and Istefanopulos, Y., “A new variable structure PID-controller design for robot manipulators, *Control System Technology*”, *IEEE Trans.*, Vol 13, No. 1, pp. 122-130, (2005).
- [19] Meza, J., Santibáñez V., Soto, R., Llama, M., “Stable fuzzy self-tuning PID control of robot manipulators”, *Systems, Man and Cybernetics IEEE International Conference*, pp. 2624-2629, (2009).
- [20] Dehghani, A., Khodadadi, H., “A fuzzy logic self-tuning PID control for a single-link flexible joint robot manipulator in the presence uncertainty”, *Automation and Systems 15th International Conference*, pp. 186-191, (2015).
- [21] Sifuentes, J., Santibáñez, V., Medina, J. L. M., “A globally asymptotically stable nonlinear PID regulator with fuzzy self-tuned PD gains for robot manipulators”, *World Automation Congress*, pp.573-778, (2014).
- [22] Meza, J., Santibáñez V., Soto, R., Llama, M., “Fuzzy self-tuning PID semiglobal regulator”, *Industrial Electronics, IEEE Transactions*, Vol, 56, No. 6, pp. 2624-2629, (2012).
- [23] Kumar, P. P., Kar, I., Behera, L., “Variable-gain controllers using T-S fuzzy model”, *Systems Man and Cybernetics, Part B: Cybernetics IEEE Trans.*, Vol.36, No. 6, pp. 1442-1449, (2006).
- [24] Xiaobo, G., Aiguo, S., and Yan Z., “Neural Network Control for Tele-rehabilitation based on variable gains”, *Biomedical Engineering and Informatics, International Conference*, vol. 2, pp. 778-782, (2008).
- [25] Walsh, A., and Forbes, J., “Analysis and synthesis of input strictly passive gain-schedule controllers”, *Journal of the Franklin Institute, Elsevier*, Vol. 354, pp. 1285-1201, (2018).
- [26] Limón, M. A., Reyes F., and Gonzales, E., “Unbounded regulators with variable gains for a direct-drive robot manipulator”, *Automatika: Journal for control, Measurement, Electronics, Computing and Communications Korema IFAC*, vol. 57, No. 2, pp. 484-495, (2016).



- [27] Reyes, F., Al-Hadithi, B. M., “*An asinh-type regulator for robot manipulators with global asymptotic stability*”, *Automatika*, Vol. 61, No. 4, pp. 574-586, (2020).
- [28] Reyes F., Cid J and Campuzano C. “*Development of an experimental platform with open architecture for robot manipulators*”, *IAESTED International Conference*, pp. 67-80, (1999).
- [29] Kelly R., Santibáñez V. and Reyes F. “*On Saturated-Proportional Derivative feedback with adaptive gravity compensation of robot manipulators*”, *International Journal Adaptive control and signal processing*, Vol. 10, pp. 465-479, (1996).



Errores en la relación de transmisión de los engranajes cilíndricos de dientes rectos

Reyes Méndez Alfonso Daniel, Félix Beltrán Olga Guadalupe, Mendoza Quiñonez
Jesús Alberto

Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Electrónica, Maestría en
Ciencias de la Electrónica, Opción en Automatización
alfonso.reyes@alumno.buap.mx

Resumen

En algunos mecanismos utilizados en mecatrónica la precisión es importante y solo gracias a ella el mecanismo puede cumplir su función. La bibliografía clásica no trata el modo cómo los errores de fabricación y montaje influyen en la cinemática funcional del mecanismo. En este trabajo, se muestra cómo los errores de fabricación asociados a los radios bases y a los ángulos de presión de una transmisión por engranajes influyen en el error de su relación de transmisión teórica, para la cual no se toma en cuenta estos errores. Para resolver este problema se plantea un mecanismo equivalente de cuatro barras que sustituye al modelo geométrico del piñón, del engrane y del bastidor que los soporta. Para ello se plantea las ecuaciones de restricción cinemática del mecanismo equivalente en el campo de las posiciones, tomando las diferenciales totales de estas ecuaciones para encontrar la relación entre los errores en la relación de transmisión y los errores en los parámetros geométricos del mecanismo equivalente de cuatro barras. Los resultados se muestran haciendo uso de MATLAB. Se obtiene que el error para la transmisión por engranajes aquí estudiada puede llegar a ser hasta de un 3 % cuando los grados de precisión IT de fabricación no son exigentes.

Palabras clave: Tolerancias dimensionales, relación de transmisión, involuta, transmisiones por engranajes cilíndricos rectos, errores en la relación de transmisión.

1. Introducción

Los engranes son elementos de máquinas que transmiten movimiento mediante sus dientes que embonan de una manera sucesiva transmitiendo movimiento de un eje giratorio a otro [1].

Existen numerosas aplicaciones de precisión en las que una razón de velocidad angular constante se debe transmitir de un eje a otro. Debido a la diversidad que existe de tipo de engranes, los ejes de entrada no tienen que estar alineados ni paralelos, así mismo se pueden obtener diferentes razones de velocidad angular variables si los engranes utilizados son no circulares [2].

El método de encontrar las ecuaciones de restricción cinemática de un mecanismo y sus diferenciales totales también se conoce con el nombre de método de los desplazamientos virtuales de los eslabones de un mecanismo [3].

En este trabajo, se presenta un modelo matemático para encontrar cómo los errores asociados a las tolerancias dimensionales del radio base y del ángulo de presión de la transmisión, influyen en el error en la relación de transmisión. Para esto, se modela la transmisión por engranajes como un mecanismo de cuatro barras equivalente cuyas cuatro barras son: el bastidor definido como la distancia entre centros fijos, el radio base del piñón como elemento de entrada, el radio base del engrane conducido y la línea de acción normal como la biela.

A través de MATLAB se grafican los errores en la relación de transmisión en función de las tolerancias dimensionales de los parámetros geométricos del mecanismo de cuatro barras equivalente. Se presentan ejemplos del cálculo de las tolerancias dimensionales de una pareja de engranes rectos, con un radio base del engrane de 37 mm y un radio base del piñón de 19 mm, para varios grados de calidad IT. Se determina su relación de transmisión teórica y se compara con la obtenida al tomar en cuenta las tolerancias dimensionales de sus parámetros geométricos, para un juego de calidades IT de ISO correspondientes a IT01, IT6, IT9, IT12 e IT16, de acuerdo a la tabla de calidades IT normalizadas.

1.1 Radio de Curvatura de una curva involuta

Para comprender los fundamentos matemáticos relacionados con este trabajo partiremos de la siguiente definición: “la curva involuta de un círculo es el lugar geométrico de la punta de una cuerda que se desenrolla y se mantiene tensa a partir de un círculo conocido como círculo base” [1].

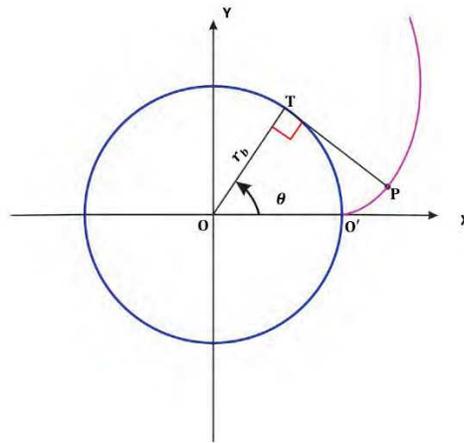


Figura 1. Círculo base y su curva involuta.

En la figura 1 se presenta la curva involuta generada a partir de un círculo de radio base r_b . La longitud del arco $\widehat{O'T}$ está dada a partir de la ecuación (1).

$$\widehat{O'T} = \theta r_b \quad (1)$$

Por definición, el ángulo θ está definido como se señala en (2).

$$\theta = \frac{\widehat{O'T}}{r_b} \quad (2)$$

De acuerdo a la figura 1, la longitud de arco también queda definida según (3).

$$\widehat{O'T} = \overline{TP} \quad (3)$$

A su vez, la longitud \overline{TP} se puede expresar en función del ángulo θ y del radio base r_b como se indica en la ecuación (4).

$$\overline{TP} = \theta r_b \quad (4)$$

Por otra parte, el ángulo θ entre la línea \overline{OT} y la línea de referencia horizontal $\overline{OO'}$ es el mismo ángulo que se forma entre la línea \overline{TP} y la vertical que pasa por T , como se muestra en la figura 1. La proyección del segmento \overline{TP} sobre los ejes x e y es la indicada en (5).



$$\overline{TP} = \theta r_b (\text{sen}(\theta)\hat{i} - \text{cos}(\theta)\hat{j}) \quad (5)$$

Por otro lado, las ecuaciones paramétricas del punto P están dadas en (6).

$$\begin{aligned} x &= r_b (\text{cos}(\theta) + \theta \text{sen}(\theta)) \\ y &= r_b (\text{sen}(\theta) - \theta \text{cos}(\theta)) \end{aligned} \quad (6)$$

Para una curva definida en forma paramétrica por las ecuaciones $x = x(\theta)$, $y = y(\theta)$, entonces su curvatura K en cualquier punto (x, y) viene dada por la ecuación (7). El numerador es un polinomio con base en una función suave y también lo es el denominador, el resultado estará dado también por una función suave [2].

$$K = \frac{|x'y'' - y'x''|}{|(x')^2 + (y')^2|^{\frac{3}{2}}} \quad (7)$$

Por lo tanto, de acuerdo a la ecuación (7), necesitamos obtener las primeras y segundas derivadas de las ecuaciones paramétricas. Las primeras derivadas se describen en las ecuaciones dadas en (8).

$$\begin{aligned} x' &= r_b (\theta \text{cos}(\theta)) \\ y' &= r_b (\theta \text{sen}(\theta)) \end{aligned} \quad (8)$$

Las segundas derivadas de las ecuaciones paramétricas indicadas en (6) están dadas en (9).

$$\begin{aligned} x'' &= r_b (\text{cos}(\theta) - \theta \text{sen}(\theta)) \\ y'' &= r_b (\text{sen}(\theta) + \theta \text{cos}(\theta)) \end{aligned} \quad (9)$$

Al sustituir las ecuaciones (8) y (9) en la ecuación (7), se obtendrá la expresión para la curvatura en cualquier punto de la curva involuta, como se muestra en la ecuación (10).

$$K = \frac{r_b(\theta \text{cos}(\theta))(r_b(\text{sen}(\theta) + \theta \text{cos}(\theta))) - r_b(\theta \text{sen}(\theta))(r_b(\text{cos}(\theta) - \theta \text{sen}(\theta)))}{[(r_b \theta \text{cos}(\theta))^2 + (r_b \theta \text{sen}(\theta))^2]^{\frac{3}{2}}} \quad (10)$$

Finalmente, desarrollando el álgebra obtenemos la curvatura descrita por la ecuación (11).

$$K = \frac{1}{\theta r_b} \quad (11)$$

Como el radio de curvatura en un punto arbitrario está dado como el inverso de la curvatura K [2], la expresión para el radio de curvatura en un punto arbitrario de la involuta se describe en la ecuación (12). El radio de curvatura está denotado por la letra ρ .

$$\rho = \frac{1}{K} \quad (12)$$

Al sustituir la ecuación (12) en la ecuación (11) obtenemos el radio de curvatura en un punto arbitrario de la involuta, como se indica en (13).

$$\rho = \theta r_b = \overline{TP} \quad (13)$$

De este resultado se enuncia el siguiente corolario:

El centro de curvatura de una curva de tipo involuta en un punto arbitrario siempre yacerá sobre el círculo base de la misma involuta, así mismo la evoluta de la involuta de un círculo base es el propio círculo base, estando el radio de curvatura en función del radio base y su correspondiente ángulo [1].

2. Desarrollo

2.1 Modelación de un mecanismo equivalente de cuatro barras para un engranaje cilíndrico de dientes rectos

En la figura 2, se muestra dos involutas en contacto o “engranadas” y se representan los dientes del piñón y del engrane. De la figura 2 es posible observar el mecanismo de 4 barras equivalente formado por el radio base del engrane, el radio base del piñón, la línea de acción y la distancia entre centros. Este mecanismo equivalente consiste de una etapa de engranado. De acuerdo a la geometría es posible definir la relación de transmisión como se indica en la ecuación (14).

$$i = \frac{O_4P}{O_2P} \quad (14)$$

En la ecuación (14), i representa la relación de transmisión. El segmento $\overline{O_4P}$ es la línea horizontal desde el centro del engrane al punto de paso y $\overline{O_2P}$ es la línea horizontal desde el centro del piñón al punto de paso, siendo P dicho punto de paso donde los círculos de paso del engrane y el piñón son tangentes entre sí.

Podemos extraer de la figura 2 el mecanismo equivalente de cuatro barras como se indica en líneas rojas. En la figura 3 se muestra el mecanismo equivalente de cuatro barras, el cual está conformado por dos lazos vectoriales triangulares.

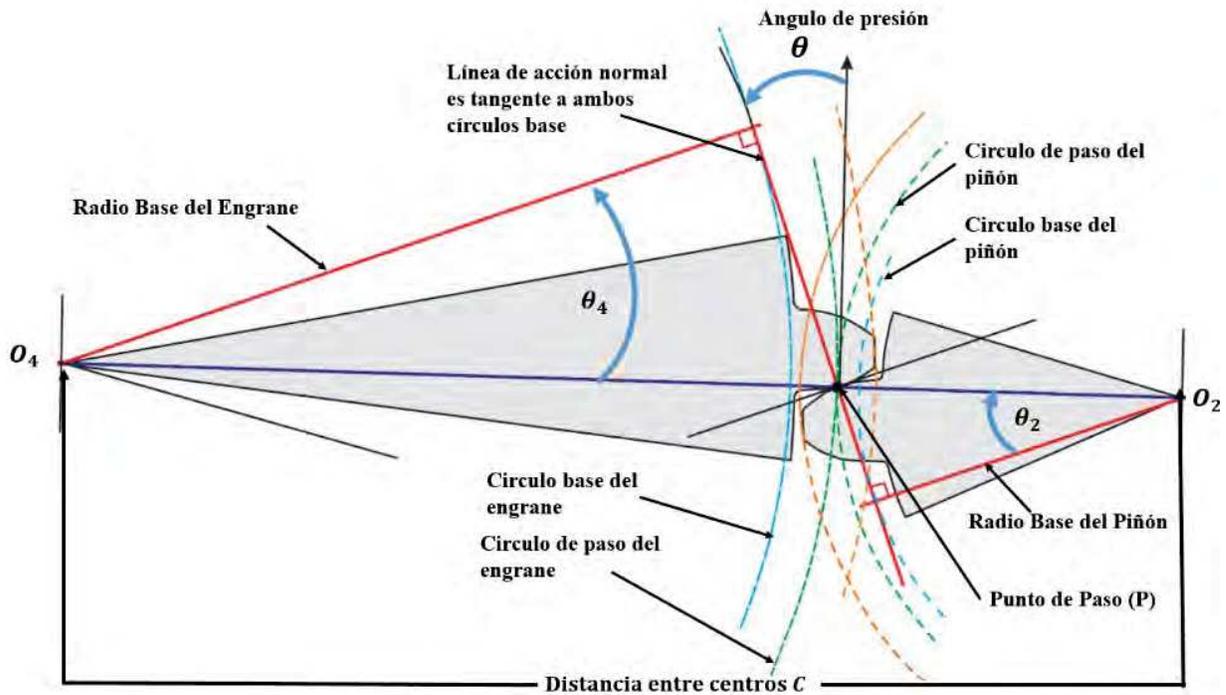


Figura 2. Geometría de contacto entre un piñón y un engrane.

De acuerdo al modelo equivalente de la figura 3, el mecanismo de cuatro barras está conformado por: la longitud l_1 correspondiente al radio base del piñón, las longitudes l_2 y l_3 asociadas a la longitud de la línea de acción, la longitud l_4 es el radio base del engrane, la longitud l_5 corresponde a la línea horizontal $\overline{O_4P}$ y la longitud l_6 es la línea horizontal $\overline{O_2P}$. A su vez, el ángulo θ_2 representa el ángulo de presión del piñón y el ángulo θ_4 el ángulo de presión del engrane. Teóricamente los ángulos θ_2 y θ_4 son iguales [3].

A partir del modelo equivalente se modelan las ecuaciones de restricción cinemática sobre los ejes x , y , respectivamente. Las ecuaciones del primer lazo vectorial están dadas como se muestra en (15).

Las ecuaciones para el segundo lazo vectorial están dadas en (16).

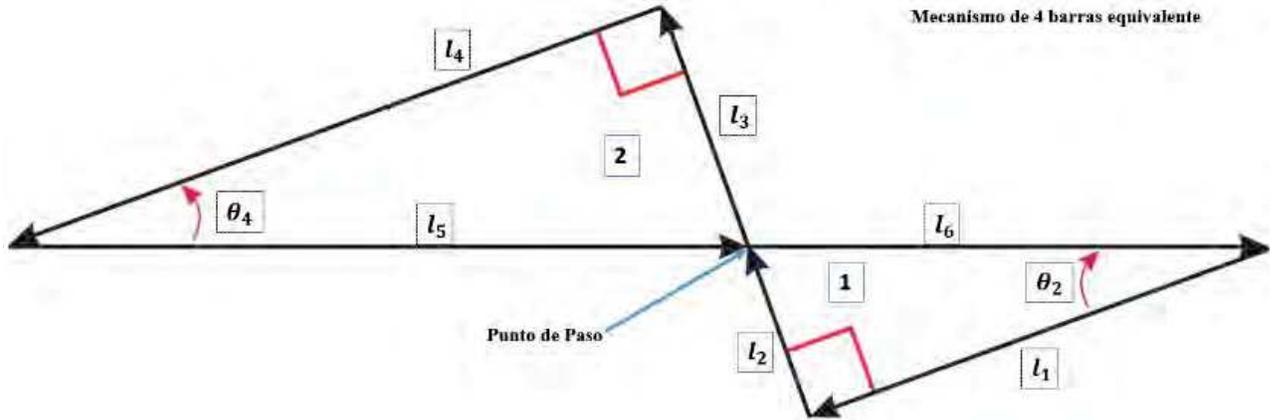


Figura 3. Geometría de contacto entre un piñón y un engrane.

$$\begin{aligned} l_1 \cos(\theta_2) + l_2 \cos\left(\frac{\pi}{2} - \theta_2\right) - l_6 &= 0 \\ -l_1 \sin(\theta_2) + l_2 \sin\left(\frac{\pi}{2} - \theta_2\right) &= 0 \end{aligned} \quad (15)$$

$$\begin{aligned} l_3 \cos\left(\frac{\pi}{2} - \theta_4\right) + l_4 \cos(\theta_4) - l_5 &= 0 \\ l_3 \sin\left(\frac{\pi}{2} - \theta_4\right) - l_4 \sin(\theta_4) &= 0 \end{aligned} \quad (16)$$

De acuerdo a las funciones trigonométricas fundamentales [4] indicadas en (17), es posible reestructurar las ecuaciones de restricción cinemática, reescribirlas y denotarlas como se indica en (18).

$$\begin{aligned} \cos\left(\frac{\pi}{2} - \theta\right) &= \sin(\theta) \\ \sin\left(\frac{\pi}{2} - \theta\right) &= \cos(\theta) \end{aligned} \quad (17)$$

En las ecuaciones dadas en (18) tenemos las ecuaciones de manera más compacta que nos permitirá un manejo mucho más adecuado de las operaciones.

$$\begin{aligned} l_1 \cos(\theta_2) + l_2 \sin(\theta_2) - l_6 &= 0 \\ -l_1 \sin(\theta_2) + l_2 \cos(\theta_2) &= 0 \\ l_3 \sin(\theta_4) + l_4 \cos(\theta_4) - l_5 &= 0 \\ l_3 \cos(\theta_4) - l_4 \sin(\theta_4) &= 0 \end{aligned} \quad (18)$$

De las ecuaciones dadas en (18) podemos despejar los parámetros geométricos l_2 , l_3 , l_5 y l_6 , en función de los radios bases del piñón y el engrane, respectivamente, como se indica en la ecuación (19).

$$\begin{aligned} l_2 &= l_1 \tan(\theta_2) \\ l_3 &= l_4 \tan(\theta_4) \\ l_5 &= l_4 (\cos(\theta_4) + \tan(\theta_4) \sin(\theta_4)) \\ l_6 &= l_4 (\cos(\theta_2) + \tan(\theta_2) \sin(\theta_2)) \end{aligned} \quad (19)$$



Los dos lazos vectoriales están en función de las variables del mecanismo de cuatro barras equivalente $l_1, l_2, l_3, l_4, l_5, l_6, \theta_2, \theta_4$. A las cuatro ecuaciones de restricción cinemática del mecanismo equivalente dadas en (18,) se le aplicará la diferencial total [5] [6] con respecto a las variables implicadas. Obtenemos las ecuaciones señaladas en (20)

$$\begin{aligned} \cos(\theta_2)\delta l_1 + \sin(\theta_2)\delta l_2 - \delta l_6 - l_1 \sin(\theta_2)\delta\theta_2 + l_2 \cos(\theta_2)\delta\theta_2 &= 0 \\ -\sin(\theta_2)\delta l_1 + \cos(\theta_2)\delta l_2 - l_1 \cos(\theta_2)\delta\theta_2 - l_2 \sin(\theta_2)\delta\theta_2 &= 0 \\ \cos(\theta_4)\delta l_4 + \sin(\theta_4)\delta l_3 - \delta l_5 + l_3 \cos(\theta_4)\delta\theta_4 - l_4 \sin(\theta_4)\delta\theta_4 &= 0 \\ -\sin(\theta_4)\delta l_4 + \cos(\theta_4)\delta l_3 - l_4 \cos(\theta_4)\delta\theta_4 - l_3 \sin(\theta_4)\delta\theta_4 &= 0 \end{aligned} \quad (20)$$

Las diferenciales de las ecuaciones de restricción cinemática expresadas en forma matricial se denotan como se indica en (21).

$$\begin{bmatrix} \sin(\theta_2) & 0 & 0 & -1 \\ \cos(\theta_2) & 0 & 0 & 0 \\ 0 & \sin(\theta_4) & -1 & 0 \\ 0 & \cos(\theta_4) & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta l_2 \\ \delta l_3 \\ \delta l_5 \\ \delta l_6 \end{bmatrix} + \begin{bmatrix} \cos(\theta_2) & 0 & -l_1 \sin(\theta_2) + l_2 \cos(\theta_2) & 0 \\ -\sin(\theta_2) & 0 & -l_1 \cos(\theta_2) - l_2 \sin(\theta_2) & 0 \\ 0 & \cos(\theta_4) & 0 & l_3 \cos(\theta_4) - l_4 \sin(\theta_4) \\ 0 & -\sin(\theta_4) & 0 & -l_3 \sin(\theta_4) - l_4 \cos(\theta_4) \end{bmatrix} \begin{bmatrix} \delta l_1 \\ \delta l_4 \\ \delta l_{\theta_2} \\ \delta l_{\theta_4} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (21)$$

Despejando las variables de interés, que son δl_5 y δl_6 , se puede determinar la influencia de las tolerancias en la relación de transmisión de un engranaje cilíndrico de dientes rectos, ya que la relación de transmisión viene dada por el cociente $\frac{l_5}{l_6}$ [7].

Las matrices **A** y **B** mostradas en (22) quedan expresadas en función de los ángulos de presión del engrane y el piñón, así como de las longitudes l_1, l_2, l_3, l_4 que componen el mecanismo de cuatro barras equivalente.

$$\mathbf{A} = \begin{bmatrix} \sin(\theta_2) & 0 & 0 & -1 \\ \cos(\theta_2) & 0 & 0 & 0 \\ 0 & \sin(\theta_4) & -1 & 0 \\ 0 & \cos(\theta_4) & 0 & 0 \end{bmatrix} \quad (22)$$

$$\mathbf{B} = \begin{bmatrix} \cos(\theta_2) & 0 & -l_1 \sin(\theta_2) + l_2 \cos(\theta_2) & 0 \\ -\sin(\theta_2) & 0 & -l_1 \cos(\theta_2) - l_2 \sin(\theta_2) & 0 \\ 0 & \cos(\theta_4) & 0 & l_3 \cos(\theta_4) - l_4 \sin(\theta_4) \\ 0 & -\sin(\theta_4) & 0 & -l_3 \sin(\theta_4) - l_4 \cos(\theta_4) \end{bmatrix}$$

Podemos notar que ambas matrices son cuadradas de orden 4 x 4. La expresión (21) se puede reescribir en una forma compacta como se indica en (23).

$$\mathbf{A} \begin{bmatrix} \delta l_2 \\ \delta l_3 \\ \delta l_5 \\ \delta l_6 \end{bmatrix} + \mathbf{B} \begin{bmatrix} \delta l_1 \\ \delta l_4 \\ \delta l_{\theta_2} \\ \delta l_{\theta_4} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (23)$$

Las incógnitas de interés son δl_5 y δl_6 , las cuales serán despejadas de la expresión matricial compacta dada en (23).

$$\begin{bmatrix} \delta l_2 \\ \delta l_3 \\ \delta l_5 \\ \delta l_6 \end{bmatrix} = -\mathbf{A}^{-1} \mathbf{B} \begin{bmatrix} \delta l_1 \\ \delta l_4 \\ \delta l_{\theta_2} \\ \delta l_{\theta_4} \end{bmatrix} \quad (24)$$

La expresión completa se muestra en (25), donde tenemos que la matriz **A** es invertible y por lo tanto es posible realizar el producto matricial indicado en (24). El resultado de la operación será una matriz de 4 x 4.



$$\begin{bmatrix} \delta l_2 \\ \delta l_3 \\ \delta l_5 \\ \delta l_6 \end{bmatrix} = - \begin{bmatrix} \sin(\theta_2) & 0 & 0 & -1 \\ \cos(\theta_2) & 0 & 0 & 0 \\ 0 & \sin(\theta_4) & -1 & 0 \\ 0 & \cos(\theta_4) & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} \cos(\theta_2) & 0 & -l_1 \sin(\theta_2) + l_2 \cos(\theta_2) & 0 \\ -\sin(\theta_2) & 0 & -l_1 \cos(\theta_2) - l_2 \sin(\theta_2) & 0 \\ 0 & \cos(\theta_4) & 0 & l_3 \cos(\theta_4) - l_4 \sin(\theta_4) \\ 0 & -\sin(\theta_4) & 0 & -l_3 \sin(\theta_4) - l_4 \cos(\theta_4) \end{bmatrix} \begin{bmatrix} \delta l_1 \\ \delta l_4 \\ \delta \theta_2 \\ \delta \theta_4 \end{bmatrix} \quad (25)$$

En la expresión (26) se han sustituido las variables correspondientes a l_2 y l_3 en función de las variables l_1 y l_4 correspondientes a los radios del piñón y el engrane de acuerdo a las ecuaciones dadas en (19).

$$\begin{bmatrix} \delta l_2 \\ \delta l_3 \\ \delta l_5 \\ \delta l_6 \end{bmatrix} = - \begin{bmatrix} \sin(\theta_2) & 0 & 0 & -1 \\ \cos(\theta_2) & 0 & 0 & 0 \\ 0 & \sin(\theta_4) & -1 & 0 \\ 0 & \cos(\theta_4) & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} \cos(\theta_2) & 0 & 0 & 0 \\ -\sin(\theta_2) & 0 & -l_1 \left(\cos(\theta_2) + \frac{\sin^2(\theta_2)}{\cos(\theta_2)} \right) & 0 \\ 0 & \cos(\theta_4) & 0 & 0 \\ 0 & -\sin(\theta_4) & 0 & -l_4 \left(\cos(\theta_4) + \frac{\sin^2(\theta_4)}{\cos(\theta_4)} \right) \end{bmatrix} \begin{bmatrix} \delta l_1 \\ \delta l_4 \\ \delta \theta_2 \\ \delta \theta_4 \end{bmatrix} \quad (26)$$

El resultado de la multiplicación de la matriz inversa A^{-1} con la matriz B afectado por el signo negativo se escribe en (27).

$$\begin{bmatrix} \delta l_2 \\ \delta l_3 \\ \delta l_5 \\ \delta l_6 \end{bmatrix} = \begin{bmatrix} \tan(\theta_2) & 0 & \frac{l_1 \cos(\theta_2) + \tan(\theta_2) \sin(\theta_2)}{\cos(\theta_2)} & 0 \\ 0 & \tan(\theta_4) & 0 & \frac{l_4 \cos(\theta_4) + \tan(\theta_4) \sin(\theta_4)}{\cos(\theta_4)} \\ 0 & \cos(\theta_4) + \frac{\sin^2(\theta_4)}{\cos(\theta_4)} & 0 & \frac{l_4 \sin(\theta_4) (\cos(\theta_4) + \tan(\theta_4) \sin(\theta_4))}{\cos(\theta_4)} \\ \cos(\theta_2) + \frac{\sin^2(\theta_2)}{\cos(\theta_2)} & 0 & \frac{l_1 \sin(\theta_2) (\cos(\theta_2) + \tan(\theta_2) \sin(\theta_2))}{\cos(\theta_2)} & 0 \end{bmatrix} \begin{bmatrix} \delta l_1 \\ \delta l_4 \\ \delta \theta_2 \\ \delta \theta_4 \end{bmatrix} \quad (27)$$

De la ecuación (27) se deduce los valores δl_2 , δl_3 , δl_5 y δl_6 . De estos, tomaremos los errores δl_5 y δl_6 , necesarios para calcular los errores buscados en la relación de transmisión de los engranajes cilíndricos de dientes rectos. La ecuación (28) muestra el despeje.

$$\begin{aligned} \delta l_2 &= \tan(\theta_2) \delta l_1 + \frac{l_1 (\cos(\theta_2) + \tan(\theta_2) \sin(\theta_2))}{\cos(\theta_2)} \delta \theta_2 \\ \delta l_3 &= \tan(\theta_4) \delta l_4 + \frac{l_4 (\cos(\theta_4) + \tan(\theta_4) \sin(\theta_4))}{\cos(\theta_4)} \delta \theta_4 \\ \delta l_5 &= \left(\cos(\theta_4) + \frac{\sin^2(\theta_4)}{\cos(\theta_4)} \right) \delta l_4 + \frac{l_4 \sin(\theta_4) (\cos(\theta_4) + \tan(\theta_4) \sin(\theta_4))}{\cos(\theta_4)} \delta \theta_4 \\ \delta l_6 &= \left(\cos(\theta_2) + \frac{\sin^2(\theta_2)}{\cos(\theta_2)} \right) \delta l_1 + \frac{l_1 \sin(\theta_2) (\cos(\theta_2) + \tan(\theta_2) \sin(\theta_2))}{\cos(\theta_2)} \delta \theta_2 \end{aligned} \quad (28)$$

Sabemos también que la relación de transmisión [7] está dada por el cociente entre las variables l_5 y l_6 como se describe en (29).

$$i = \frac{l_5}{l_6} \quad (29)$$

La relación de transmisión de un engranaje es idealmente constante [3] y adimensional. Al sustituir las expresiones que involucran a l_5 y l_6 dadas en (19), en la ecuación (29), tenemos que la relación de transmisión en un engranaje cilíndrico recto está dada por la expresión (30).

$$i = \frac{l_4 (\cos(\theta_4) + \tan(\theta_4) \sin(\theta_4))}{l_1 (\cos(\theta_2) + \tan(\theta_2) \sin(\theta_2))} \quad (30)$$



3. Resultados

Si ahora tomamos la diferencial total de la ecuación (29), obtendremos el error buscado en la relación de transmisión en función de los errores en el radio base y en el ángulo de presión.

La expresión para el diferencial total de la relación de transmisión se indica en (31).

$$\delta i = \frac{\delta l_5 l_6 - l_5 \delta l_6}{l_6^2} \quad (31)$$

En la ecuación (31) δi también será adimensional, por lo tanto, la relación de transmisión real, no teórica, está dada por la combinación de las variables correspondientes a l_5 , l_6 , δl_5 , δl_6 y se expresa en la ecuación (32).

$$i = \frac{l_4(\cos(\theta_4) + \tan(\theta_4)\text{sen}(\theta_4))}{l_1(\cos(\theta_2) + \tan(\theta_2)\text{sen}(\theta_2))} + \frac{\delta l_5 l_6 - l_5 \delta l_6}{l_6^2} \quad (32)$$

En esta ecuación (32) los valores de δl_5 y δl_6 se obtienen de la ecuación (28). Las magnitudes l_5 y l_6 son las distancias en las que la línea de acción divide a la línea de centros, es decir, la distancia desde el centro del engrane al punto de paso y la distancia desde el centro del piñón al mismo punto.

3.1 Primer Caso de Estudio Particular

Se presentan dos casos de estudio particular, el primero de ellos corresponde a un par de engranajes que tienen diferentes longitudes en sus radios base. En la tabla 1 se muestran los datos de entrada para este problema, el radio base del engrane es de 37 mm y el radio base del piñón de 19 mm. En este ejemplo se presenta ruedas dentadas de diámetros bastante pequeños. Este tipo de ruedas pueden ser encontradas por ejemplo, en mecanismos de reloj o en servomotores pequeños.

Tabla 1. Radios base del piñón y el engrane.

Parámetro	Longitud [mm]
l_1 (radio base piñón)	19
l_4 radio base engrane)	37

Los valores de δl_1 (Tolerancia dimensional del radio base del piñón), δl_4 (Tolerancia dimensional del radio base del engrane), $\delta \theta_2$ (Tolerancia dimensional del error angular del ángulo de presión para el piñón) y $\delta \theta_4$ (Tolerancia dimensional del error angular del ángulo de presión para el engrane) serán obtenidos a través de una tabla IT de medidas normalizadas, como se muestra en la tabla 2.

Los valores δl_1 y δl_4 son extraídos de la Norma ISO 286 según la tabla 2. En esta tabla vienen los diferentes tipos de calidades IT [8] de acuerdo al tamaño nominal de las dimensiones l_1 y l_4 , del radio base del piñón y del radio base del engrane. Para las dimensiones que ocuparemos en este trabajo se utilizarán las calidades IT01, IT6, IT9, IT12 e IT16. La calidad IT 01 es una calidad de ultra precisión, la calidad IT 6 es una calidad de precisión, la calidad IT 9 es una calidad estándar que se puede dar en los talleres con facilidad, y las calidades IT 12 e IT 16 son bajas. Las calidades IT están dadas en micrones [9]. Para este caso, asumiremos $\delta \theta_2$ y $\delta \theta_4$ iguales a cero, para solo medir la influencia de los errores en los radios base en la relación de transmisión.



Tabla 2. Tabla de calidades IT normalizada (Tolerancias dadas en micrones) [8].

Díámetros [mm]	IT 01	IT 0	IT 1	IT 2	IT 3	IT 4	IT 5	IT 6	IT 7	IT 8	IT 9	IT 10	IT 11	IT 12	IT 13	IT 14	IT 15	IT 16
d≤3	0.3	0.5	0.8	1.2	2	3	4	6	10	14	25	40	60	100	140	250	400	600
3<d≤6	0.4	0.6	1	1.5	2.5	4	5	8	12	18	30	48	75	120	180	300	480	750
6<d≤10	0.4	0.6	1	1.5	2.5	4	6	9	15	22	36	58	90	150	220	360	580	900
10<d≤18	0.5	0.8	1.2	2	3	5	8	11	18	27	43	70	110	180	270	430	700	1100
18<d≤30	0.6	1	1.5	2.5	4	6	9	13	21	33	52	84	130	210	330	520	840	1300
30<d≤50	0.6	1	1.5	2.5	4	7	11	16	25	39	62	100	160	250	390	620	1000	1600
50<d≤80	0.8	1.2	2	3	5	8	13	19	30	46	74	120	190	300	460	740	1200	1900
80<d≤120	1	1.5	2.5	4	6	10	15	22	35	54	87	140	220	350	540	870	1400	2200
120<d≤180	1.2	2	3.5	5	8	12	18	25	40	63	100	160	250	400	630	1000	1600	2500
180<d≤250	2	3	4.5	7	10	14	20	29	46	72	115	185	290	460	720	1150	1850	2900
250<d≤315	2.5	4	6	8	12	16	23	32	52	81	130	210	320	520	810	1300	2100	3200
315<d≤400	3	5	7	9	13	18	25	36	57	89	140	230	360	570	890	1400	2300	3600
400<d≤500	4	6	8	10	15	20	27	40	63	97	155	250	400	630	970	1550	2500	4000

En la tabla 3 se muestran los errores δl_1 , δl_4 admitidos para las calidades IT01, IT6, IT9, IT12 e IT16. Se puede notar que mientras mayor es el grado IT mayor es el error admisible permitido y menos exigente la calidad.

Tabla 3. Tolerancias para los radios bases del piñón y el engrane (tomado de la Norma ISO 286).

Parámetro	IT 01 [mm]	IT 6 [mm]	IT 9 [mm]	IT 12 [mm]	IT 16 [mm]
δl_1	0.0006	0.016	0.062	0.25	1.6
δl_4	0.0008	0.019	0.074	0.21	1.9

La tabla 4 indica las diferentes relaciones de transmisión que se obtienen cuando los radios bases son de 19 y 37 mm, para ángulos de presión comerciales entre 20 y 25 grados y los distintos grados IT de tolerancias para los radios base. La primera columna es la relación de transmisión teórica de acuerdo a la ecuación (14). La tabla 4 es obtenida a través de MATLAB.

De la simulación en MATLAB se obtiene que la relación de transmisión, según el grado IT de calidad, es esencialmente constante para ángulos de presión entre 20 y 25 grados, siendo estos ángulos los más empleados en la industria de la fabricación de engranajes. Sin embargo, las relaciones de transmisión obtenidas al tomar en cuenta las tolerancias dimensionales en los radios bases, difieren de la relación de transmisión teórica [10], en mayor o menor medida obteniéndose un error hasta de -3.2 % cuando la calidad de las tolerancias en los radios bases es baja (IT16).

La tabla 5 muestra el porcentaje de error máximo existente entre las relaciones de transmisión teórica y la real para este caso de estudio.



Tabla 4. Diferentes relaciones de transmisión para diferentes valores de calidad IT

Grados	Relación de transmisión teórica	Relación de transmisión con IT 01	Relación de transmisión con IT 6	Relación de transmisión con IT 9	Relación de transmisión con IT 12	Relación de transmisión con IT 16
20.0	1.947	1.947	1.947	1.943	1.932	1.881
20.5	1.947	1.947	1.947	1.943	1.932	1.881
21.0	1.947	1.947	1.947	1.943	1.932	1.882
21.5	1.947	1.947	1.947	1.943	1.932	1.882
22.0	1.947	1.947	1.947	1.943	1.932	1.882
22.5	1.947	1.947	1.947	1.943	1.932	1.883
23.0	1.947	1.947	1.947	1.943	1.933	1.883
23.5	1.947	1.947	1.947	1.943	1.933	1.884
24.0	1.947	1.947	1.947	1.943	1.933	1.884
24.5	1.947	1.947	1.947	1.943	1.933	1.885
25.0	1.947	1.947	1.947	1.943	1.933	1.886

Tabla 5. Porcentajes de error máximos entre la relación de transmisión teórica y la real cuando los radios base del piñón y el engrane son de 19 mm y 37 mm, respectivamente, y el ángulo de presión varía entre 20 y 25 grados.

Grados	Porcentaje de Error para IT 01 [%]	Porcentaje de Error para IT 6 [%]	Porcentaje de Error para IT 9 [%]	Porcentaje de Error para IT 12 [%]	Porcentaje de Error para IT 16 [%]
20.0	-0.001	-0.033	-0.130	-0.772	-3.392
20.5	-0.001	-0.033	-0.129	-0.767	-3.371
21.0	-0.001	-0.033	-0.128	-0.763	-3.351
21.5	-0.001	-0.033	-0.128	-0.758	-3.330
22.0	-0.001	-0.033	-0.127	-0.753	-3.308
22.5	-0.001	-0.032	-0.126	-0.748	-3.287
23.0	-0.001	-0.032	-0.125	-0.743	-3.264
23.5	-0.001	-0.032	-0.124	-0.738	-3.242
24.0	-0.001	-0.032	-0.123	-0.733	-3.219
24.5	-0.001	-0.031	-0.122	-0.727	3.195
25.0	-0.001	-0.031	-0.121	-0.722	-3.172

En la figura 4 se muestra el porcentaje de error entre la relación de transmisión real y la teórica para ángulos de presión entre 20 y 25 grados y tolerancias dimensionales IT de ultra precisión a muy baja precisión.

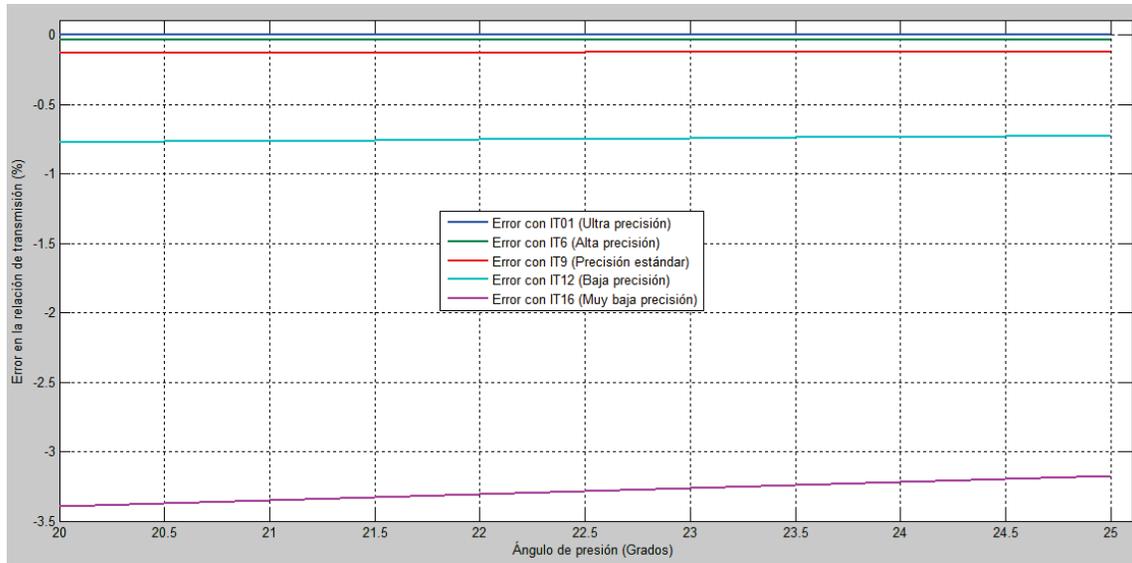


Figura 4. Errores en la relación de transmisión para el caso 1 para diferentes valores de ángulos de presión y tolerancias dimensionales IT de los radios base del piñón (19 mm) y del engrane (37 mm).

1.1 Segundo Caso de Estudio Particular

Para un siguiente ejemplo usaremos una pareja de engranes rectos con un radio base del engrane de 370 mm y un radio base del piñón de 190 mm. La tabla 6 muestra los parámetros de entrada para este caso.

Tabla 6. Parámetros de entrada de los radios base .

Parámetro	Longitud [mm]
l_1 (radio base piñón)	190
l_4 (radio base engrane)	370

En la tabla 7 se muestran los errores δl_1 , δl_4 , asociados a los parámetros anteriores con las calidades IT01, IT6, IT9, IT12 e IT16 respectivamente.

Tabla 7. Tolerancias para los radios bases del piñón y el engrane (tomado de la Norma ISO 286).

Parámetro	IT 01 [mm]	IT 6 [mm]	IT 9 [mm]	IT 12 [mm]	IT 16 [mm]
δl_1	0.003	0.036	0.140	0.57	3.6
δl_4	0.002	0.029	0.115	0.46	2.9

La tabla 8 muestra las diferentes relaciones de transmisión para diferentes ángulos de presión comerciales de engranes. Se incluye la relación de transmisión teórica de acuerdo a la ecuación (14), y diferentes valores de relación de transmisión en función de los grados IT de la tabla 2. La tabla 8 es obtenida a través de MATLAB.

En la figura 5, podemos visualizar de manera gráfica como las relaciones de transmisión son prácticamente constantes, al aumentar en el caso 2 los radios base tanto del piñón como del engrane.



Se observa que la relación de transmisión no cambia con el ángulo de presión y los errores son ahora mucho menores que en el caso 1.

Tabla 8. Diferentes relaciones de transmisión para diferentes valores de calidad IT

Grados	Relación de transmisión teórica	Relación de transmisión con IT 01	Relación de transmisión con IT 6	Relación de transmisión con IT 9	Relación de transmisión con IT 12	Relación de transmisión con IT 16
20.0	1.947	1.947	1.947	1.946	1.943	1.925
20.5	1.947	1.947	1.947	1.946	1.943	1.925
21.0	1.947	1.947	1.947	1.946	1.943	1.925
21.5	1.947	1.947	1.947	1.946	1.943	1.925
22.0	1.947	1.947	1.947	1.946	1.943	1.925
22.5	1.947	1.947	1.947	1.946	1.943	1.925
23.0	1.947	1.947	1.947	1.946	1.943	1.925
23.5	1.947	1.947	1.947	1.946	1.943	1.926
24.0	1.947	1.947	1.947	1.946	1.944	1.926
24.5	1.947	1.947	1.947	1.946	1.944	1.926
25.0	1.947	1.947	1.947	1.946	1.944	1.926

La tabla 9 muestra el porcentaje de error máximo existente en las relaciones de transmisión, al tomar en cuenta las tolerancias dimensionales de los radios bases y los ángulos de presión del piñón y del engrane, tanto como la relación de transmisión teórica.

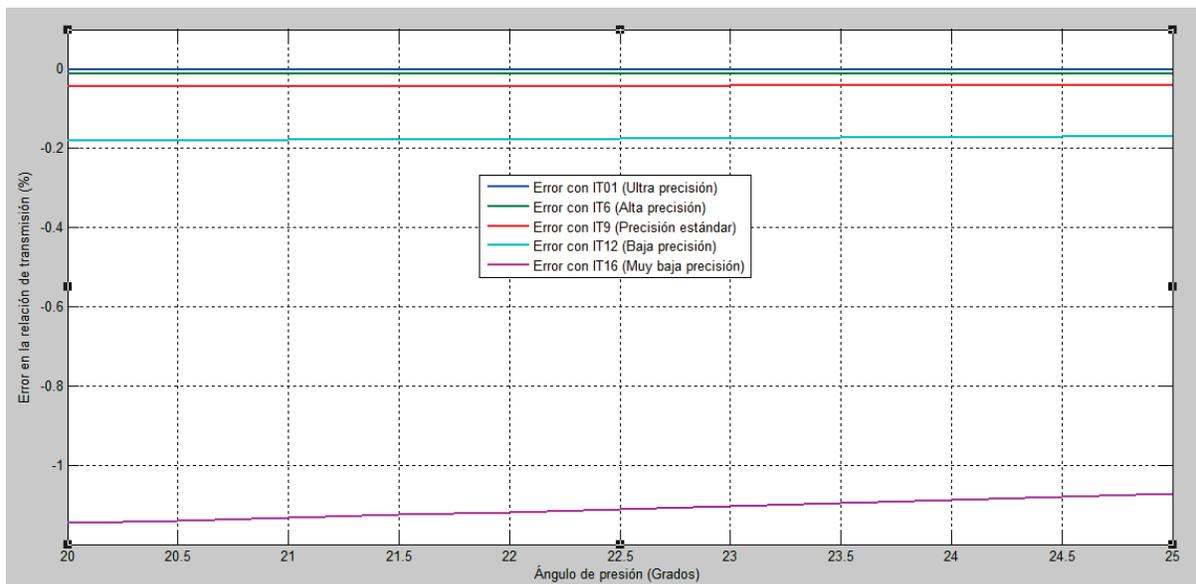


Figura 5. Errores en la relación de transmisión para el caso 2 para diferentes valores de ángulos de presión y tolerancias dimensionales IT de los radios base del piñón (190 mm) y del engrane (370 mm).



Tabla 9. Porcentajes de errores máximos entre la relación de transmisión teórica y la real cuando los radios base del piñón y el engrane son de 190 mm y 370 mm respectivamente, y el ángulo de presión varía entre 20 y 25 grados.

Grados	Porcentaje de Error para IT 01 [%]	Porcentaje de Error para IT 6 [%]	Porcentaje de Error para IT 9 [%]	Porcentaje de Error para IT 12 [%]	Porcentaje de Error para IT 16 [%]
20.0	-0.001	-0.011	-0.043	-0.181	-1.146
20.5	-0.001	-0.011	-0.043	-0.180	-1.140
21.0	-0.001	-0.011	-0.043	-0.179	-1.133
21.5	-0.001	-0.011	-0.043	-0.178	-1.125
22.0	-0.001	-0.011	-0.042	-0.176	-1.118
22.5	-0.001	-0.011	-0.042	-0.175	-1.111
23.0	-0.001	-0.011	-0.042	-0.174	-1.103
23.5	-0.001	-0.010	-0.042	-0.173	-1.096
24.0	-0.001	-0.010	-0.041	-0.172	-1.088
24.5	-0.001	-0.010	-0.041	-0.170	-1.080
25.0	-0.001	-0.010	-0.041	-0.169	-1.072

2. Conclusiones

De acuerdo a los datos obtenidos, sabemos que entre más pequeñas son las ruedas dentadas, mayor es el porcentaje de error encontrado en la relación de transmisión. El porcentaje de error entre la relación de transmisión teórica y la que resulta al tomar en cuenta las tolerancias dimensionales en el radio base y en los ángulos de presión, aumenta conforme disminuye el grado de precisión IT normalizado de las dimensiones lineales y angulares de estos dos parámetros geométricos. El método de definir las ecuaciones geométricas de restricción cinemática y tomar las diferenciales totales de estas ecuaciones, usado en el mecanismo equivalente de cuatro barras estacionario, se puede extender para abarcar mecanismos de cuatro barras móviles. El conocimiento de cómo las tolerancias dimensionales y las posiciones fundamentales de estas tolerancias influye en la relación de transmisión, permite asignar grados IT normalizados y posiciones fundamentales más racionales, con el fin de disminuir el porcentaje de error respecto a la relación de transmisión teórica.

Referencias

- [1] Ángeles, A. J. (1978). Análisis y Síntesis Cinemáticas de Sistemas Mecánicos, Limusa Wiley: Ciudad de México.
- [2] Rutter J. (2000).). Geometry of Curves. Chapman H. Mathematics Series Ingles 1st Edition.
- [3] Norton, R. L. (2013). Diseño de Maquinaria. McGraw-Hill. Quinta Edición.
- [4] Stewart, J. (2001). Cálculo multivariable. Thomson. Cuarta edición.
- [5] Fogarasy A., Smith M. (1998). The influence of manufacturing tolerances on the kinematic performance mechanisms. Part C: Journal of Mechanical Engineering Science. Proc. Inst. Mech. Engrs. Vol 209. Pp. 35-45.
- [6] Mendoza, J. Otero L. (2002). Influencia de las tolerancias dimensionales de fabricación de los eslabones de un mecanismo de cuatro barras sobre el posicionado preciso del elemento de salida. Revista Universidad, Ciencia y Tecnología. Vol. 6 Núm. 21. Pp. 43-49.
- [7] Mott, R. (1992). Diseño de Elementos de Máquinas. Edición Prentice Hall, segunda edición. México
- [8] Norma ISO 286. Español



- [9] Norma ASME Y14.5 2009 Español.
- [10] Rothenhofer G., Walsh C., Slocum A. (2010). Transmission ratio based analysis and robust design of mechanisms. Precision Engineering 34. Pp. 790–797.



Un primer acercamiento al diseño de sombreros asistido por una red adversaria-generativa

Salazar-Colores Sebastián¹, Carrillo-López David², Gerardo Flores¹, Moya-Sánchez E. Ulises³, Ayar-Rojas Nadia⁴

Centro de Investigaciones en Óptica (CIO)¹
Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)²
Gobierno de Jalisco (Director de IA) y Universidad Autónoma de Guadalajara (UAG)³
Universidad de Guanajuato (UG)⁴

sebastian.salazar@cio.mx

Resumen

La actual revolución industrial 4.0 tiene como una de sus metas principales el incorporar inteligencia artificial a los procesos industriales para realizarlos de manera más eficiente. Dado que en México la inteligencia artificial no es tan conocida y mucho menos adoptada por las pequeñas y medianas empresas, en este trabajo se propone un primer acercamiento a la incorporación de un sistema de asistencia al proceso creativo basado en inteligencia artificial, el cual está basado en una red neuronal adversaria generativa completamente conectada. El caso de estudio de la investigación es el diseño de sombreros. Los resultados obtenidos muestran que la red neuronal es capaz de aprender y combinar las características que definen a un sombrero permitiendo generar diseños que no ha visto previamente. Más aún, los resultados también permiten vislumbrar que la red neuronal utilizada permitiría la asistencia en el diseño de una gran cantidad de productos importantes para pequeñas y medianas empresas del país como instrumentos musicales, ropa, calzado y artesanías.

Palabras clave: Diseño asistido por inteligencia artificial, Redes adversarias-generativas, Redes adversarias-generativas convolucionales profundas, Diseño sombreros.

1. Introducción

Actualmente existen poblaciones de México que basan su economía en la producción de determinados productos, de carácter mayormente artístico donde la creatividad de diseños es un factor muy importante. Como ejemplo está Paracho, Michoacán. y sus guitarras, donde cierta variedad de materiales y dimensiones juegan un papel importante en el diseño y sonido final del instrumento [1]; en el Estado de Guanajuato encontramos a Moroleón y Uriangato dedicados a la elaboración de ropa [2], San Francisco del Rincón con sombreros, cuyo mercado destaca por la cantidad de exportaciones y León con sus empresas fabricantes-exportadoras de calzado [3, 4], con un mercado exportador también muy importante. En el contexto del turismo, están los alebrijes, cuyo origen tiene por lugar la Ciudad de México, pero son base de la economía de los pueblos de Oaxaca, Arrazola Xoxocotlán y San Martín Tilcajete [5, 6]. Para todos los artículos mencionados, su gran variedad de diseños exteriores busca que lo atractivo de los mismos sea el detonante de su venta.

En particular, y enfocándonos en el caso de estudio que aborda este trabajo, el tamaño del mercado del sombrero en San Francisco del Rincón cuya dimensión alcanzó, en términos de exportaciones, más de 10 millones de dólares en 2019 [7] (sólo superado por la industria del calzado y autopartes), constituye una fuente importante de ingresos para dicho municipio. Actualmente y



debido a situaciones bien conocidas, la exportación estatal de prendas de vestir tuvo una caída importante (alrededor del 30%) [8]. Sin embargo, con la reapertura gradual de los mercados, dicha situación puede cambiar y, con fortuna, recuperarse.

Desde hace décadas, en México la tecnología aplicada a la manufactura ha sido utilizada para mejorar y hacer más eficiente la producción. Desde la introducción de máquinas de fabricación específicas hasta la inclusión de robots, la industria se ha visto beneficiada en su crecimiento y, por extraño que parezca, igualmente en el número de personal [9].

También la tecnología aplicada a la asistencia de diseño por computadora ha mejorado la capacidad y precisión de diseños. Desde finales de la década de los ochenta [10], la computadora ha encontrado lugar en distintos campos de aplicación (arte, industria, manejo de información, medicina, ciencia, entre otros). A medida que sus capacidades se han vuelto cada vez más poderosas (y su tamaño ha disminuido en una proporción similar), su utilización casi se ha estandarizado, no sólo con el uso de programas de paquetería básicos, sino con programas para aplicaciones más específicas. En este contexto, los programas CAD (de *Computer Aided Design*) juegan un papel sumamente importante y han reemplazado casi en su totalidad a las técnicas de diseño por dibujo a mano, abriendo posibilidades de mejoras en la comunicación de ideas y en la integración de los resultados [11]. Cabe resaltar que las computadoras impulsaron la tercera revolución industrial [12].

La cuarta revolución industrial llamada también Industria 4.0, la cual estamos viviendo actualmente, nos lleva al campo interdisciplinario donde no basta con saber sólo de una disciplina académica/tecnológica [12]) en el mundo. Una de las áreas con mayor impacto de esta nueva revolución es la aplicación de la inteligencia artificial y en específico el aprendizaje profundo a tareas como diseño asistido por inteligencia artificial. Por ejemplo: en el área mecánica y diseño de ejes y transejes [13], las sillas Elbo y Generico (concebidas dentro del concepto *Generative Design* [14]) [15 y 16], estructuración de ejes de ruedas de patinetas [17], y construcción automatizada de sitios *web* por inteligencia artificial [18].

Hasta donde sabemos, en la industria nacional sobre todo en pequeñas y medianas empresas no existen sistemas que brinden asistencia al diseño creativo basados en inteligencia artificial, todo se realiza mediante procesos tradicionales. El incorporar este tipo de asistencia podría incrementar la calidad y reducir los costos de los productos obtenidos. La incorporación del estado del arte en el tema de tecnologías de la información, y/o asistencia por computadora en el diseño, podría traer beneficios en este contexto.

En este trabajo se muestra un primer acercamiento y resultados iniciales al aplicar una red neuronal llamada Red adversaria-generativa convolucional profunda o *Deep Convolutional Generative Adversarial Network* para asistir al proceso creativo del diseño específicamente de sombreros. Esto se consigue alimentando a la red neuronal utilizada con miles de ejemplos de sombreros actuales. La red neuronal es capaz de generar diseños nuevos que permitirían a los diseñadores inspirarse y rediseñar de forma rápida distintos modelos. Además, es posible manipular el espacio latente de distintas salidas de la red neuronal, con lo cual pueden mezclarse características de los sombreros.

El presente trabajo está estructurado de la siguiente forma: en la sección 2, se presentan los fundamentos teóricos y conceptuales usados para la implementación de la red; en la sección 3, se explican las configuraciones usadas en el diseño del experimento; en la sección 4, se presentan los resultados obtenidos; y finalmente, se presentan las conclusiones y una perspectiva de futuras investigaciones basadas en la presente.



2. Marco Teórico

Para describir lo que es una red adversaria generativa primero es importante describir qué es el aprendizaje profundo y las estructuras más comunes que lo componen.

2.1 Aprendizaje profundo

El aprendizaje profundo o *deep learning*, es una rama del aprendizaje automático, que a su vez es una rama de la inteligencia artificial. La inteligencia artificial es un término general que se refiere a las técnicas que permiten a las computadoras imitar el comportamiento humano. El aprendizaje automático representa un conjunto de algoritmos entrenados en datos que hacen todo esto posible [21].

El aprendizaje profundo está inspirado en la estructura del cerebro humano. Los algoritmos de aprendizaje profundo intentan sacar conclusiones similares a las de los humanos analizando continuamente los datos con una estructura lógica determinada. Para lograr esto, el aprendizaje profundo utiliza una estructura de varios niveles de algoritmos llamados redes neuronales. En el aprendizaje profundo se utilizan comúnmente redes neuronales convolucionales, las cuales están formadas de capas convolucionales y funciones de activación, como el rectificador lineal (ReLU, por sus siglas en inglés), Leaky ReLU, sigmoide o tangente hiperbólico. A continuación, describimos de forma general estos conceptos.

2.2 Redes neuronales convolucionales

Una red neuronal convolucional (CNN, por sus siglas en inglés) es una conocida arquitectura de aprendizaje profundo inspirada en el mecanismo natural de percepción visual de los seres vivos. Una CNN puede obtener representaciones efectivas de la imagen original, lo que hace posible reconocer patrones visuales directamente de los píxeles en bruto, con poco o ningún preprocesamiento.

2.3 Capas convolucionales

El componente principal de una CNN es la capa convolucional, la cual contiene un conjunto de filtros o kernels cuyos parámetros deben ser aprendidos. Entonces podemos obtener diferentes mapas de características haciendo una convolución de la entrada con el kernel aprendido y la aplicación que se encuentra en una función de activación no lineal [21]. El valor de la característica en la posición (i, j) en el k -ésimo mapa de características de la l -ésima capa, $z_{i,j,k}^l$, es calculada como sigue:

$$z_{i,j,k}^l = w_k^{lT} x_{i,j}^l + b_k^l \quad (1)$$

donde w_k^l y b_k^l son el vector de peso y el término de sesgo del k -ésimo filtro de la l -ésima capa respectivamente; y $x_{i,j}^l$ es la ventana de entrada centrada en la posición (i, j) en l -ésima capa [21]. Aunque normalmente se recorren los índices del kernel desde (i, j) de uno en uno, algunas veces se realizan cambios de posición, pasos o saltos con distintos intervalos o strides. El proceso inverso de la convolución es conocido como convolución transpuesta y tiene como meta incrementar el tamaño del mapa de características. Con la finalidad de aumentar las capacidades en el entrenamiento de las redes neuronales, se han creado técnicas de regularización como el dropout o normalización por lotes [21].

2.4 Funciones de activación

Existe una gran variedad de funciones de activación utilizadas en el aprendizaje profundo, entre las más importantes podemos mencionar la función sigmoide, tangente hiperbólico y la unidad lineal rectificadora (ReLU) [21]. La función sigmoide es definida como:

$$a_{i,j,k} = \frac{1}{1+e^{-z_{i,j,k}}} \quad (2)$$

Esta función es comúnmente utilizada en regresión logística, ya que produce un rango más suave o continuo de valores entre 0 y 1 que la función escalonada. La función tangente hiperbólica (tanh) es una función similar a la función sigmoïdal que permite tener valores entre -1 y 1.

ReLU [21] es un tipo de función de activación que se utiliza en casi todas las redes neuronales convolucionales. Matemáticamente, se define como:

$$a_{i,j,k} = \max(z_{i,j,k}, 0) \quad (3)$$

ReLU es lineal para todos los valores positivos, y cero para todos los valores negativos. Por su definición, la función ReLU es barata de calcular, converge más rápido y también induce a la dispersión en unidades ocultas y permite a la red obtener fácilmente representaciones dispersas [21]. Finalmente, la función Leaky ReLU es una variante de ReLU que tiene una respuesta para valores negativos.

2.6 Redes neuronales adversarias generativas

Las Redes Generativas Adversas (GAN, por sus siglas en inglés) presentadas en 2014 [22], son una clase de algoritmos de inteligencia artificial que se utilizan en el aprendizaje no supervisado, implementados por un sistema de dos redes neuronales que compiten entre sí. Las GAN se implementan usando capas convolucionales y deconvolucionales (convolucionales transpuestas) para el generador G , que está diseñado para producir salidas que no pueden distinguirse de las imágenes reales. Por otro lado, una red neuronal convolucional (CNN) constituye el discriminador D , capaz de hacer lo mejor posible en la detección de falsificaciones hechas por (G). Se puede ver este ejemplo en la Figura 1.



Figura 1. Analogía del enfoque utilizado en una red neuronal adversaria generativa.

2.7 Arquitectura GAN

Definimos un discriminador (D) y un generador (G) que se entrenan simultáneamente para resolver el problema adversario min-max [22]:

$$\min_G \max_D V(D, G) = E_{I \sim p_{train}(I)} [\log D(I)] + E_{z \sim p_G(z)} [\log (1 - D(G(z)))] \quad (4)$$

La entrada del generador G es una imagen de ruido aleatorio z , es decir, $G(z)$. $G(z)$ se utiliza como entrada al discriminador D junto con la imagen real I . La salida de D es la clasificación de éstas, 0 para las imágenes generadas y 1 para la real. Nuestro objetivo final es entrenar G para

generar imágenes que el discriminador clasifique como reales. En la Figura 2 es mostrado un diagrama que muestra en términos generales el funcionamiento de una GAN.

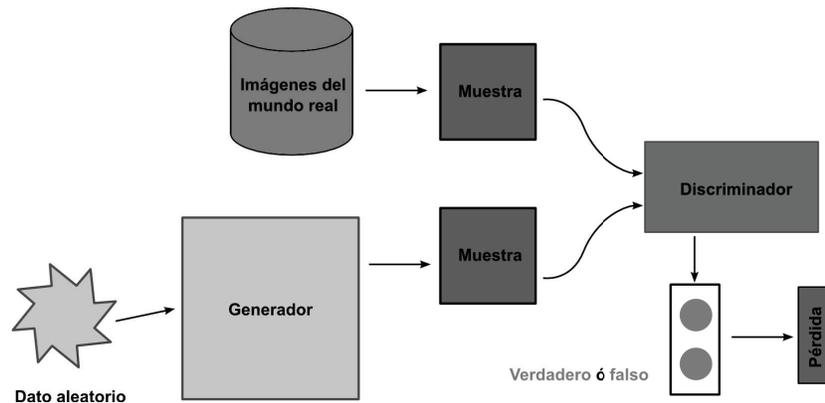


Figura 2. Esquema de una GAN donde se puede observar la relación generador-discriminador.

2.8 Red neuronal adversaria-generativa convolucional profunda

DCGAN es una de las variantes más utilizadas de GANs. Sus principales características en el generador G [23]:

- Reemplaza todas las operaciones de *max pooling* con pasos convolucionales mayores a 1 (stride > 1).
- Usa la convolución transpuesta para el sobremuestreo.
- Elimina las capas completamente conectadas.
- Usa la normalización de lotes (*batch normalization*) excepto en la capa de salida del generador y la capa de entrada del discriminador.
- Usa la función de activación ReLU excepto para la salida que usa tanh.

Y en el discriminador D :

- Usa la función de activación LeakyReLU.
- Es un clasificador clásico con una etapa de convoluciones y otra completamente conectada.

En la Figura 3 se muestra la estructura clásica de una DCGAN.

3. Experimentación

En esta sección se presenta la configuración y proceso de obtención de los conjuntos de datos utilizados, la arquitectura, algunos ejemplos y la comparación de distintos resultados.

3.1 Hardware utilizado

Los experimentos fueron implementados en un procesador Ryzen 7 1700x con 64 GB de memoria RAM y una tarjeta gráfica Nvidia GTX 1070 Ti. El sistema operativo utilizado es Linux Ubuntu 20.04 dotado de Python 3.7, librerías OpenCV, Keras 2.3.0 y Tensorflow 1.15.

3.2 Datos utilizados

Dado que no fueron encontradas bases de datos públicas disponibles, fue necesario recolectar las imágenes de sitios públicos. Las imágenes obtenidas para la creación de la base de datos propia a juicio de los autores deben tener las siguientes características:

- a) La cantidad de imágenes deben ser de al menos el orden de miles.
- b) Las imágenes deben mostrar claramente al sombrero.
- c) Los sombreros deben aparecer en el centro sin otros objetos alrededor.
- d) El fondo debe de ser de preferencia del mismo color, en nuestro caso fue elegido blanco.
- e) Los sombreros deben ser tan diversos como sea posible.
- f) No se deben infringir derechos de autor.

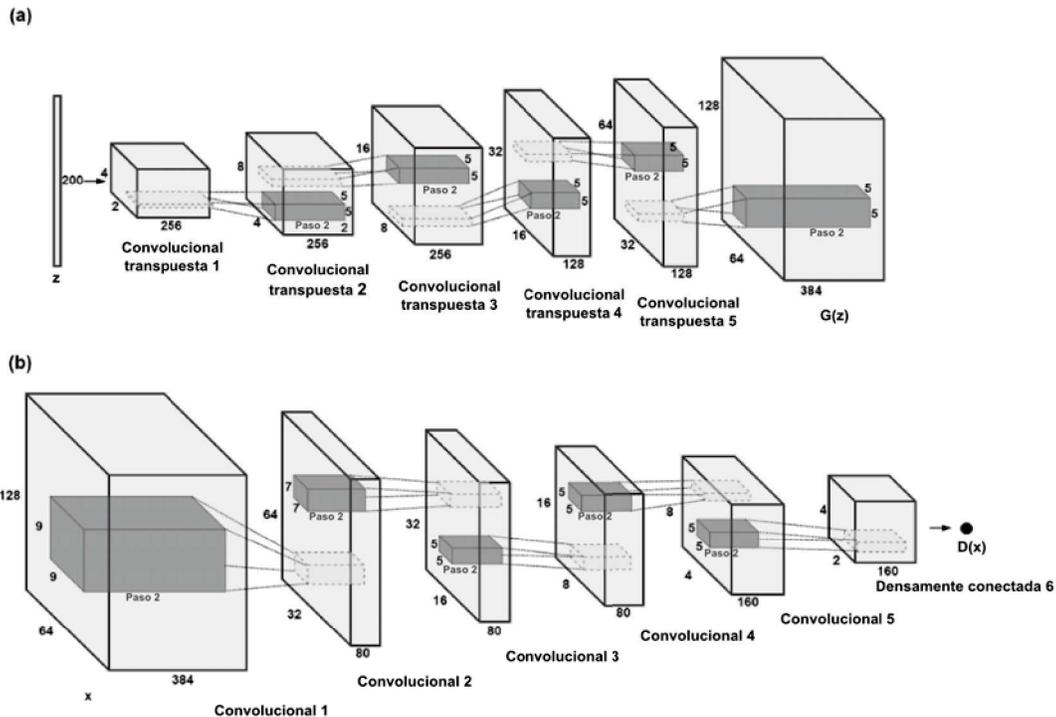


Figura 3. Arquitectura clásica DCGAN. (a) Red neuronal generador G . (b) Red neuronal discriminador D .

Por lo anterior, fue decidido obtener las imágenes de dos sitios públicos: Bing images y Amazon, las cuáles permiten descargar información libremente con fines de investigación. Para apoyar el proceso de búsqueda y descarga se hizo uso de la técnica de *raspado web*, mejor conocida como *web scrapping*. *Web scrapping* es un término utilizado para describir el uso de un programa o algoritmo para extraer y procesar grandes cantidades de datos de la web. Específicamente fueron utilizados los programas de python de código abierto [24] y [25]. EL proceso de depurado y alineación fue realizado manualmente. La base de datos fue conformada de 2456 imágenes distintas de sombreros. Algo importante de mencionar es que adicionalmente las imágenes obtenidas fueron redimensionadas a una resolución de 256×256 píxeles, lo cual permite que el almacenamiento sea fácil de realizar. Además, como se mostrará más adelante, este tamaño permite entrenar exitosamente y en tiempos relativamente cortos la red neuronal. En la Figura 4 se muestra un ejemplo del proceso llevado a cabo.

3.3 Entrenamiento

La red DCGAN utilizada fue tomada de [24]. La decisión de utilizar esta implementación está basada en su diseño de propósito general y que ha sido modificada para incrementar el rendimiento de la versión original presentada en [25]. Las modificaciones más importantes fueron las siguientes:

- a) Un kernel más grande y más filtros: El tamaño del kernel de las capas 2 y 3 del generador se incrementó en 1. A medida que los kernels más grandes cubren más área debería capturar más información.



- b) Intercambio de etiquetas aleatorio (Generado=Verdadero, Real=Falso): Esto ayuda con el flujo de gradientes.
- c) Ruido de instancia agregado para estabilizar el entrenamiento: Se agregó una capa de ruido Gaussiano en la parte superior del Discriminador, con la idea de evitar que el Discriminador sea demasiado bueno demasiado pronto, lo que impedirá que el Generador aprenda y posiblemente converja.



Figura 4. Ejemplo de sombreros en la base de datos generada.

En la Tabla 1 se muestra la arquitectura de la red neuronal Discriminadora *D* donde RG es el ruido Gaussiano agregado a una imagen de entrada, C1 denota Capa convolucional->Activación Leaky ReLU->Regularización vía Dropout(0.25)-> Operación average pooling); C2 denota Capa convolucional -> Regularización vía Normalización de Batch->Activación Leaky ReLU->Regularización vía Dropout(0.25)->Operación average pooling. FC denota la etapa completamente conectada, la cual se forma por un aplanamiento Flatten (4096)->Capa densa (128)->Activación Leaky ReLU->Capa densa (1)->Activación Sigmoide. En la Tabla 2 se muestra la arquitectura del Generador *G*. C1 denota Capa convolucional transpuesta->Activación ReLU; C2 denota Capa convolucional -> Regularización vía Normalización de Lotes->Activación ReLU->Sobremuestreo (2, 2). C3 denota Capa convolucional -> Activación ReLU->Sobremuestreo (2, 2). Finalmente C4 indica Capa convolucional -> Activación Sigmoide.

Tabla 1. Arquitectura del Generador *D*.

Capa	Filtros de Convolución	Kernel	Longitud pasos (Stride)	Definición	Tamaño	
					Entrada	Salida
0	RG (0.2)				(256, 256, 3)	(256, 256, 3)
1	8	3	1	C1	(256, 256, 3)	(128, 128, 8)
2	16	3	1	C2	(128, 128, 8)	(64, 64, 16)
3	32	3	1	C2	(64, 64, 16)	(32, 32, 32)
4	64	3	1	C2	(32, 32, 32)	(16, 16, 64)
5	128	3	1	C2	(16, 16, 64)	(8, 8, 128)
6	256	3	1	C2	(8, 8, 128)	(4, 4, 16)
7	FC				(4, 4, 16)	(1)

Tabla 2. Arquitectura del Generador G.

Capa	Filtros de Convolución	Kernel	Longitud pasos (Stride)	Definición	Tamaño	
					Entrada	Salida
1	256	4	1	C1	(1, 1, 4096)	(4, 4, 256)
2	256	4	1	C2	(4, 4, 256)	(8, 8, 256)
3	128	4	1	C2	(8, 8, 256)	(16, 16, 128)
4	64	3	1	C2	(16, 16, 128)	(32, 32, 64)
5	32	3	1	C2	(32, 32, 64)	(64, 64, 32)
6	16	3	1	C2	(64, 64, 32)	(128, 128, 16)
7	8	3	1	C3	(128, 128, 16)	(256, 256, 8)
8	3	3	1	C4	(256, 256, 8)	(256, 256, 3)

Para entrenarla fue utilizado un tamaño de lote de 16, una tasa de aprendizaje de 0,0002, y como ya ha sido mencionado, una resolución de 256 por 256 píxeles para las imágenes de salida. Los pesos fueron inicializados usando una distribución Gaussiana con media cero y una desviación estándar de 0.02. Se utilizó el Momentum Adaptable (ADAM) como una función de optimización. La red fue entrenada por 250 épocas a lo largo de 12 horas. En la Figura 5 se muestran resultados de la red neuronal en distintas épocas, donde puede observarse el aprendizaje obtenido en el entrenamiento.



Figura 5. Sombreros generados en distintas épocas.

Es sumamente importante mencionar que los sombreros obtenidos por la red neuronal **no existen!**, son generados mediante el espacio latente z . Es por ello que podrían servir de inspiración y apoyo al desarrollo creativo de nuevas tendencias para el diseñador. En la Figura 6 se muestran diversos diseños de sombreros generados por la red neuronal, con diversas configuraciones aleatorias del espacio latente z .



Figura 6. Sombreros generados con distintas configuraciones de z .

A pesar de que los resultados obtenidos por la DCGAN son en su mayoría buenos, existen algunas configuraciones del espacio latente z que producen aberraciones. En la Figura 7 se muestran algunos ejemplos de casos generados fallidos. Para disminuir estos errores en la literatura se plantean tres soluciones: mejorar la estructura de la red, incrementar el número de muestras y mejorar la calidad de las muestras.

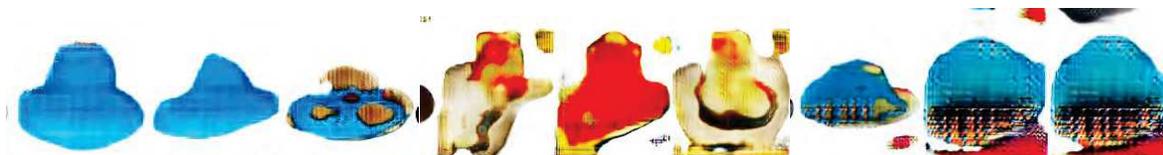


Figura 7. Aberraciones y casos fallidos producidas por la DCGAN.

4. Conclusiones

En este trabajo es propuesta la aplicación de una red neuronal DCGAN para la asistencia en el proceso creativo del diseño de sombreros. Los satisfactorios resultados de este primer acercamiento muestran la factibilidad del proceso propuesto, ya que la red utilizada es capaz de comprender y replicar por sí sola las características intrínsecas de este objeto. Hemos observado que en términos generales la generación de la red neuronal aproxima muy bien el concepto de sombrero; sin embargo, hemos detectado que los sombreros generados presentan artefactos y deformidades en algunos casos. Consideramos que esto puede ser mejorado adecuando la red neuronal, incrementando el tamaño de la base de datos y mejorando la selección y preprocesamiento de sombreros. Es por lo anterior que como trabajo futuro se plantea la sintonización adecuada de la arquitectura e hiper-parámetros de la red neuronal para incrementar su rendimiento, el incremento de la base de datos generada y el desarrollo de una interfaz de usuario. Una vez conseguido lo



anterior, se comenzarán a realizar pruebas en entornos reales agregando la manipulación del espacio latente del generador. Adicionalmente, se realizarán pruebas con la generación de otro tipo de objetos.

Referencias

- [1] Guitarras de Paracho, <<https://www.guitarrasdeparachomichoacan.com/>>, septiembre, 2020.
- [2] Plaza Textil, <<https://plazatextil.com.mx/>>, septiembre, 2020.
- [3] Gracia Hernández, M. Teoría industrial: un vínculo con el clúster del calzado. Tla-melaua, 11(42), pp.116-135, 2017.
- [4] ¿Por qué León es la capital del Calzado?, <<https://fabricantesdeleon.com/leon-la-capital-mundial-del-calzado/#:~:text=Guanajuato%20es%20la%20regi%C3%B3n%20m%C3%A1s,curtido%20y%20acabado%20de%20piel.>>, septiembre, 2020
- [5] Alebrijes: la fascinante historia detrás de estas coloridas artesanías mexicanas, <<https://mymodernmet.com/es/que-es-un-alebrije/>>, septiembre, 2020.
- [6] Oaxaca Lindo, <<https://oaxacalindo.com/alebrijes/>>, septiembre, 2020.
- [7] Cartera económica COFOCE, cierre oficial 2019, COFOCE (Coordinadora de Fomento al Comercio Exterior de Guanajuato), <<https://cofoce.guanajuato.gob.mx/>>, Guanajuato, México.
- [8] Cartera Económica COFOCE, primer semestre 2020, COFOCE (Coordinadora de Fomento al Comercio Exterior de Guanajuato), <<https://cofoce.guanajuato.gob.mx/>>, Guanajuato, México.
- [9] Brown F., Domínguez L., Nuevas tecnologías en la industria maquiladora de exportación, Comercio exterior, vol. 39, núm. 3, pp. 215-223, México, marzo de 1989.
- [10] Muy Interesante, Cómo será el ordenador de los noventa, Muy interesante, año VI, núm. 12-011289, pp. 4-12, México, junio de 1989.
- [11] Rojas Lazo O., Rojas Rojas L., Diseño asistido por computador, Ind. Data, núm. 9, 2006.
- [12] Schab K., La cuarta revolución industrial, Penguin Random House, Editorial España, España.
- [13] Chagoyén Méndez C. A., Un sistema basado en casos para el diseño de árboles, Centro Azúcar, vol. 27, no. 4, 2000, p. 64+, septiembre, 2020.
- [14] Autodesk, Generative Design, <<https://www.autodesk.com/solutions/generative-design>>, septiembre, 2020.
- [15] Presten B., Hutchinson J., Bass C. The Elbo Chair, Dreamcatcher, Autodesk, <<https://gallery.autodesk.com/fusion360/projects/68792/elbo-chair--generated-in-project-dreamcatcher-made-with-fusion-360>>, septiembre, 2020.
- [16] Spade Studio, Generico Chair, <<https://www.spade-studio.de/projects/generico-chair/>>, septiembre, 2020.
- [17] Turney D., Thrashing the Future of Design With Topology Optimization and Skateboard Trucks, Redshift, Autodesk, <<https://www.autodesk.com/redshift/skateboard-trucks-topology-optimization/>>, septiembre, 2020.
- [18] Läufer A., Artificial intelligence and design: my new relationship crisis, t3n, <<https://t3n.de/news/kuenstliche-intelligenz-design-1079033/>>., septiembre, 2020.
- [21] Goodfellow, I., Bengio, Y., Courville, A. and Bengio, Y. Deep learning (Vol. 1). Cambridge: MIT press, 2016.
- [22] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. Generative adversarial nets. In Advances in neural information processing systems (pp. 2672-2680), 2014.
- [23] Radford, A., Metz, L. and Chintala, S.. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434, 2015.
- [24] DCGAN256, <<https://github.com/t0nberryking/DCGAN256/>>, septiembre, 2020.
- [25] DCGAN256, <<https://github.com/manicman1999/GAN256/>>, septiembre, 2020.



Modelado y Control Lineal de Sistemas Físicos Mediante Prototipos: Una Estrategia de Aprendizaje

Velázquez Izguerra Joel, Pérez Silva Ricardo Bladimir, Soria Medina Abraham
Estefano, Peñaloza Mendoza Guillermo Rey

Departamento de Ingeniería Biomédica, Instituto Tecnológico Superior de Pátzcuaro, Michoacán,
México

joel_vizguerra@hotmail.com, persilricvla@gmail.com, estefano01020@gmail.com, grey@itspa.edu.mx

Resumen

La introducción de la robótica educativa a la formación académica en los últimos años, como herramienta esencial en el proceso de aprendizaje en el área de programación, ciencias básicas y electrónica ha incrementado exponencialmente. Por ejemplo, el uso de kits de robots para la introducción a la lógica de programación de actividades simples, o el uso de robots móviles para el desplazamiento mediante el uso de coordenadas y matrices, o como en el caso que nos interesa, cuando se inicia en el mundo de los sistemas de control, la teoría nos prevé de mucha información que en ocasiones no se alcanza a comprender realmente cómo influye realmente en los sistemas o la importancia de la manipulación de cada constante dentro del mismo. Por ello, en este trabajo, se plantea el diseño e implementación de un par de prototipos robóticos cuya finalidad es proporcionar una demostración práctica del control Proporcional, Integral y Derivativo (PID), así como los efectos de la variación de las constantes del mismo. En estos prototipos se pueden variar las constantes de error para observar, tanto física como gráficamente en tiempo real, la manera en que influyen en la acción de control. El par de prototipos son dispositivos que se encargan de mantener su posición ante cualquier perturbación que sea introducida en su sistema, para ello, cuentan con un módulo MPU6050, el cual permite obtener el ángulo de inclinación del dispositivo con respecto a una referencia, para así mismo efectuar la acción de control respecto a la señal obtenida.

Palabras clave: Robótica educativa, Sistemas de control, PID, Módulo, MPU6050.

1. Introducción

En la actualidad los dispositivos robóticos son parte de nuestra vida diaria, por lo cual, la necesidad de integrarlos como parte de nuestra formación ha crecido rápidamente, llegando a ser una herramienta fundamental para el proceso de enseñanza-aprendizaje. Empleando el enfoque constructivista en la educación se referencia a Jean Piaget y a Seymour tomando como base que el conocimiento se obtiene por construcciones mentales, lo que genera la teoría fundamentalista del uso de la tecnología en los centros educativos [1].

Este modelo permite la implementación de cualquier herramienta para lograr un objetivo educacional, particularmente, ha potencializado, el uso de dispositivos robóticos en la enseñanza para la ejemplificación de técnicas de complejas como lo son el control de sistemas, la lógica computacional, el uso de matrices, entre otros. Un caso particular dentro de los sistemas robóticos que ha atraído el interés en los últimos años son los dispositivos equilibristas o balancines, los cuales se basan en el principio del péndulo invertido. Estos consisten en un péndulo o varilla que gira libremente sobre uno de sus extremos, y a diferencia de los demás robots sobre ruedas que normalmente son estables, el balancín es un sistema inestable por naturaleza, debido a que no es completamente actuado.

2. Marco teórico

La implementación de la robótica como herramienta de apoyo en la educación se originó en 1983 dentro del Laboratorio del Instituto Tecnológico de Massachusetts (MIT) cuando se desarrolló el primer lenguaje de programación educativo para niño llamado logos [2]. El surgimiento de kits de robótica ha permitido que la inserción se dé de manera integral, ya que éstos se caracterizan por no exigir un conocimiento avanzado de electrónica o de programación a los estudiantes que van tomando estas herramientas por primera vez. Así mismo, al incrementar los conocimientos, la introducción de sistemas independientes a los kits es fundamental para la implementación de sistemas más complejos que permitan ejemplificar y demostrar teorías o conceptos más complicados.

Un ejemplo claro de lo mencionado anteriormente, es la complejidad en el diseño de los robots balancines, debido a que necesitan ser controlados constantemente para mantenerse en sentido vertical y que la reacción ante perturbaciones externas sea rápida y no oscilante, esto permite demostrar técnicas de control diferentes y la importancia de las acciones de control. La aplicación de este tipo de robots a nivel comercial ha sido muy reconocida y es usado principalmente en el transporte personal y en juguetes didácticos para niños [3]. En la Figura 1, puede observarse un tipo de robot balancín, cuyo propósito es mantenerse en su posición vertical aún ante perturbaciones externas generadas al moverse hacia adelante o hacia atrás.



Figura 1. Robot balancín.

Siguiendo con los ejemplos, otro dispositivo equilibrista de uso común son los robots voladores, también llamados drones, los cuales requieren un control preciso que permita mantener el dispositivo en una posición horizontal y que además permita su control remoto. En estos se deben controlar las fuerzas de empuje que generan cada una de las hélices sobre las demás, así como contrarrestar el efecto de la gravedad, para esto es importante mantener la velocidad de giro de cada hélice de tal manera que sea la adecuada para mantener horizontal el dron.

Hoy en día, las aplicaciones de los sistemas basados en drones son utilizados de manera cotidiana, para filmación, entretenimiento, rescate, o incluso en el área agrícola, permitiendo fumigar zonas mediante control a distancia [4]. En la Figura 2 puede observarse un dron comercial cuya finalidad es el entretenimiento.



Figura 2. Dron comercial.

Ambos dispositivos hacen uso del control lineal denominado PID, cuyas siglas representan los aspectos proporcional, integral y derivativo que conforman la acción de control. El control PID fue desarrollado por el matemático, ingeniero y científico ruso Nikolai Fyodorovich Minorsky. La idea fundamental detrás de un controlador PID es leer la variable del proceso a la salida y obtener a la salida del actuador un término de compensación para la entrada sumando la parte proporcional, integral y derivativa [5]. En la Figura 3 se observa un esquema general de un proceso de una planta con retroalimentación, donde se aplica el controlador al sistema para obtener una acción de control que corrija a la planta.

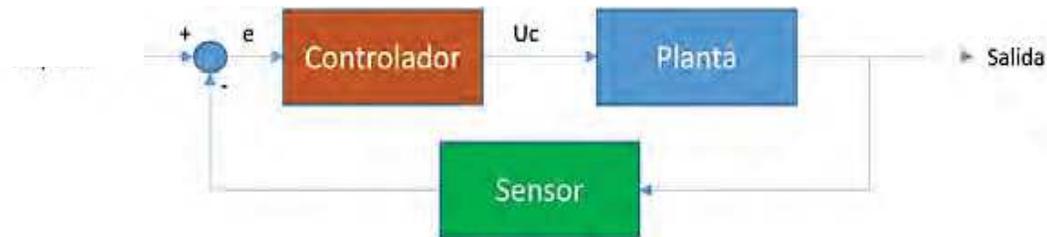


Figura 3. Aplicación del control en un sistema.

La diferencia entre la variable de proceso y la referencia que se desea conseguir es utilizada por el algoritmo de control para determinar la salida que se envía a la planta del sistema. Esta salida se obtiene a partir del algoritmo de control PID, el cual está formado por tres parámetros distintos: el proporcional, el integral y el derivativo. Estos tres parámetros se suman para así obtener la salida necesaria para conseguir el control [6]. Este proceso se puede observar de manera esquematizada en la Figura 4.

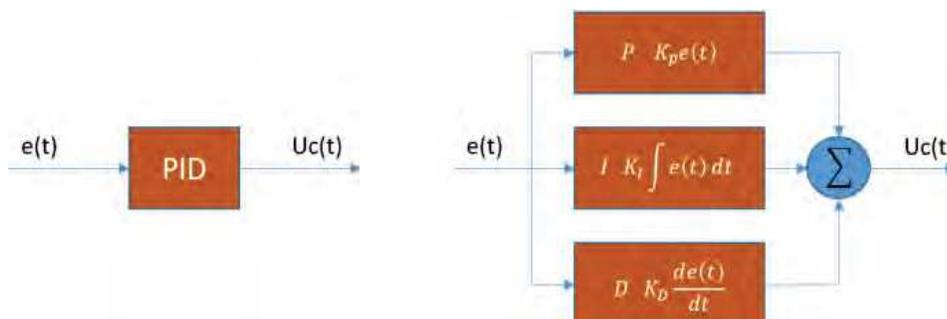


Figura 4. Control PID.

La variación de esos 3 parámetros conlleva a una distinta acción de control que puede o no estabilizar el sistema, por lo que se pueden tener 2 parámetros correctos y aun así el sistema no funcionará correctamente. Para encontrar los valores adecuados de dichas constantes, se pueden tomar en cuenta las condiciones que se muestran en la Tabla 1 [7].



Tabla 1.- Relaciones entre el sistema y las variables del PID

	Kp Aumenta	Ti Disminuye	Td Aumenta
Estabilidad	Se reduce	Disminuye	Aumenta
Velocidad	Aumenta	Aumenta	Aumenta
Error estacionario	No eliminado	Eliminado	No eliminado

3. Planteamiento del problema

Las técnicas de modelado matemático y de control clásico son de vital importancia en la industria actual, debido a que permiten la optimización de procesos, así como la automatización de los mismos. Esto ha llevado la introducción de estas herramientas en la formación de ingenieros, sin embargo, el aprendizaje de estas no es cosa simple. Por una parte, los modelados matemáticos de los sistemas físicos emplean matemáticas avanzadas, que si bien, su entendimiento es difícil para la mayoría, la visualización de su aplicación práctica es más compleja; esto junto a los diversos tipos de control aplicados en la industria, como el proporcional, integral y derivativo (PID), y el cálculo de diversas constantes para un óptimo desempeño, generan en los estudiantes problemas para entender la importancia práctica de esto y lo que representa cada uno de los valores obtenidos en el funcionamiento de los sistemas.

3.1 Solución propuesta.

Diseñar dos prototipos robóticos que requieran el uso de un control PID, el primero de ellos en base a un péndulo invertido simple y el segundo siendo un sistema motor-hélice-balancín, con la finalidad de que se logre tanto observar como registrar el comportamiento de los sistemas al efecto de una perturbación y al cambio de los valores de las constantes requeridas por dicho control.

De esta manera se espera lograr la comprensión de la importancia del modelado y las técnicas de control en las aplicaciones prácticas, así como visualizar los efectos de la variación en los parámetros de las acciones de control independientes dentro del control PID.

4. Metodología

El primer prototipo consiste en un balancín con motores brushless cuyo objetivo es lograr la estabilización en el ángulo que el usuario determine. Los valores de las constantes Kp, Ki y Kd pueden ser ajustadas por el usuario a través de un teclado, para observar la manera en que influyen en el control del sistema y así comprender mejor el funcionamiento del control PID al poder observar de manera directa su aplicación y como afectan los valores de las constantes al comportamiento del sistema controlado.

Material utilizado:

- 2 motores brushless
- 2 propelas
- 2 Electronic Speed Control (ESC)
- Arduino Mega
- Teclado matricial 4x4
- 2 baleros
- Tornillos y tuercas
- 1 MPU6050
- Batería LiPo 4S
- Barra de 55cm
- Madera tipo MDF
- Eje metálico
- Laptop
- 1 Interruptor

El armado de la parte electrónica se observa en la Figura 5, y consta de varios componentes como los motores brushless, los cuales deben de ser conectados con su correspondiente Electronic Speed Controller (ESC), este será el encargado de mandar las señales correctas al motor para que este gire a la velocidad deseada, del ESC salen cuatro cables, dos de ellos para alimentarlo con la batería LiPo mientras que los dos restantes son señal y GND, el pin de señal se conecta a cualquier pin digital del Arduino mientras que GND es conectado a tierra en el microcontrolador. Para utilizar el módulo MPU6050 solo hacen falta cuatro de sus pines, de los cuales dos son para la alimentación (3.3V y GND) y los dos restantes (SCL y SDA) que son los encargados de realizar el envío de datos hacia el Arduino mediante protocolo I2C. Por último, se realiza la conexión del teclado matricial el cual permite ingresar los valores deseados del ángulo de inclinación en el que se desea que el prototipo se mantenga estable y de las constantes k_p , k_i y k_d para la acción de control.

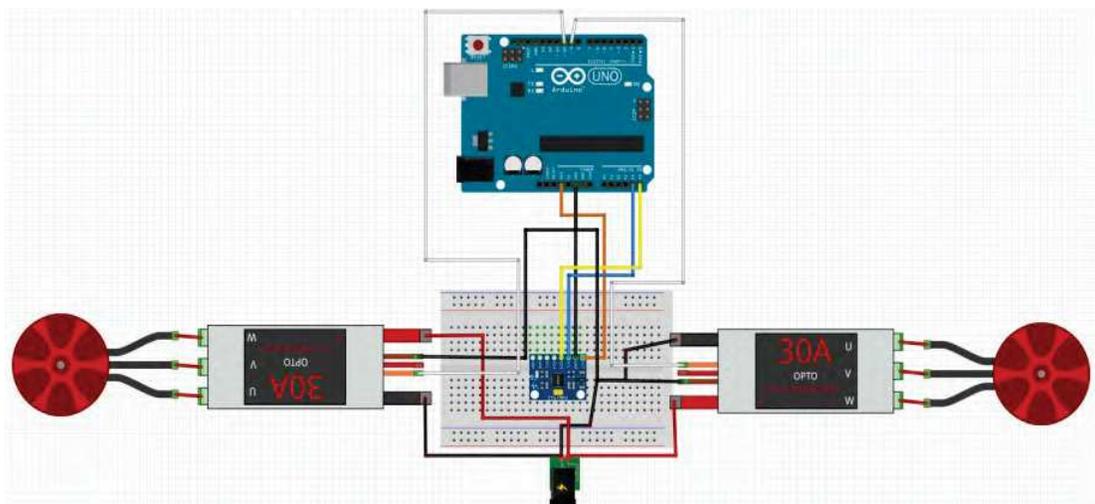


Figura 5. Diagrama de conexión del primer robot balancín.

Para la construcción de la estructura física del prototipo, se puede utilizar una barra de madera o de metal, a la cual le fue marcado el centro para posteriormente realizar una perforación en dicho lugar, y después fue introducido el eje sobre el cual el prototipo se balancearía. Este eje fue fijado a la barra mediante tuercas y arandelas; posteriormente se elaboró la base del prototipo, la cual fue realizada principalmente con madera MDF, dejando una altura de aproximadamente 30cm respecto al suelo. De esta manera se permitió que el eje principal tuviese un ángulo de inclinación máximo de aproximadamente $\pm 45^\circ$; 2 soportes fueron colocados en los extremos de la base, en donde se colocaron los baleros que facilitaron el desplazamiento del prototipo.

A continuación, la barra fue colocada entre los 2 baleros. Por último, fueron colocados los motores en los extremos de la barra, así como el microcontrolador y el módulo MPU650. Estos últimos fueron ubicados lo más centrado posible entre cada motor. El resultado final del prototipo se aprecia en la Figura 6.

El segundo prototipo consiste en un robot balancín de péndulo invertido. El diseño del robot balancín fue realizado mediante el software SOLIDWORKS®, como se observa en la Figura 7. El robot cuenta con tres partes diseñadas y elaboradas mediante la tecnología de diseño e impresión 3D las cuales corresponden a la base de soporte para los motores, la estructura para la electrónica y la tapadera de la estructura. El acceso al usuario para la modificación de las constantes K_p , K_i y K_d es facilitado mediante botones que permiten aumentar o disminuir sus valores y de esta manera observar el efecto que tienen en la acción de control del prototipo, de esta manera se evita el tener que conectar el prototipo a la computadora cada vez que se desee cambiar el valor de alguna constante, facilitando el proceso de calibración del mismo.

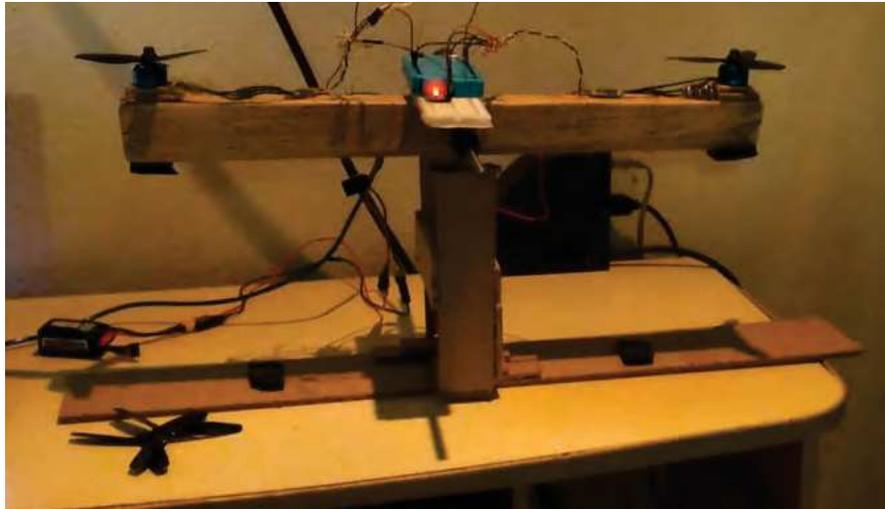


Figura 6.- Balancín con motores brushless.



Figura 7.- Diseño en 3D de la estructura del segundo robot balancín.

Material utilizado:

- 2 motorreductores
- Arduino Nano
- Llantas para motorreductores
- Driver L298N
- Botones
- Piezas impresas en 3D
- Tornillos y tuercas
- 1 MPU6050
- Batería LiPo 7.4V
- Laptop
- Cables para conexión
- Switch

En la Figura 8 se observa el esquema de conexión de los componentes electrónicos del robot. Mediante botones colocados en la superficie del robot, se establecerá el tipo de controlador PID, mediante los valores que se asignen a las constantes de la acción de control, el cual tiene como objetivo responder ante las fuerzas que intenten desestabilizar el sistema (fuerzas que quieran inclinar al robot) para así poder responder rápidamente ante dichas fuerzas y evitar que el robot pierda la estabilidad. La salida de este control PID se verá reflejado en los actuadores para los cuales se utilizarán motorreductores con ayuda de un puente H L298N y alimentados con una batería LiPo de 7.4v. La función de estos motorreductores es la de girar correspondientemente a la fuerza que esté intentando

desestabilizar el prototipo su posición vertical, esto para garantizar el equilibrio del robot balancín. En la Figura 9 se observa el prototipo finalizado del segundo robot balancín.

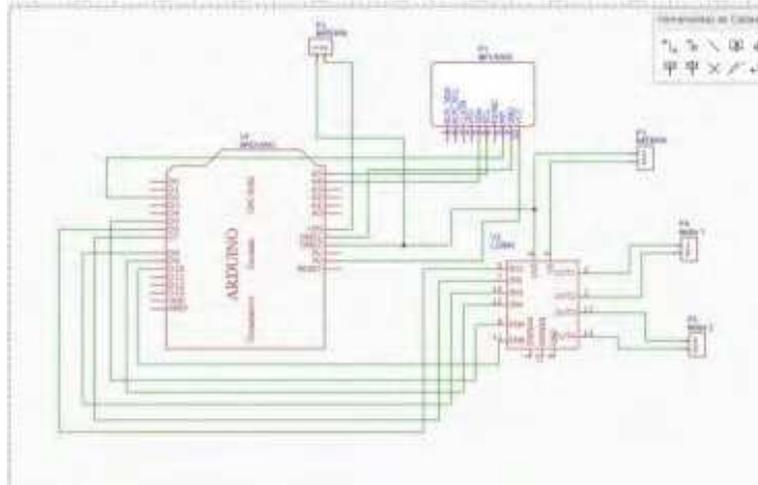


Figura 8. Diagrama de conexión de los elementos del segundo robot balancín.



Figura 9. Prototipo del segundo robot balancín.

Ambos prototipos utilizan el módulo MPU6050, el cual cuenta con un acelerómetro y un giroscopio. Mediante este dispositivo se toman las lecturas de inclinación en grados dentro del código de programación del microcontrolador. El acelerómetro del módulo mide la aceleración, por lo que se requiere transformar esas lecturas a grados. Para obtener los ángulos de aceleración del eje x y del eje y, se utilizan las fórmulas (1) y (2).

$$\theta_{AcelY} = \text{atan}\left(\frac{x}{\sqrt{y^2 + z^2}}\right) \quad (2)$$

$$\theta_{AcelX} = \text{atan}\left(\frac{y}{\sqrt{x^2 + z^2}}\right) \quad (2)$$



El módulo mide también la velocidad angular mediante el giroscopio, y si conocemos el ángulo inicial, es posible añadir el valor que indica el giroscopio para conocer el ángulo a cada momento. Esto se representa en el código mediante las ecuaciones (3) y (4).

$$\theta_y = \text{anguloYanterior} + \text{datoY} - \text{tiempotranscurrido} + \theta_{AcelY} \quad (3)$$

$$\theta_x = \text{anguloXanterior} + \text{datoX} - \text{tiempotranscurrido} + \theta_{AcelX} \quad (4)$$

La variable “tiempotranscurrido” es el tiempo que pasa cada vez que se calcula la fórmula. Las variables “anguloYanterior” y “anguloXanterior” guardan el ángulo de la aceleración calculado en la iteración anterior, y las variables “datoY” y “datoX” son las lecturas del giroscopio. Para obtener una lectura más precisa y libre de perturbaciones no deseadas, se aplica un filtro digital a las ecuaciones (3) y (4), sustituyendo además las ecuaciones (1) y (2), quedando como resultado las ecuaciones (5) y (6).

$$\theta_{TotalY} = 0.98(\theta_{TotalY} + \text{datoY} - \text{tiempotranscurrido}) + 0.2\text{atan}\left(\frac{x}{\sqrt{y^2 + z^2}}\right) \quad (5)$$

$$\theta_{TotalX} = 0.98(\theta_{TotalX} + \text{datoX} - \text{tiempotranscurrido}) + 0.2\text{atan}\left(\frac{y}{\sqrt{x^2 + z^2}}\right) \quad (6)$$

Es importante recalcar que aunque anteriormente se mencionó que los valores obtenidos para las constantes mediante la teoría no siempre funcionan de la manera esperada, es necesario realizar todo el proceso de modelado de los sistemas y obtener los valores para las constantes del control, ya que es a partir de esos valores que se comenzará a realizar las pruebas en la práctica, por lo que realizar simulaciones es de gran importancia; en este caso, se tomaron datos con los prototipos ya ensamblados y en base a ellos se obtuvo la ecuación del sistema y se procede a simulación. En la Figura 10, se muestra la gráfica de respuesta del sistema general de en color azul y la gráfica del modelo ajustado a las características físicas de los prototipos en color rojo.

Una vez obtenido el modelo ajustado se generó una simulación mediante Matlab, en donde se aprecia el comportamiento del sistema sin un control aplicado y con el control aplicado, esto se puede observar en la Figura 11, donde la gráfica azul representa al sistema sin control aplicado, mientras la gráfica de color azul claro representa al sistema con el control aplicado, como se puede observar, cuando el sistema no cuenta con ningún tipo de control, este logra llegar a estabilizarse, sin embargo, no alcanza el valor deseado, por lo que no funcionara de la manera en que se desea, al introducir el control PID, este mejora la respuesta del sistema, permitiendo que responda como se desea, de esta manera el sistema llega al valor deseado y se se mantiene ahí.

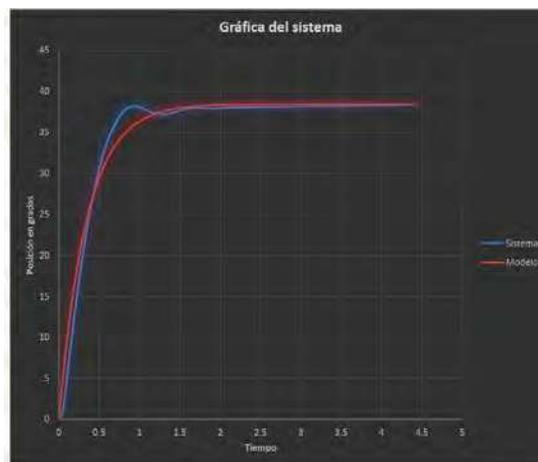


Figura 10.- Gráfica del sistema y del modelado obtenido.

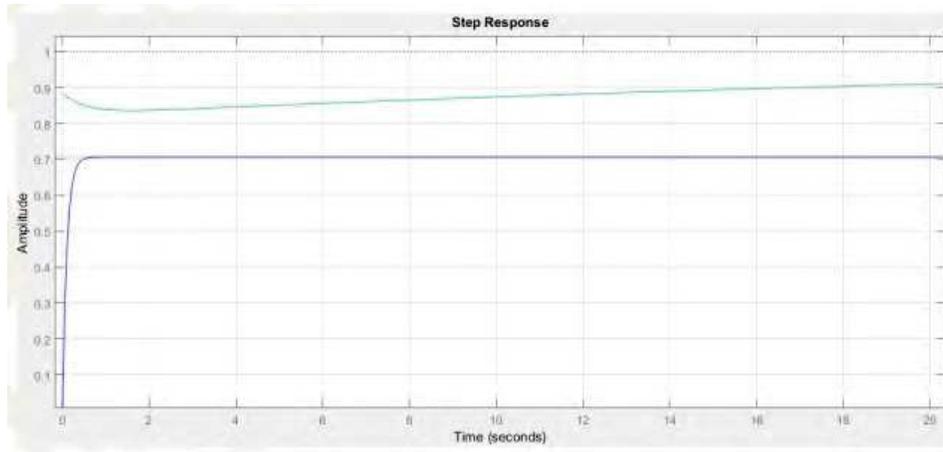


Figura 11.- Simulación del sistema.

5. Resultados

Los prototipos elaborados cumplen con el objetivo deseado al lograr una eficaz demostración de cómo afectan los valores de las constantes K_p , K_i y K_d que se les ingresen. En el caso del robot balancín, su objetivo es mantener la posición vertical a pesar de fuerzas externas que afecten su equilibrio, por lo que su setpoint es fijo. En la Figura 12 puede observarse una prueba realizada para observar su estabilidad. Variando las constantes se obtienen distintos resultados que además pueden ser visualizados de manera gráfica mediante la función de Serial Plotter del IDE de Arduino, como se observa en la Figura 13, con lo que se obtiene una opción extra para observar el proceso de control. Los valores que se ingresen a los prototipos afectan a la acción de control, siendo nula o efectiva, dependiendo de su valor, pueden estabilizar correcta y rápidamente el sistema, realizando una estabilización después de muchas oscilaciones, o no logrando un control esperado.



Figura 12.- Prueba del primer robot balancín.



Figura 13.- Gráfica obtenida durante la acción de control del robot balancín.

En el caso del prototipo con los motores brushless, además de variar las constantes de la acción de control, es posible cambiar el set point del dispositivo para que se logre una estabilización en ángulos inclinados, y de esta manera apreciar de una manera un poco distinta la acción de control que se lleva a cabo, esto se puede observar en la Figura 16.

De manera similar al anterior prototipo, mediante el Serial Plotter del IDE Arduino, puede observarse en tiempo real la gráfica resultante del ángulo de inclinación, con la cual se puede trabajar para analizar tiempos de respuesta, entre otros detalles. En la Figura 14 se observa el prototipo estabilizado en una posición horizontal, y en la Figura 15 se observa la gráfica correspondiente a esa estabilización.

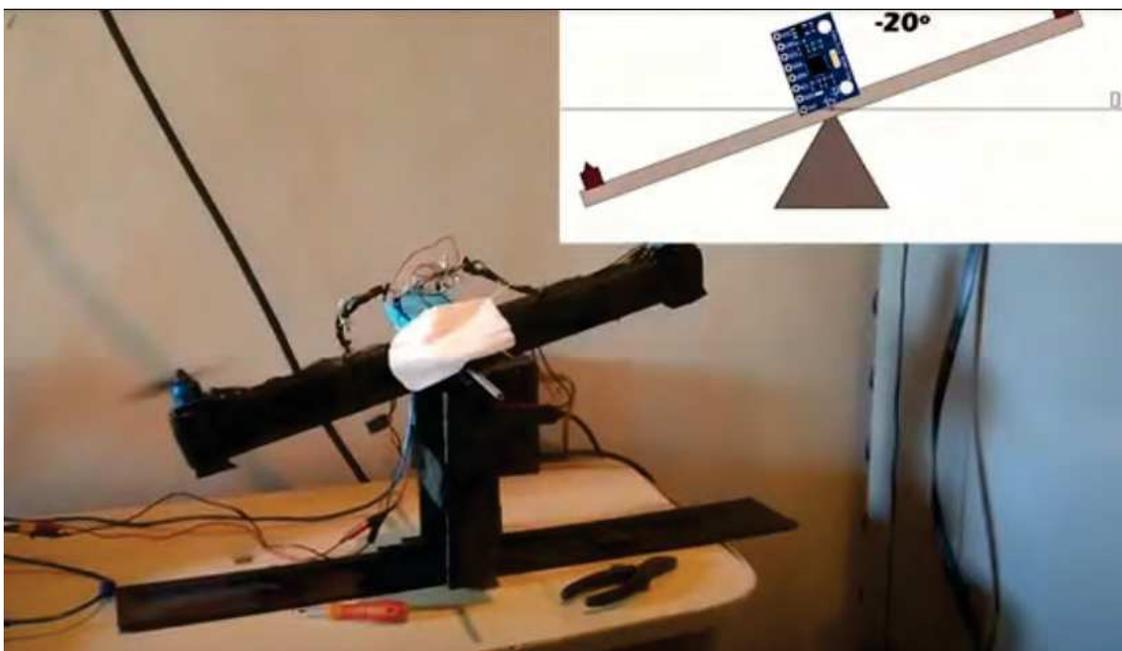


Figura 14.- Prototipo de balancín con motores brushless, equivalente a medio eje de un dron.

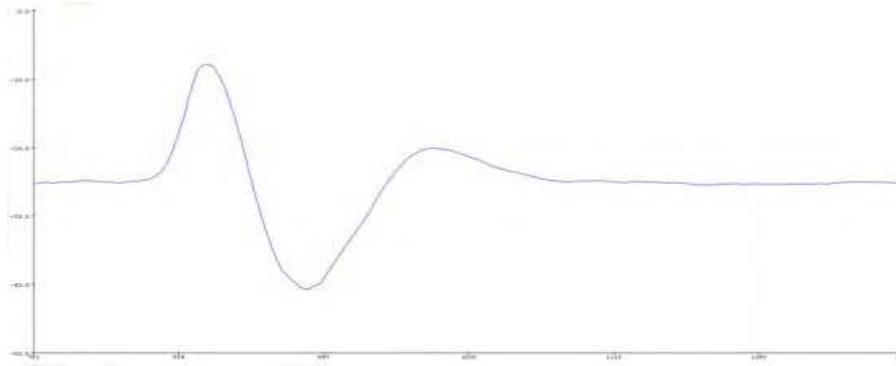


Figura 15.- Gráfica obtenida del ángulo de inclinación durante la estabilización del prototipo.

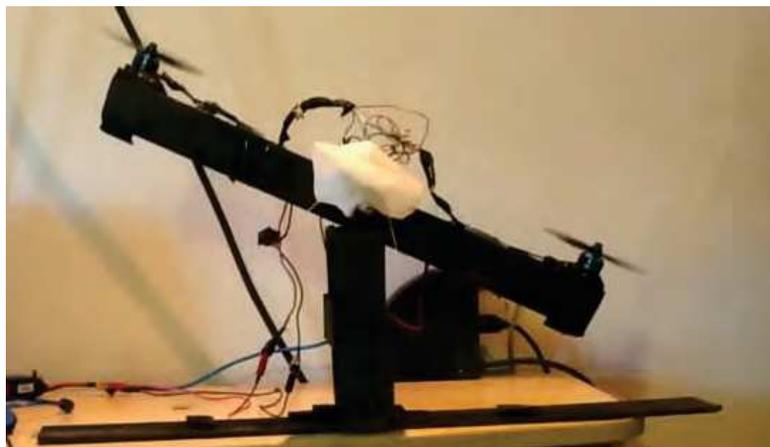


Figura 16.- Set Point establecido en -25° .

Algo importante a tener en cuenta es que el Set Point que se desee establecer debe estar en -45° y 45° , esto debido a la limitación física que representa el soporte del prototipo, sin embargo, si se deseara contar con más rango de movimiento siempre es posible incrementar la altura de la base del prototipo.

6. Conclusiones

Realizando este proyecto podemos primeramente mencionar que de un sistema puede obtenerse el modelo matemático, y mediante este puede obtenerse su función de transferencia, lo cual es muy importante debido a que proporciona la facilidad de poder simular el sistema implementado, observando varios detalles como que el sistema sea inestable por naturaleza, o que sea necesario agregar un controlador PID para la estabilidad del sistema. También puede notarse que mediante la simulación es más fácil encontrar el valor de las constantes K_p , K_i y K_d . De manera complementaria, estos prototipos pueden ser utilizados como una herramienta de aprendizaje aunada a las simulaciones realizadas, con el fin de observar tanto de manera teórica como práctica el funcionamiento en tiempo real del control PID.

Otro detalle a tener en cuenta es que en las simulaciones el sistema puede alcanzar completamente la estabilidad. En cambio, en las pruebas físicas no siempre es así, debido a que existen elementos externos que generan diferencias mínimas en las simulaciones y en las pruebas reales empleadas, por lo cual podemos concluir el modelar y simular el sistema que se quiera controlar nos puede ayudar bastante para ver la respuesta que este tendrá y nos permitirá tener un panorama más



amplio a la hora de implementar controladores que garanticen su estabilidad. Mediante la implementación de estos prototipos se vuelve posible apreciar estas diferencias entre lo teórico y lo práctico, que sirven para generar experiencia y en futuros proyectos tomar dichos detalles en cuenta al momento de realizar el control del sistema deseado.

Referencias

- [1] Energytrend. (14 de Enero de 2020). Wikipedia. Obtenido de https://es.wikipedia.org/wiki/Bater%C3%ADa_de_pol%C3%ADmero_de_litio
- [2] Pozo, E. G. (2005). Técnicas para la Implementación de la Robótica en la Educación Primaria. Recuperado el 10 de mayo de 2011, de Complubot: http://complubot.educa.madrid.org/actividades/inrerdidac_robotica_primaria.pdf.
- [3] Geekfactory. (14 de Noviembre de 2016). Geekfactory. Obtenido de <https://www.geekfactory.mx/tienda/motores-y-controladores/controlador-de-velocidadmotor-brushless/>
- [4] Master Ingenieros. (5 de Junio de 2015). Master Ingenieros. Obtenido de <http://masteringenieros.com/wp-content/uploads/pdf/MI50N/-www-adsnt-recursosmasteringenieros-file-motor.pdf>
- [5] Ogata, K. (2010). Ingeniería de control moderna. Madrid: Pearson.
- [6] Orlando. (25 de Abril de 2014). HETPRO. Obtenido de <https://hetprostore.com/TUTORIALES/modulo-acelerometro-y-giroscopio-mpu6050-i2c-twi/>
- [7] Vazquez, A. (24 de Febrero de 2012). Rice Propulsion. Obtenido de http://www.ricepropulsion.com/TNLS/historia_de_la_propela.htm
- [8] H. Pilamala, «Robot Balancing Basico,» Enero 2019. [En línea]. Available: https://www.researchgate.net/publication/331972800_Robot_Balancing_Basico. [Último acceso: 05 Marzo 2020].
- [9] S. J. Pincin, «Diseño y construcción de un robot balancin,» Junio 2016. [En línea]. Available: <http://ricabib.cab.cnea.gov.ar/557/1/1Pincin.pdf>. [Último acceso: 05 Marzo 2020].
- [10] S. T. Ausejo, «Diseño y construcción de un robot auto-balanceado mediante Arduino,» 19 Junio 2017. [En línea]. [Último acceso: 02 Mayo 2020].



Nuevo modelo de fricción para robots manipuladores basado en funciones hiperbólicas

Rodríguez Hernández Ernesto Daniel, Sánchez García Brandon Erick, Reyes Cortés Fernando, Oviedo Barriga José Luis, Vargas Treviño María Aurora.

Facultad de Ciencias de la Electrónica, Opción en Automatización
Benemérita Universidad Autónoma de Puebla
Maestría en Ciencias de la Electrónica.
Dirección: 18 sur y Av. San Claudio. Col. San Manuel C.U.
Tel. 2222295500 ext. 7410
e-mail: d_derneasto@hotmail.com

Resumen

En este artículo se presenta una propuesta de modelo de fricción enfocado a robots manipuladores de transmisión directa basado en funciones continuas que pretende emular el comportamiento de modelos dinámicos de fricción tales como el modelo de Dahl o LuGre, incluyendo aspectos relevantes de los modelos dinámicos tales como el efecto Stribeck, conservando una estructura estática que simplifica su implementación en algoritmos de control con compensación de fricción, que además puede ser competitivo con modelos de fricción dinámicos modernos. La propuesta aquí presentada incluye un análisis cualitativo del modelo, la características estáticas y dinámicas que representa, así como la sintonización de parámetros constantes de fricción, además de su respectivo análisis experimental en un robot manipulador antropomórfico de 3GDL experimental de transmisión directa.

Palabras clave: Fricción, modelo, robot, LuGre, Coulomb, Stribeck, viscosa, histéresis.

1. Introducción

Los robots manipuladores se han convertido en una parte importante de la industria al mejorar las técnicas de producción realizando tareas con alta precisión y bajo costo, tales como manipulación de objetos, soldadura, pintura, entre otras aplicaciones. Los avances que se han logrado en la robótica se deben en gran medida a la implementación del conocimiento de múltiples áreas. En el control de robots manipuladores, es fundamental conocer a fondo el modelo dinámico del robot, así como contar con un control robusto y de alto desempeño, lo que incluye al fenómeno de fricción, y pares gravitacionales.

El control de robots manipuladores se ha convertido en un pilar fundamental de la industria, por lo que es necesario contar un modelo de que represente la dinámica de los sistemas robóticos de manera completa y precisa, favoreciendo así el diseño de estructuras de control robustas y precisas. Uno de los elementos presentes en el modelo dinámico de un robot manipulador son los pares de fricción provocados por el rozamiento de las superficies de los servoactuadores y piezas móviles. Estos pares de fricción moldean la respuesta del sistema generando un efecto disipativo en los servoactuadores alterando la respuesta final del robot. Los avances generados en la rama de la tribología han dado paso a nuevos modelos de fricción basados en la interacción de superficies a nivel microscópico, dando paso a su vez a algoritmos de control de robots manipuladores con compensación de fricción eliminando los efectos negativos de este fenómeno.



El fenómeno de fricción se encuentra en todos aquellos sistemas que involucran partes en movimiento. Sus efectos han sido ampliamente estudiados desde su descubrimiento en 1493 por Leonardo Da Vinci con sus estudios del movimiento de bloques sobre superficies [1] [2], los cuales representaron el comienzo de la ciencia moderna de la tribología. Los avances en el estudio de la fricción han permitido un mejor entendimiento de este fenómeno y sus efectos en el desempeño de sistemas robóticos, en los cuales se produce un efecto disipativo que moldea la respuesta transitoria, generando un amortiguamiento natural en los actuadores.

Comúnmente, la fricción en robots manipuladores es representada por una componente de fricción de Coulomb, y una componente de fricción viscosa. La fricción de Coulomb, con base en la ley de fricción de Coulomb, en la cual se estipula que la fuerza de fricción es contraria en dirección al movimiento de un objeto sobre una superficie, y proporcional a la fuerza normal del mismo. La fricción viscosa incluida en modelos clásicos de fricción para robots manipuladores representa otra característica macroscópica importante del fenómeno de fricción, la cual es que la fuerza de fricción aumenta de manera proporcional a la velocidad de movimiento entre superficies. Estos 2 elementos han resultado suficientes para representar el comportamiento esperado de la fuerza de fricción sobre los actuadores de un robot, así como estimar el torque necesario para compensar el efecto de amortiguamiento ocasionado por el rozamiento, sin embargo, gracias a los avances en instrumentación, y los estudios en tribología, ahora se comprende con mayor profundidad el comportamiento de la fricción a niveles microscópicos [3].

Mediante una serie de experimentos en los que se observó la dependencia de la velocidad en los rodamientos de bolas, Stribeck estipuló que la fricción decrece a medida que la velocidad aumenta en un rango de velocidades específico. Estas variaciones, no contempladas en los modelos de fricción basados en observaciones macroscópicas, producen un cambio significativo en los modelos de fricción que sí consideran el efecto Stribeck. El efecto Stribeck aparece en las regiones de baja velocidad y es producido por la separación entre las uniones microscópicas creadas entre las asperezas de las superficies en contacto.

Uno de los modelos de fricción más completos que existen es el modelo de LuGre [4] (denominado así por la colaboración entre las universidades de Lund y Grenoble), el cual es del tipo dinámico debido a la naturaleza dinámica del fenómeno de fricción [5]. Este modelo posee una estructura similar a los modelos clásicos de fricción, con la adición de la curva de efecto Stribeck, así como una región de histéresis. La principal desventaja de este modelo y de todos los modelos de fricción dinámicos, es que su implementación incrementa el número de estados del sistema, lo que podría interpretarse como un incremento de los grados de libertad del robot.

Considerando lo anterior se propone que es posible diseñar un modelo de fricción que represente de manera apropiada el fenómeno de fricción en robots manipuladores, incluyendo aspectos de los modelos de fricción dinámicos como el modelo de LuGre, sin incrementar el número de variables de estado del sistema.

2. Diseño y análisis del modelo de fricción

2.1 Fricción en robots manipuladores

El modelo dinámico de un robot manipulador proporciona información general entre los pares aplicados a los servomotores y el movimiento de sus articulaciones, y es un aspecto central del estudio de sistemas robóticos sin importar su estructura o geometría. Por lo tanto, la aportación de los fenómenos que puedan producir cambios en el sistema debe ser considerados. El modelo dinámico de un robot de n grados de libertad está dado por la ecuación (1).

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + f_f(\dot{q}), \quad (1)$$



donde:

- $\mathbf{q} \in \mathbb{R}^n$ es el vector de coordenadas articulares,
- $\dot{\mathbf{q}} \in \mathbb{R}^n$ es el vector de velocidades articulares,
- $\ddot{\mathbf{q}} \in \mathbb{R}^n$ es el vector de aceleraciones articulares,
- $M(\mathbf{q}) \in \mathbb{R}^{n \times n}$ es la matriz de inercia
- $C(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ es la matriz de fuerzas centrípetas y de Coriolis,
- $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ es el vector de pares gravitacionales,
- $\mathbf{f}_f(\dot{\mathbf{q}}) \in \mathbb{R}^n$ es el vector de pares de fricción.
- $\boldsymbol{\tau} \in \mathbb{R}^n$ es el vector de pares aplicados.

Tal como se mencionó anteriormente, existen diferentes modelos de fricción que pueden ser utilizados en el modelo dinámico de un robot manipulador; para el caso general, los modelos estáticos son los más sencillos de implementar y relacionan directamente el movimiento de las articulaciones con los pares de fricción generados por el rozamiento de los servoactuadores del robot.

1.1 Modelo clásico de fricción

El modelo clásico encontrado frecuentemente en la literatura de control de robots manipuladores posee 2 componentes fundamentales y es una función únicamente dependiente de la velocidad, por lo que se considera un modelo estático, el cual está representado por la ecuación (2).

$$\boldsymbol{\tau}_f = \mathbf{b}\dot{\mathbf{q}} + \mathbf{f}_c \text{signo}(\dot{\mathbf{q}}), \quad (2)$$

donde:

- $\boldsymbol{\tau}_f \in \mathbb{R}^n$ es el vector de pares de fricción,
- $\mathbf{b} \in \mathbb{R}^n$ es el vector de coeficientes de fricción viscosa,
- $\mathbf{f}_c \in \mathbb{R}^n$ es el vector de coeficientes de fricción de Coulomb.

En el modelo de fricción clásico se obtiene una aproximación general del torque disipado por el rozamiento. La fricción viscosa considerada en este modelo es representada como una pendiente cuya magnitud depende de la velocidad de desplazamiento de las superficies bajo rozamiento (en este caso, los servoactuadores y algunos otros elementos móviles del robot). Un aspecto significativo del modelo clásico es su estructura discontinua debido a la presencia de la función signo, la cual se utiliza para representar la ley de fricción de coulomb, estableciendo una disipación de torque de magnitud constante y dependiente del sentido de giro del motor. La adecuada selección de los parámetros de fricción viscosa y fricción de Coulomb permiten una aproximación de la fricción en el robot y a su vez, hace posible compensar dicha pérdida de energía mediante un control con este modelo.

La discontinuidad de la componente de fricción de Coulomb es una característica necesaria para la correcta representación de la fricción estática del robot, que se produce cuando el robot se encuentra en una posición fija por un tiempo determinado. Por otra parte, la discontinuidad presente en la ley de fricción de Coulomb es también un aspecto no favorable para el análisis matemático de aspectos importantes del sistema, tales como el estudio de estabilidad de un controlador con compensación de fricción. Lo anterior, aunado a la falta de aspectos microscópicos del fenómeno de fricción, tal como la curva de efecto Stribeck, nos lleva a la conclusión de que es posible mejorar el modelo clásico de fricción para ajustarlo de mejor manera a el comportamiento real del fenómeno de fricción.

3. Propuesta de un modelo de fricción

Partiendo del comportamiento del fenómeno de fricción observado en servoactuadores, se diseñó un modelo de fricción a partir de la adecuada combinación de funciones hiperbólicas, expresando matemáticamente la fricción como una función estática dependiente de la velocidad, que a su vez es una función continua. El modelo de fricción propuesto en el siguiente:



$$\tau_f = f_c \alpha \frac{\tanh^{2h-1}(\delta \dot{q})}{1 + \operatorname{atan}^{2s}(\delta \dot{q})} + b \dot{q}, \quad (3)$$

donde:

- $\alpha \in \mathbb{R}_+$ es el coeficiente de normalización,
- $\delta \in \mathbb{R}_+$ es el coeficiente de disipatividad,
- $h \in \mathbb{N}_+ > 0$ es el coeficiente de histéresis,
- $s \in \mathbb{R}_+$ es el coeficiente de efecto Stribeck.

La ecuación (3), que representa el modelo de fricción propuesto, es claramente un modelo del tipo estático, y contiene componentes de los modelos dinámicos y estáticos. La estructura matemática de este modelo de fricción facilita su implementación en un controlador con compensación de fricción de manera similar a la compensación de gravedad.

El coeficiente de normalización α es una constante de diseño que ajusta la componente de fricción de Coulomb en estado estable en 1 y -1 para los giros positivos y negativos del servomotor, este parámetro debe ajustarse con respecto al comportamiento general del modelo y ajustar la componente de fricción de Coulomb. La aportación del coeficiente de normalización se ejemplifica en las figuras 1 y 2, donde se puede apreciar que la selección adecuada del coeficiente de normalización ajusta los límites del componente de fricción de Coulomb (de 0.5Nm a 1.0Nm).

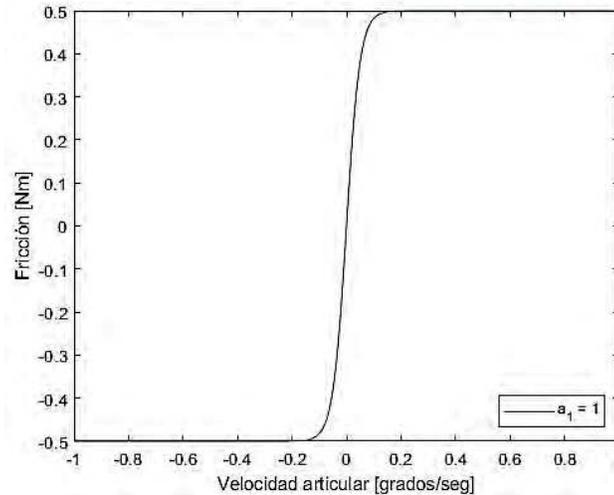


Figura 1. Modelo de fricción con variaciones en coeficiente de normalización, a) $s_1 = 0$, $h_1 = 1$, $a_1 = 1$, $b_1 = 0$, $d_1 = 20$, $f_{c1} = 1$;

El coeficiente disipatividad reduce la pendiente en la región de transición entre el sentido de giro del servomotor, es seleccionado de acuerdo con el comportamiento deseado de la fricción; este elemento del modelo de fricción es esencial para conservar la estructura continua de la función sin emplear funciones discontinuas tales como la función signo. La aportación del coeficiente de disipatividad se ejemplifica en la figura 3 donde se observan los cambios en la pendiente de fricción viscosa a medida que se ajusta el coeficiente de disipatividad.

Los aspectos más relevantes de este modelo son aquellos relacionados con aspectos microscópicos del fenómeno de fricción, tales como la histéresis y la curva de efecto Stribeck. El coeficiente de histéresis modifica el comportamiento del modelo en la región de baja velocidad del robot, en la que el rozamiento de micro asperezas y las vibraciones de los servomotores generan una región



de un grosor determinado por la interacción de estos elementos; con este parámetro de diseño se modifica el grosor de la región de baja velocidad.

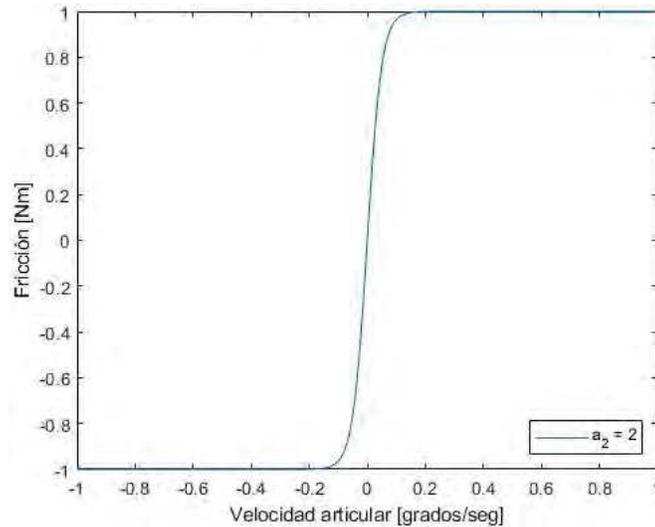


Figura 2. Modelo de fricción con variaciones en coeficiente de normalización, $s_2 = 0$, $h_2 = 1$, $a_2 = 2$, $b_2 = 0$, $d_2 = 20$, $fc_2 = 1$.

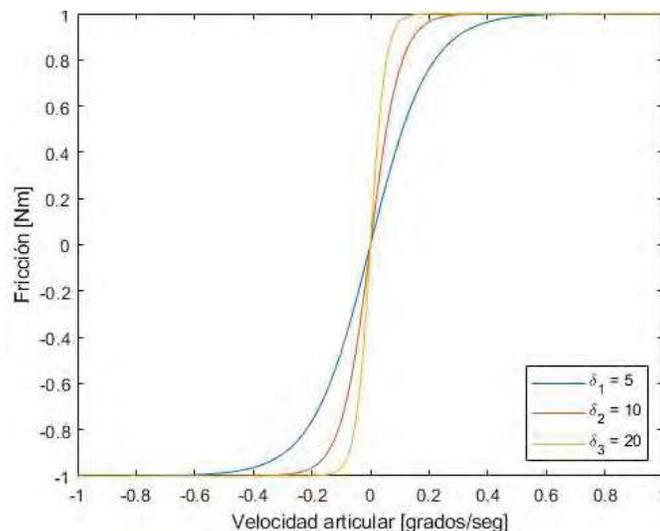


Figura 3. Modelo de fricción con variaciones en coeficiente de disipatividad, a) $s_1 = 0$, $h_1 = 1$, $a_1 = 2$, $b_1 = 0$, $d_1 = 5$, $fc_1 = 1$; b) $s_2 = 0$, $h_2 = 1$, $a_2 = 2$, $b_2 = 0$, $d_2 = 10$, $fc_2 = 1$; c) $s_3 = 0$, $h_3 = 1$, $a_3 = 2$, $b_3 = 0$, $d_3 = 20$, $fc_3 = 1$.

El aporte del coeficiente de histéresis al modelo de fricción se muestra en la figura 4. El coeficiente de histéresis pretende principalmente generar una separación de grosor variable en la región de baja velocidad del servoactuador. En un modelo dinámico, la histéresis se produce por variaciones en los pares de fricción, los cuales son dependientes de los valores anteriores de las variables de estado del sistema. En este modelo, este comportamiento se emula al permitir generar una región de forma y tamaño similar a la histéresis dinámica y del mismo modo, puede observarse en casos de estudio con robots experimentales. La histéresis artificial producida en este modelo cuenta con una pendiente idéntica a la encontrada en la componente de fricción viscosa, esta relación pierde relevancia en



aplicaciones reales de compensación donde se requiera de una respuesta inmediata del modelo cuando el robot parte del reposo, sin embargo, puede ser usado como un freno artificial para evitar el movimiento libre producido por perturbaciones no deseadas que se encuentren dentro de un determinado rango.

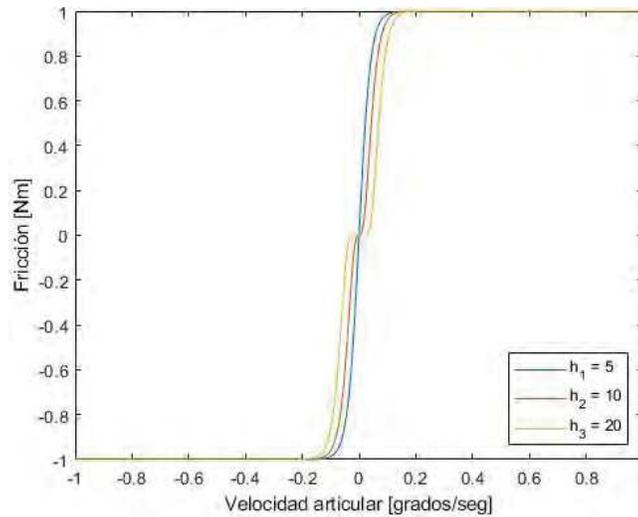


Figura 4. Modelo de fricción con variaciones en coeficiente de histéresis, a) $s_1 = 0$, $h_1 = 1$, $a_1 = 2$, $b_1 = 0$, $d_1 = 25$, $fc_1 = 1$; b) $s_2 = 0$, $h_2 = 2$, $a_2 = 2$, $b_2 = 0$, $d_2 = 25$, $fc_2 = 1$; c) $s_3 = 0$, $h_3 = 6$, $a_3 = 2$, $b_3 = 0$, $d_3 = 25$, $fc_3 = 1$.

Por otra parte, el coeficiente de efecto Stribeck moldea la forma de la curva de efecto Stribeck, este es también un parámetro de diseño que se selecciona de acuerdo con el comportamiento esperado de la fricción en el robot. La selección de los parámetros de histéresis y Stribeck, pueden emplearse de manera análoga al comportamiento de modelos dinámicos si es que se conoce el comportamiento de estos elementos en el sistema. La aportación del coeficiente de Stribeck al modelo de fricción se muestra en la figura 4.

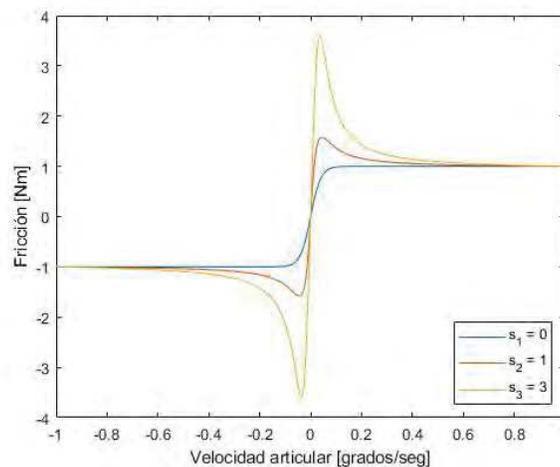


Figura 5. Modelo de fricción con variaciones en coeficiente de Stribeck, a) $s_1 = 0$, $h_1 = 1$, $a_1 = 2$, $b_1 = 0$, $d_1 = 25$, $fc_1 = 1$; b) $s_2 = 1$, $h_2 = 1$, $a_2 = 3.34$, $b_2 = 0$, $d_2 = 25$, $fc_2 = 1$; c) $s_3 = 3$, $h_3 = 1$, $a_3 = 6.49$, $b_3 = 0$, $d_3 = 25$, $fc_3 = 1$.

En este modelo, se incorporan en una misma componente elementos microscópicos y macroscópicos de la fricción, por lo que la sintonización de ganancias requiere que se reajusten los parámetros repetidamente y se consideren posibles alteraciones inesperadas en la respuesta



ocasionadas por variaciones en algún parámetro sensible. El componente de fricción viscosa se implementa de la misma forma que en modelos estáticos y dinámicos, y su aportación al modelo de fricción se ejemplifica en la figura 5.

La combinación de los elementos incluidos en el modelo da paso a un amplio rango de diseño en cuanto al comportamiento esperado de la fricción en un robot manipulador. Otra ventaja notable en el modelo de fricción es la capacidad de incluir directamente elementos del modelo de fricción clásico, tal como los coeficientes de fricción viscosa y de Coulomb. Uno de los parámetros que se encuentra mayormente limitado en esta versión del modelo de fricción es el coeficiente de Stribeck, ya que solo puede representarse con números positivos y es muy sensible a cambios, tal como se ilustra en la figura 5, sin embargo, los errores provocados por el coeficiente de Stribeck pueden ser compensados mediante alguna variable auxiliar, tal como el coeficiente de disipación o el coeficiente de normalización. La combinación de los coeficientes de Stribeck e histéresis puede diseñarse priorizando el comportamiento deseado de cualquiera de los 2 elementos y posteriormente ajustarse mediante el coeficiente de normalización.

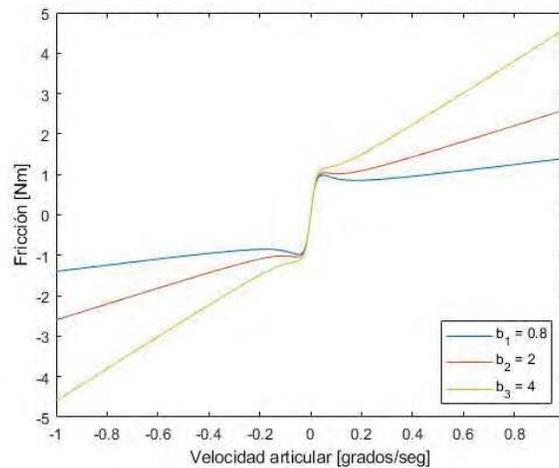


Figura 6. Modelo de fricción con variaciones en coeficiente de fricción viscosa. a) $s_1 = 1$, $h_1 = 1$, $a_1 = 2$, $b_1 = 0.8$, $d_1 = 25$, $fc_1 = 1$; b) $s_2 = 1$, $h_2 = 1$, $a_2 = 3.34$, $b_2 = 2$, $d_2 = 25$, $fc_2 = 1$; c) $s_3 = 1$, $h_3 = 1$, $a_3 = 6.49$, $b_3 = 4$, $d_3 = 25$, $fc_3 = 1$.

En el coeficiente de Stribeck, se representa una pendiente negativa similar a la fricción viscosa. Este fenómeno es frecuentemente ignorado en modelos estáticos, por lo que su inclusión en este modelo representa una ventaja con respecto a estos modelos. Otra característica presente en el fenómeno de fricción comúnmente ausente en los modelos de tanto dinámicos como estáticos es la asimetría. La asimetría en el fenómeno de fricción produce pares de fricción de magnitudes diferentes cuando se tiene una dirección de giro u otra. Esta desigualdad propicia un error en la estimación de los pares de fricción, sin embargo, asumiendo que se conozca la diferencia promedio entre los pares de fricción de un robot en particular, es posible implementar dicha diferencia entre los pares de fricción positivos y negativos con el modelo de fricción aquí presentado mediante la elección de valores fraccionarios para los parámetros de efecto Stribeck e histéresis; de esta forma, el par de fricción negativo modifica su forma con respecto al par de fricción positivo, el cual mantiene su estructura.

Es evidente que la implementación de este modelo de fricción en el control de un robot manipulador puede realizarse de manera similar a cualquier otro modelo estático sin pérdida de generalidad, lo que supone una gran ventaja para su uso en sistemas robóticos que ya cuenten con compensación de fricción mediante modelos de fricción estáticos.

2. Resultados experimentales

2.1 Robot experimental

Con el fin de validar el desempeño del modelo de fricción propuesto en este artículo, se realizaron una serie de experimentos con un robot de investigación de 3 grados de libertad en el laboratorio de robótica y control de la Benemérita Universidad Autónoma de Puebla (figura 6). El robot seleccionado es del tipo antropomórfico y posee servoactuadores de transmisión directa sin escobillas, por lo que el torque proporcionado no se ve afectado por factores externos al fenómeno de fricción tales como el juego mecánico propio de los sistemas con reductores. De este modo, es posible medir de forma acertada el torque disipado por el rozamiento de superficies en las articulaciones del robot.

El robot además cuenta con encoders de hasta 1024000 pulsos/rev, que permiten la adquisición de datos relevantes del fenómeno de fricción en los servoactuadores. La tabla 1 muestra las características de los servoactuadores del robot experimental.

Tabla 1. Servoactuadores del robot experimental.

Eslabón	Modelo	Torque [Nm]	Encoder [p/rev]
Base	DM1050A	50	1024000
Hombro	DM1150A	150	1024000
Codo	DM1015B	15	655360



Figura 7. Robot antropomórfico experimental de la Benemérita Universidad Autónoma de Puebla

2.2 Resultados experimentales

La validación experimental del modelo de fricción aquí propuesto se realizó mediante el análisis del movimiento en el robot manipulador de 3 grados de libertad ante una entrada de pares aplicados en direcciones tanto positivas como negativas en un tiempo de 19.174 segundos, en los cuales se recopilaron 7671 datos discretos. Como resultado, el robot manipulador experimental de desplazaba por su espacio de trabajo cambiando la dirección de giro de sus servoactuadores, registrando los parámetros de posición, velocidad, aceleración y pares aplicados durante el progreso del experimento.



En el experimento realizado se tomaron en cuenta los parámetros físicos relacionados al fenómeno de fricción en el robot manipulador experimental previamente conocidos, mostrados en la tabla 2. Una vez obtenidos los datos del experimento, estos se analizaron empleando el software MATLAB, generando los pares de fricción del robot manipulador producidos por el modelo de fricción propuesto correspondiente a cada lectura discreta de velocidad articular proporcionada por el encoder.

Los pares aplicados introducidos a los servomotores del robot se calcularon con la estructura matemática mostrada en la ecuación 4, conservando las características anteriormente mencionadas y considerando el par máximo soportado por cada servomotor.

$$\tau = \beta \sin (0.2t) \quad (4)$$

donde:

- t es el tiempo en segundos desde que el robot fue encendido.
- β es el vector de pares máximos soportados por cada uno de los servoactuadores del robot.

Tabla 2. Parámetros físicos del robot manipulador.

Parámetro	Notación	Valor
Coeficiente de fricción de Coulomb	f_{c1}	5.17 Nm
	f_{c1}	1.90 Nm
	f_{c1}	1.73 Nm
Coeficiente de fricción viscosa	b_1	2.28 (Nmseg ²)/grad
	b_2	1.20 (Nmseg ²)/grad
	b_3	0.17 (Nmseg ²)/grad
Coeficiente de disipatividad	δ_1	45
	δ_2	40
	δ_3	85
Coeficiente de efecto Stribeck	s_1	1.00
	s_2	1.00
	s_3	0.92
Coeficiente de efecto histéresis	h_1	1.00
	h_2	0.95
	h_3	0.92

A continuación, se presentan los pares de fricción obtenidos por el modelo de fricción para cada uno de los servoactuadores del robot manipulador, calculado a partir de los parámetros físicos de cada servoactuadores como se indica en la tabla 2. Los resultados esperados en este experimento son pares de fricción que posean la estructura característica del fenómeno de fricción, incluyendo algunos de los elementos relevantes no incluidos en la mayoría de los modelos estáticos (principalmente curva de efecto Stribeck y región de histéresis), de modo que su aportación pueda ser implementada posteriormente en algún controlador con compensación de fricción.

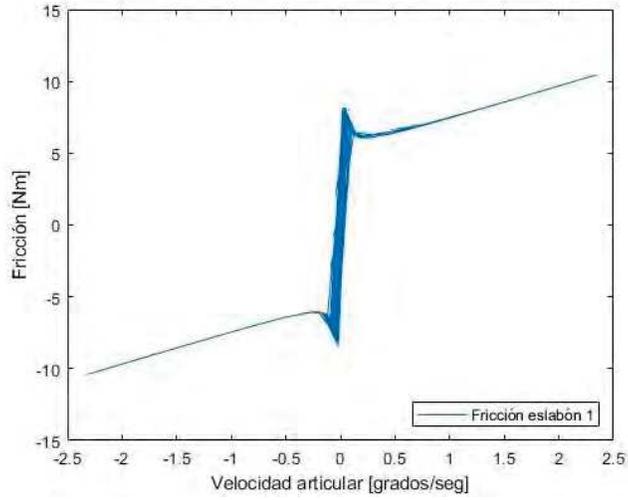


Figura 8. Pares de fricción eslabón 1 (cadera)

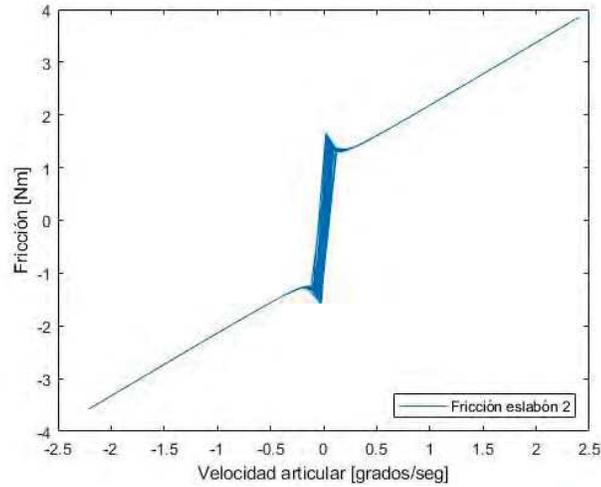


Figura 9. Pares de fricción eslabón 1 (hombro)

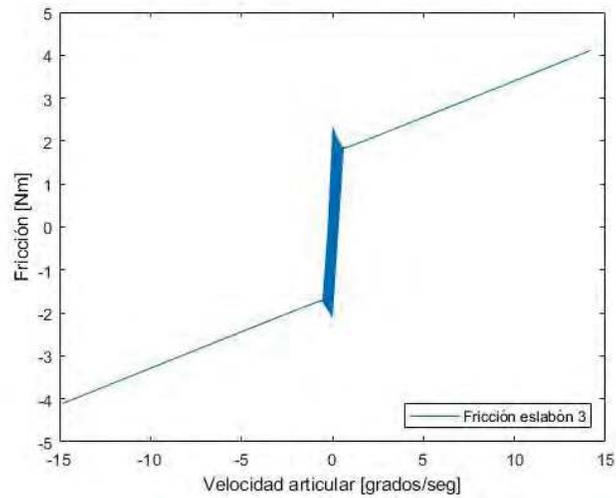


Figura 10. Pares de fricción eslabón 1 (codo)



En las figuras 7 a 9 se muestran los resultados de los pares calculados del modelo de fricción para cada una de las 3 articulaciones del robot. En la figura 7, se muestra la respuesta calculada de los pares de fricción para el servomotor de la cadera, en la figura 8, se muestran los pares de fricción modelados en el hombro, y en la figura 9 se muestran los pares de fricción en el codo del robot. Es notable que los elementos macroscópicos del fenómeno de fricción se encuentran en el modelo, de la misma forma que lo harían en cualquier otro modelo estático.

En el caso de los parámetros físicos del modelo de fricción relacionados a fenómenos microscópicos, concretamente aquellos relacionados con el efecto Stribeck y la histéresis, son en este caso, aproximaciones seleccionadas de manera arbitraria con el propósito de volver evidente su aportación al modelo de fricción. Estos elementos del fenómeno de fricción son del tipo dinámico, por lo que incluirlos en el modelo dinámico de un robot manipulador requiere normalmente de la inclusión de variables de estado dedicadas para cada uno de los servoactuadores en un robot, esto incrementa el número de variables de estado del sistema por lo menos al doble. Una interpretación distinta de la inclusión de estas variables de estado al modelo dinámico del robot manipulador para usos prácticos podría ser que se incrementa el número de grados de libertad del sistema, por lo tanto, su complejidad y a su vez, la capacidad de cómputo necesaria para controlarlo en tiempo real.

Debido a que la aportación del fenómeno de fricción resulta ser considerablemente pequeña en comparación con los pares aplicados a los servomotores en un robot, muchas veces se termina ignorando al fenómeno de fricción en el controlador del sistema, sin embargo, al implementar un modelo de fricción estático como el que se muestra en este artículo, se conserva la simplicidad del modelo dinámico del robot manipulador y se incrementa la flexibilidad en la estructura deseada del fenómeno de fricción en los servomotores del robot con respecto a los modelos estáticos clásicos de fricción.

Al comparar cualitativamente la curva característica de los modelos de fricción clásicos es notable que estos representan una desventaja importante en el análisis matemático de estabilidad de Lyapunov en controladores con compensación de fricción, pues en estos análisis es requerido que el modelo dinámico del sistema sea continuamente diferenciable, y este problema se presenta también en algunos modelos dinámicos de fricción que emplean funciones discontinuas. El uso de funciones hiperbólicas solventa este problema de análisis matemático, y conserva la funcionalidad de modelo de fricción para algoritmos de control de robots.

Las diferencias entre los pares de fricción que existen entre cada uno de los eslabones del robot se deben entre otras cosas, a la velocidad y torque máximo que es capaz de proporcionar cada servoactuador, así como el desgaste que hay en cada una de las superficies de contacto, la forma y rugosidad de las micro asperezas del material del que está fabricado el robot. La estructura del modelo de fricción está sujeta a cambios a largo plazo debido al desgaste natural de los componentes en los servoactuadores.

Otro aspecto destacable en el desempeño del modelo de fricción es la capacidad de sugerir la presencia de una región de histéresis generados por la recursividad de los datos al cambiar rápidamente entre los giros positivos y negativos de los actuadores. Esta área de alta vibración con rápidos cambios en los pares de fricción de un servoactuador es un aspecto importante de los modelos dinámicos, por lo que el uso del modelo propuesto en este artículo puede ser empleado como un recurso para representar este fenómeno en el desempeño de un robot manipulador sin necesidad de recurrir a modelos dinámicos.

3. Conclusiones

El modelo dinámico de un robot manipulador es un elemento crucial en el diseño de dispositivos y estrategias de control que cumplan con los requerimientos actuales de la industria. Uno de los elementos que produce una aportación significativa al modelo dinámico de los robots manipuladores es el fenómeno de fricción, el cual es frecuentemente simplificado o ignorado, sin embargo, contar con un modelo de fricción que represente este fenómeno de manera completa y precisa da paso a una mejora



significativa de los algoritmos de control. Una razón común para la simplificación o la no inclusión de un modelo de fricción en el control de un robot manipulador puede ser la complejidad de implementación para algunos de los modelos estáticos, así como el incremento de variables de estado del sistema por parte de los modelos dinámicos. En el modelo presentado en este trabajo se plantea una solución flexible a este problema, permitiendo sintonizar los parámetros del modelo de fricción del robot a partir del comportamiento esperado de la fricción en cada uno de los servoactuadores, así como permitir el intercambio de ciertos parámetros con los parámetros de modelos clásicos de fricción existentes. La estructura del modelo de fricción basado en funciones hiperbólicas presentado en este trabajo permite una amplia flexibilidad en el cálculo y representación de los pares de fricción en un robot manipulador, permitiendo que este se emplee de manera intuitiva, y sea fácil de implementar en algoritmos de control. Las aplicaciones de este modelo son muy variadas, principalmente se logra mejorar la precisión de los modelos dinámicos de robots manipuladores sin incrementar innecesariamente la complejidad de este, por otra parte, la compensación de fricción en robots puede emplearse en sistemas robóticos tales como exoesqueletos o simuladores, entre otras aplicaciones. El modelo de fricción aquí presentado posibilita la implementación de futuras mejoras o diseños de modelos de estructura similar, dando paso a nuevas investigaciones en el campo de la robótica relacionadas al análisis y diseño de modelos de fricción para robots manipuladores.

Referencias

- [1] 1er. Apellido y 1era. Letra del nombre de los autores. *"Título del artículo o reporte"*, Nombre de la revista. Editorial, Volumen, Número, páginas y año.
- [1] Bhushan, B., Author, and Ko, P. L., *"Introduction to Tribology"*, WILEY. USA. 2a edición, 2013.
- [2] Canudas de Wit, C., Olsson, H., *"A new model for control of systems with friction"*, IEEE Transactions on Automatic Control, IEEE, Vol. 40, pages 419-425, Mar 1995.
- [3] Sören, A., Söderberg, A., Björklund, S., *"Friction models for sliding dry, boundary and mixed lubricated contacts"*, Tribology International, vol. 40, pages 580-587, April 2007.
- [4] Olsson, H., *"Control Systems with Friction"*, Department of Automatic Control. Lund Institute of Technology (LTH). 172p, 1996.
- [5] Johansson, K., and Canudas-de-Wit, C., *"Revisiting the LuGre friction model"* in IEEE Control Systems Magazine, vol. 28, no. 6, pp. 101-114, Dec. 2008.
- [6] Piatkowski, T., *"Dahl and LuGre dynamic friction models — The analysis of selected properties. Mechanism and Machine Theory"*, Mechanism and Machine Theory, University of Technology and Life Sciences, Faculty of Mechanical Engineering, vol. 73, pp. 91-100, March 2014.
- [7] Kikuuwe, R., Takesue, N., Sano, A., Mochiyama, H., Fujimoto, H., *"Fixed-step friction simulation: from classical Coulomb model to modern continuous models"*, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, pp. 3910–3917, 2005.



Dispositivo de adquisición y Procesamiento de Señales Electromiografías

Córdova Morales Guillermo, Montiel Jiménez, Elmar, Lobato Báez Mariana, Morales Suarez Humberto, Benito Carrillo Ismael.

Instituto Tecnológico Superior de Libres
Camino Real esq. Camino Cuauhtémoc, Barrio de Tetela, 73780 Cd de Libres, Pue.
gcormora@hotmail.com

Resumen

El presente artículo describe los resultados obtenidos del desarrollo de un prototipo para el análisis de señales Electromiografías el cual permita leer señales biológicas desarrollado en la división de Ingeniería Electromecánica del Instituto Tecnológico Superior de Libres, ya que actualmente los osciloscopios actuales que están en el laboratorio de dicha institución, no tiene un buen funcionamiento para analizar este tipo de señales biológicas, es por esto que se optó por la implementación de un dispositivo con estas características, con el objetivo de poder realizar futuras investigaciones enfocadas al desarrollo de prototipos dirigidos al área de robótica, automatización e ingeniería biomédica.

Palabras clave: electromiografía, amplificador operacional, biopotenciales, interfaz por computadora, impresión 3D.

1. Introducción

Desde décadas, el ser humano ha buscado fusionar la ingeniería y la medicina, naciendo así la ingeniería biomédica, la cual ha permitido el desarrollo e innovación de tecnologías para fines médicos, como terapia o rehabilitación. Es dentro de esta disciplina que se encuentra inmerso el desarrollo de interfaces electromiografías (EMG); Según Konrad [1] una interfaz EMG es un dispositivo electrónico que hace uso de señales biológicas producidas por los músculos del cuerpo, las cuales son procesadas y analizadas por un circuito electrónico y un software. Hoy en día el desarrollo de estos dispositivos va en ascenso, a pesar de la complejidad que conlleva trabajar con este tipo de señales, su uso ya no solo se centra en el desarrollo de aplicaciones terapéuticas y de rehabilitación como el caso de las prótesis mioeléctricas, sino también, para el control de robots y brazos robóticos, países como Estados Unidos, Japón, son los que llevan la vanguardia en el desarrollo de esta tecnología. En México se cuenta con universidades e institutos que han realizado estudios en el tema, como, por ejemplo:

El Instituto Politécnico Nacional con la tesis titulada “**Implementación de señales mioeléctricas en un sistema de control para generar movimientos en una prótesis personalizada de mano**”. En este trabajo de investigación se presenta el diseño de control capaz de recibir y procesar las señales mioeléctricas generadas por el antebrazo de una persona, con la finalidad de controlar una prótesis personalizada de mano. [2]

La Universidad Autónoma de Ciudad Juárez con la investigación llamada “**Sistema de adquisición y procesamiento de las señales mioeléctricas**”. El principio de este proyecto es adquirir y controlar las señales mioeléctricas añadiendo la comunicación inalámbrica para que puedan ser utilizadas en la implementación de alguna aplicación libre de cables. En este proyecto se propone como ejemplo de aplicación el prototipo de un pedaleador de terapia inalámbrico, el cual es controlado por medio de las señales mioeléctricas del bíceps braquial que es donde se colocan electrodos de cloruro



de plata para obtener un potencial de acción generado al contraer el músculo cuando se dobla el brazo. Este potencial genera la señal que será adquirida, manipulada y transmitida en todo el sistema. [3]

En [12] Se diseñó un sistema electrónico el cual es capaz de discriminar entre la flexión del dedo meñique y pulgar de la mano. El error obtenido se encuentra al inicio de las muestras dado que es necesario contar con un entrenamiento previo. Se concluye que debido a la simplicidad de la comparación de umbrales de las envolventes de los dos canales sEMG, es posible determinar que dedo(s) se llegó a flexionar según los resultados.

2. Metodología del prototipo

La realización del proyecto está dividida en etapas de desarrollo, la adquisición y procesamiento de señales Electromiografías, como se observa en la figura 1. Posee un orden secuencial entre cada fase del proceso, por esta razón las etapas están ordenadas de la siguiente forma:

- Etapa de diseño para el circuito de acondicionamiento de señales mioeléctricas.
- Etapa de diseño para el circuito de procesamiento de señales Electromiografías
- Diseño de la PCB para la adquisición de señales EMG y Circuito de procesamiento
- Software para la visualización por computadora de las señales Electromiografías

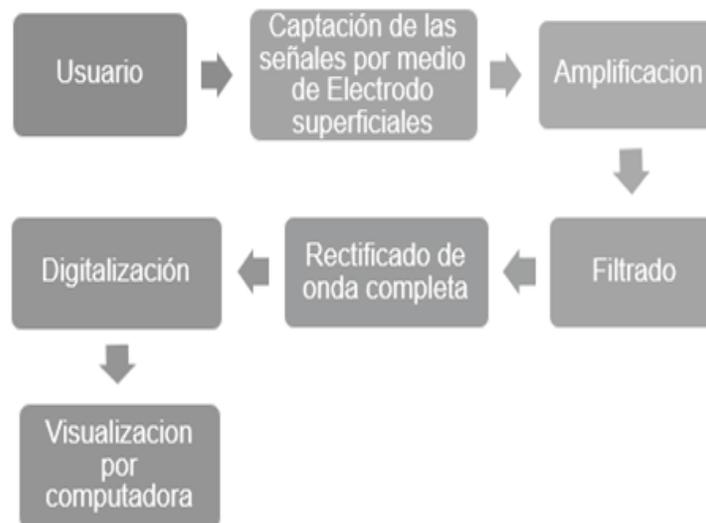


Figura 1. Funcionamiento del Dispositivo de adquisición y procesamiento de señales Electromiográficas.

2.1 Amplificación

Una señal EMG tiene rangos de amplitud que van desde 0.1 a 0.5 mV, los cuales son imposibles de poder adquirir y procesar directamente por computador [4], por ello es por lo que se emplean amplificadores de instrumentación para elevar los valores de milivolts hasta rangos de voltajes de lectura adecuados. se opta por utilizar el amplificador de instrumentación AD8226, debido a las siguientes características:

- Elevado rechazo en modo común preferentemente mayor a 80 decibeles para disminuir la interferencia debida a la red y para evitar la distorsión del biopotencial.
- Impedancias de entrada muy elevadas a 1 M Ω y así poder captar señales de bajo nivel.
- Baja corriente de polarización a la entrada.
- Disponibilidad en el mercado.
- Mayor grado de integración (menor tamaño).
- Bajo costo.



La exigencia de impedancias de entrada elevadas requiere de un tercer electrodo que permita el cierre a un punto de referencia de las corrientes de polarización a través del sujeto en observación. Para eliminar los potenciales comunes de los tres electrodos, como ruido ambiental y la alta impedancia proveniente de la piel, se aplicó la amplificación diferencial. Para obtener la ganancia deseada que es para este caso de estudio de 500, la cual favorece la adquisición de señales biológicas, elevando de milivolts o microvolts a valores comunes en unidades base, Volts, se emplean las fórmulas proporcionadas por el fabricante y se muestran en (1).

$$R_G = \frac{49.4K\Omega}{G-1} \quad (1)$$

Por lo tanto, se decide utilizar una resistencia de 100 ohm debido a que es la que se aproxima a una ganancia de 500, la configuración resultante del amplificador de instrumentación se muestra en la figura 2.

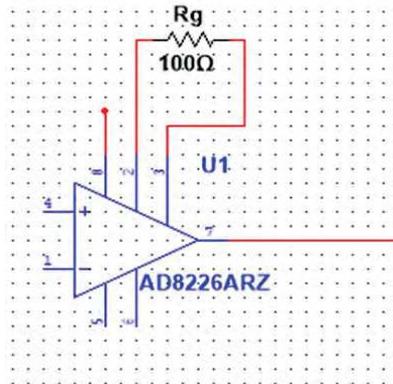


Figura 2. Configuración del amplificador operacional AD8226.

2.2 Filtro pasa altas

La frecuencia establecida para este filtro será de 10 Hz, y se representa por (2), y se utiliza un capacitor del valor comercial más común, en este caso es de 0.1μF, por lo que de la ecuación para calcular frecuencias se despeja la resistencia a utilizar y se busca el valor comercial más cercano.

$$F_H = \frac{1}{2\pi RC} \quad (2)$$

Por lo tanto, realizando los respectivos despejes, nuestra ecuación da como resultado (3).

$$R = \frac{1}{2\pi * 10F_H * 0.1 * 10^{-6}F} = 159154.96\Omega \quad (3)$$

2.3 Filtro pasa bajas

La frecuencia deseada para este filtro es de 450 Hz, tomando un capacitor de mismo valor que el filtro pasa altas, se tiene que calcular nuevamente la resistencia a utilizar en el circuito, la cual se tiene debería de ser 3.53 KΩ, ecuación (4), en el caso del cual se utilizaría la de valor comercial más cercano, 3.3KΩ. La figura 3 muestra el diagrama esquemático del filtro diseñado.

$$R = \frac{1}{2\pi * 450F_H * 0.1 * 10^{-6}F} = 3536.77\Omega \quad (4)$$

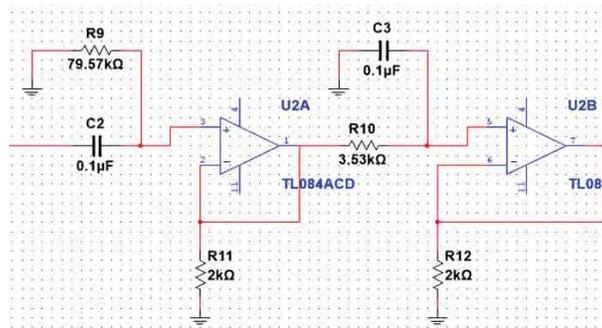


Figura 3. Diagrama eléctrico de Filtros activos.

2.4 Circuito de procesamiento

Se elegirá al ATmega328p como microcontrolador a usar. Se optó por esta opción debido a que ya se cuenta con experiencia en la programación de este dispositivo, además que presenta las siguientes ventajas para el proyecto:

- ADC de 10 bits de resolución como mínimo
- Receptor/transmisor interno serial asíncrono
- por lo menos dos pines de entrada/salida (para el manejo de la comunicación con una computadora)
- CPU de 8 bits
- Frecuencia máxima de operación del CPU: 20MHz
- Cantidad de canales del ADC: entre dos y ocho

La figura 4 muestra el diagrama esquemático para el procesamiento de señales Electromiográficas.

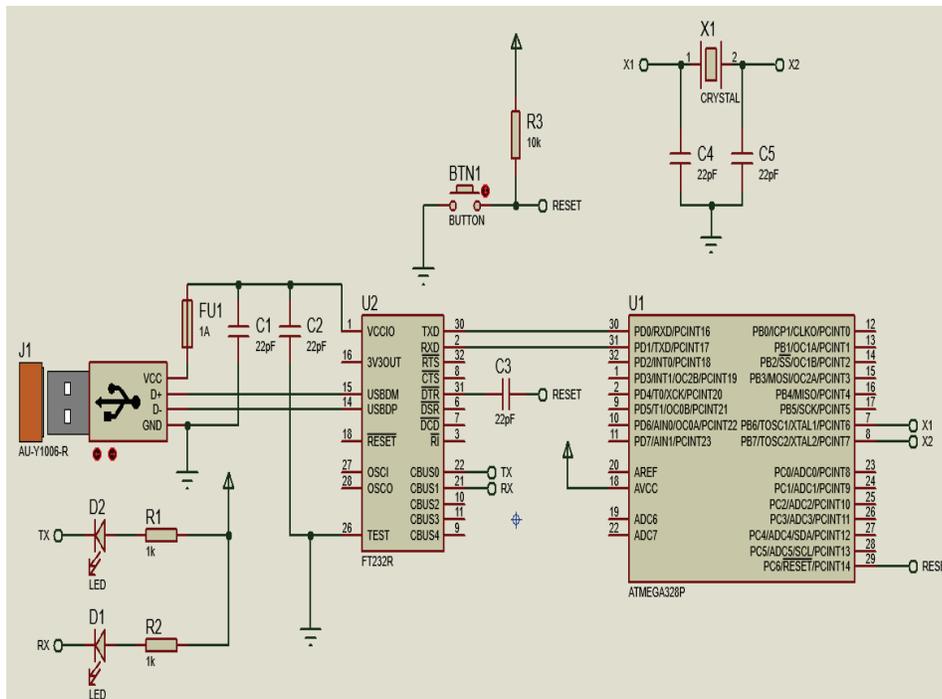


Figura 4. Diagrama electrónico de la etapa de procesamiento de señales Electromiográficas.

2.5 Diseño de la tarjeta de circuito impreso para el dispositivo de adquisición y procesamiento de señales Electromiográficas

Siguiendo las recomendaciones que establece las normas IPC se ubican los componentes en el área designada para la tarjeta de circuito impreso, en la figura 5, donde se muestra la ubicación de los componentes, también se resaltan las áreas que pertenecen a cada circuito que compone la PCB.

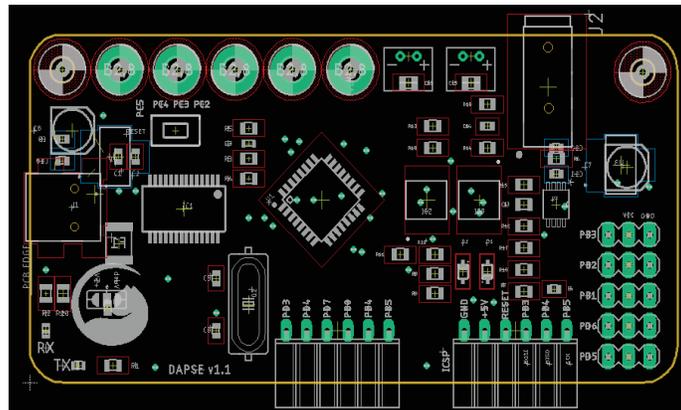


Figura 5. Ubicación de componentes en la tarjeta de circuito impreso.

En la figura 6 se muestra la vista frontal de la tarjeta de circuito impreso diseñada para el dispositivo de adquisición y procesamiento de señales Electromiográficas, además que se muestra la tarjeta electrónica ensamblada en la figura 7.

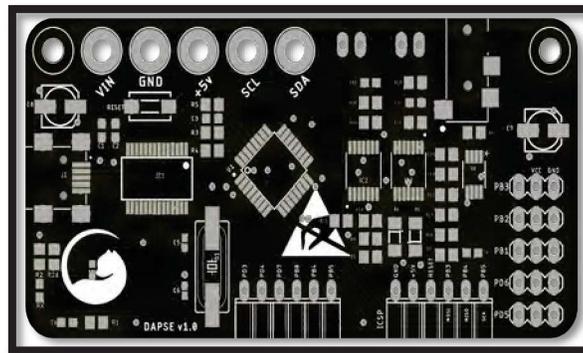


Figura 6. Vista frontal.



Figura 7. Tarjeta electrónica ensamblada (vista Frontal).

Para la fabricación de la carcasa del modelado digital del dispositivo se empleó una impresora 3D y filamento PLA, para dicha impresión se utilizó una resolución de calidad alta a 0.16 mm. La figura 8 muestra el dispositivo terminado, e incluyendo la tarjeta electrónica y la carcasa correspondiente.



Figura 8. Dispositivo completamente ensamblado (carcasa y tarjeta electrónica).

3. Resultados

A continuación, se muestran los resultados finales obtenidos.

3.1 Prueba de adquisición de señales Electromiografías

Procedimiento de comprobación:

- Para realizar la comprobación de captura de señales Electromiografías de la tarjeta electrónica ensamblada, se inicia posicionando los electrodos como se muestra en la figura 9. Los electrodos utilizados son desechables, por lo tanto, tienen un tiempo de uso limitado, se recomienda usar solo una vez durante un tiempo menor a dos horas, esto debido a que el gel que trae incorporado se seca rápidamente lo que limita su utilización y efectividad para capturar biopotenciales.

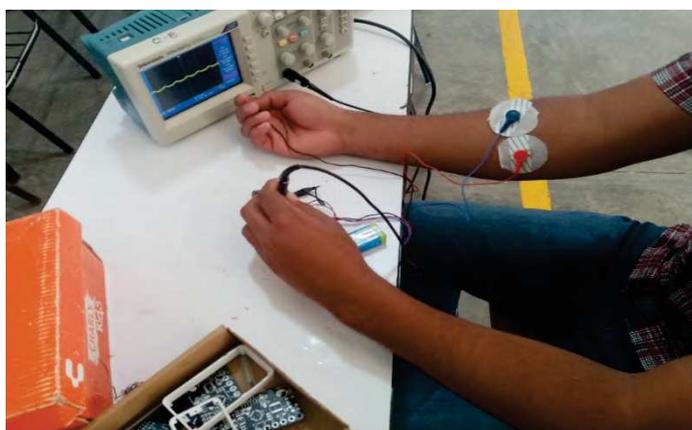


Figura 9. Posicionamiento de electrodos.

- Se conectan en las respectivas terminales dos baterías de 9 V para la alimentación del circuito de adquisición.

• La alimentación para la etapa de procesamiento y transmisión de la señal se obtendrá del puerto USB de la computadora.

Empleando un osciloscopio de la marca TEKTRONIX TDS 2001C, se capturan las siguientes señales mostradas en la figura 10, donde se muestra la señal emitida cuando se mantiene el brazo en reposo.

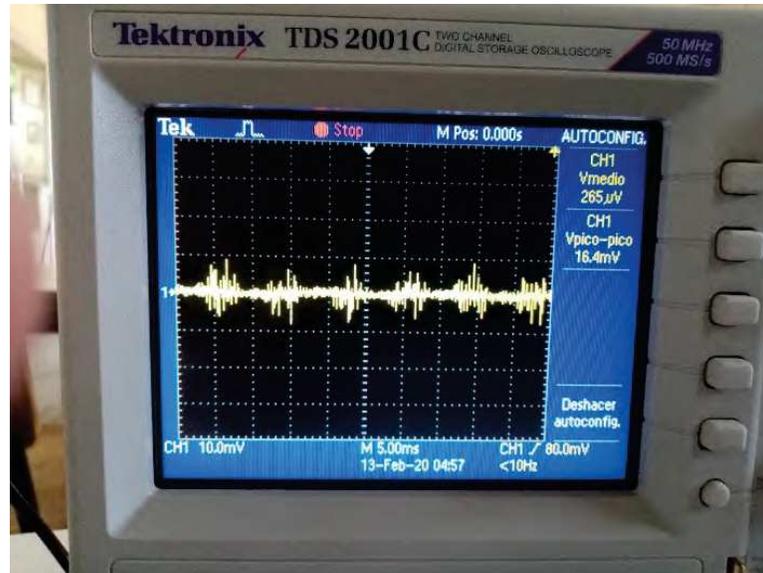


Figura 10. Señal emitida con el brazo en reposo.

A pesar de ser una señal muy débil generada por el músculo el circuito logra amplificarla hasta los 16.4 mV. Por otro lado, la figura 11 muestra la señal emitida cuando se presiona la palma de la mano.

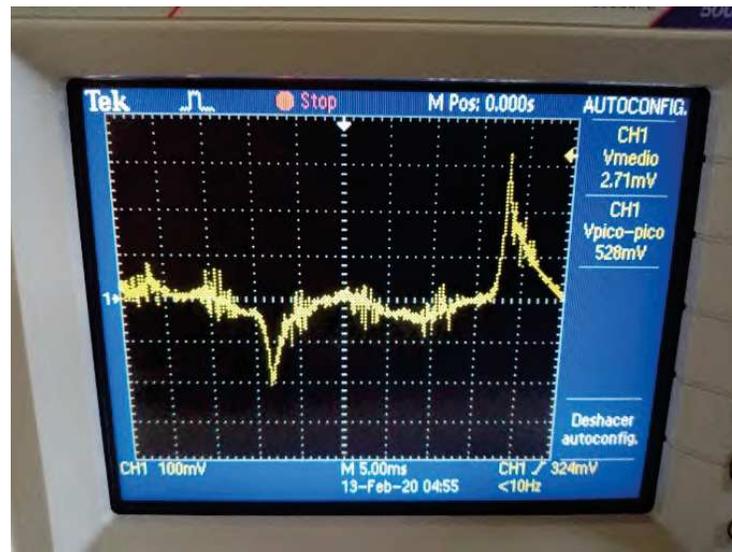


Figura 11. Señal emitida cuando se presiona la palma de la mano.

Comparación de las señales obtenidas:

Tabla 1. Comparativa.

Actividad del brazo	Periodo	Frecuencia	Amplitud
Brazo en reposo	Indeterminado	Indeterminado	16.4 mV
Presionando la palma de la mano	33.01 ms	30.30 Hz	528 mV

Se puede notar que las señales captadas por el osciloscopio son ampliamente distintas, debido a que, cuando el brazo está en un estado de reposo no existe ninguna generación de potenciales de acción por parte de las células motoras de los musculo. Sin embargo, en la prueba cuando se ejerce una contracción como la compresión de la palma de la mano, se genera un biopotencial y este es captada y amplificada por el circuito de adquisición de la tarjeta electrónica.

Por otro lado, el ruido electromagnético que existe en el ambiente afecta en parte la captura de señales mioeléctricas.

Para la comprobación de la visualización por computadora de señales EMG se realiza lo siguiente:

- Se colocan los electrodos sobre el musculo a censar
- Se conectan las baterías para alimentar el circuito de adquisición
- Se conecta el dispositivo a la computadora a través del puerto USB
- Se selecciona el puerto en la interfaz de visualización de señales EMG

En la figura 12 se muestra los potenciales de acción generados al presionar la palma de la mano, se puede apreciar que el biopotencial obtenido alcanza un valor aproximado de 2 V, por otro lado, mientras el brazo se mantiene en reposo, no se recibe ningún dato de presencia de señales EMG.

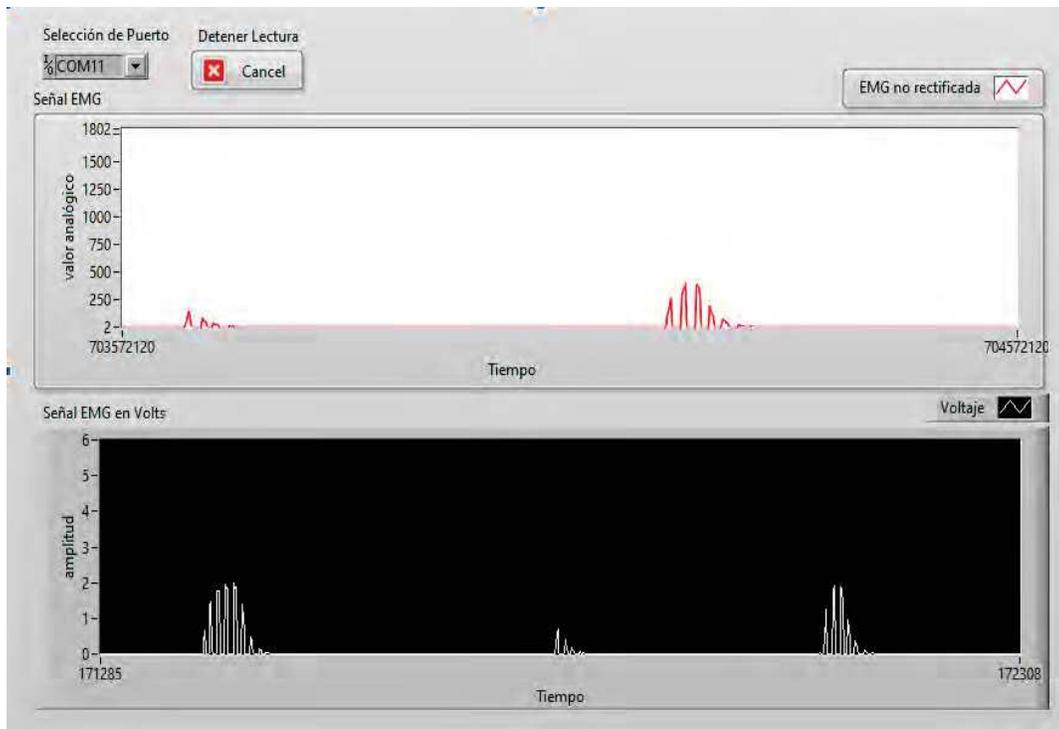


Figura 12. Señal EMG generada al presionar la palma de la mano visualizada a través de la computadora.



4. Conclusiones

Frente a la evidencia recaudada durante la elaboración de este trabajo, se concluye que se han realizado cada fase de la metodología aquí propuesta, teniendo en cuenta la complejidad que conlleva la captura de señales de origen biológico, se pudo lograr la adquisición de estos biopotenciales, por otro lado, la tarjeta electrónica que se diseñó y se manufacturo, debido a sus características es viable para otras utilidades. El dispositivo de adquisición y procesamiento de señales Electromiográficas, debe de servir como base para futuros trabajos o posibles mejoras de este.

4.1 Recomendaciones

Las señales mioeléctricas en las aplicaciones de la biomedicina tiene bastante futuro. El diseño propuesto en este documento es de base para los que deseen realizar un sistema actuado por este tipo de señales, ya que la etapa de amplificación y filtrado es necesaria y básica para poder trabajar con los potenciales de unidad motora. Las prótesis de piernas y brazos son ejemplos de dispositivos que pueden ser elaborados en base a la electromiografía. Así como también las señales mioeléctricas pueden ser utilizadas para sistemas de control, como por ejemplo para manipular el encendido y apagado de la luz, el televisor, y un sistema de potencia. También podrían implementarse en el control de abrir y cerrar la puerta, dirigir un robot o algún otro sistema futurista.

Es importante mencionar que el trabajo aquí presentado, a futuro, puede ser mejorado con el propósito de obtener mayores prestaciones en lo que se refiere a: rango y resolución de lecturas de señales Electromiográficas, nivel de procesamiento, con el cual se podría remplazar el microcontrolador por un microprocesador, todas estas mejoras contribuirían a analizar mejor los biopotenciales, abarcando muchas más áreas de investigación y desarrollo de prototipos basados en señales de carácter EMG.

Una mejora más que a futuro se podría implementar al dispositivo de adquisición y procesamiento de señales Electromiográficas es: la construcción de electrodos para la captura de señales EMG, se reducirían los gastos generados al usar el dispositivo, ya que en este trabajo los electrodos propuestos son desechables. Por último, la implementación de baterías internas de alimentación del circuito de adquisición, es ampliamente viable si se requiere un dispositivo de Electromiografía autónomo.

Referencias

- [1] P. Konrad. Version 1.0 April 2005. "The ABC of EMG", pag 20. [Online] Disponible: <http://www.analizaruchu.awf.wroc.pl/materialy/abcemg.pdf>.
- [2] Piña-Díaz, A. "Implementación de señales mioeléctricas en un sistema de control para generar movimientos en una prótesis personalizada de mano", Instituto Politécnico Nacional, 2015.
- [3] Casas, T., K. y De la Cruz, R., M., *Sistema de adquisición y procesamiento de las señales mioeléctricas*, Universidad Autónoma de Ciudad Juárez, 2013.
- [4] De Bel Maxime, Cantraine Francis, Mathys Pierre. (2005). *APPLIANCE AND METHOD FOR MEASURING AN EMG SIGNAL*. No. 11667439. Estados Unidos de América: Colecciones nacionales e internacionales de patentes.
- [5] Finneran Mark T., Bihari Thomas E., Pugh Dennis R. (1998). "COMPUTERIZED EMG DIAGNOSTIC SYSTEM". No. 98918418. Oficina Europea de Patentes (OEP): Colecciones nacionales e internacionales de patentes.
- [6] NASA. (2012, 28 octubre). "Technology Readiness Level". Recuperado 6 septiembre, 2019, de https://www.nasa.gov/directorates/heo/scan/engineering/technology/txt_accordion1.html
- [7] Flores A. (2010). "Sistema avanzado de prototipado rápido para control en exoesqueletos y dispositivos mecatrónicos". Tesis doctoral. Universidad Carlos III de Madrid.
- [12] A. C. MettingVanRijn, A. Peper, C. A. Grimbergen. (2008). "Amplifiers for bioelectric events: a design with a minimal number of parts. 2019", noviembre 16 <http://www.biosemi.com/publications/artikel7.htm>.



- [9] Programa de la asignatura "*Aplicaciones de la automática en la biomedicina*". Grado en Ingeniería Electrónica y Automática. Universidad Carlos III de Madrid. Disponible: http://www3.uc3m.es/reina/Fichas/Idioma_1/223.14061.html.
- [10] Ramírez, S. E. B. (2018, 3 octubre). "*Aplicaciones de los niveles de madurez de la tecnología*". Recuperado 6 septiembre, 2019, de <http://pivot.com.mx/2018/10/03/aplicaciones-de-los-niveles-de-madurez-de-la-tecnologia/>
- [11] STM32F4 – DISCOVERY (2012) "*Manual de usuario*". [http://www.st.com/st-web-ui/static/active/en/resource/technical/document/user_manual/DM00039084.p df](http://www.st.com/st-web-ui/static/active/en/resource/technical/document/user_manual/DM00039084.pdf).
- [12] Alva Coras. A, (2012) "*PROCESAMIENTO DE SEÑALES DE ELECTROMIOGRAFÍA SUPERFICIAL PARA LA DETECCIÓN DE MOVIMIENTO DE DOS DEDOS DE LA MANO*", Facultad De Ingeniería Escuela Profesional De Ingeniería Electronica, Lima, Perú .



Plano automatizado de reaprendizaje neuromotor para pacientes con Enfermedad Vascular Cerebral

Ángeles Martínez Juan Pablo, Pérez Prado Edgar Eduardo.

Universidad Autónoma de Querétaro, Facultad de Ingeniería.
jangeles28@alumnos.uaq.mx, ededpp98@gmail.com

Resumen

El presente proyecto se realizó con la finalidad de apoyar a las personas con eventos cardiovasculares, mediante el diseño y construcción de un aparato para la rehabilitación. Este dispositivo consiste en un plano automatizado de dos ejes y un ambiente virtual, se tiene como objetivo ayudar a los pacientes a recuperar movilidad en sus brazos y así permitirles volver a un modo de vida donde no requieran un gran apoyo de parte de terceros, también se espera reducir tiempo de rehabilitación, carga laboral a enfermeras y fisioterapeutas, esfuerzo y dinero. Una Enfermedad Vascular Cerebral consiste en una alteración en las neuronas que provoca disminución de flujo sanguíneo en el cerebro, acompañada de alteraciones cerebrales de manera momentánea o permanente. A lo largo del proyecto determinamos que aproximadamente entre un 70% y un 80% de las personas que han sufrido un EVC, tienen un deterioro en miembro superior y requieren de manejo médico prolongado para disminuir su discapacidad física.

Palabras clave: Salud, Evento Vascular Cerebral (EVC), automatización, ambiente virtual, neuromotor, Rehabilitación.

1. Introducción

Un EVC (Enfermedad Vascular Cerebral) consiste en una alteración en las neuronas que provoca la disminución del flujo sanguíneo en el cerebro, acompañada de alteraciones cerebrales de manera momentánea o permanente [1]. A lo largo del proyecto se determina que aproximadamente entre un 70% y un 80% de las personas que han sufrido un EVC, tienen un deterioro en el miembro superior y requieren de asistencia médica prolongada para disminuir su discapacidad física [2].

1.1 Antecedentes

Este proyecto surge principalmente al entender la problemática que nos planteó el Dr. Fernando Cruz, quien trabaja en el ISSSTE, la problemática radica en el hecho de la pérdida de movilidad en brazos a causa de un evento cerebrovascular (EVC), fue así que nos interesamos en desarrollar un proyecto enfocado para mejorar su proceso de rehabilitación y posteriormente al conocer el departamento de fisioterapia donde brindan el servicio, nos propusimos crear un equipo que ayudé a resolver por medios digitales.

¿La realidad virtual podría ayudar a la rehabilitación del brazo en algunos pacientes con accidente cerebrovascular?

Encontramos que ya existen entornos virtuales desarrollados para el mismo propósito en conjunto con su respectivo análisis y según un estudio piloto clínico publicado en 'Journal of NeuroEngineering and Rehabilitation'. Los investigadores encontraron que el uso de la realidad virtual para aumentar la confianza del paciente en el uso de su brazo paralizado puede ser crítico para la recuperación [3].



Sin embargo, encontramos que todos ellos tienen una característica en común, la cual es que no cuentan con la estimulación física externa, es decir no poseen un elemento físico tangible con el cual puedan interactuar en conjunto con el entorno virtual. Con esto queremos mencionar nuestra hipótesis, la cual es: El hecho de que la realidad virtual podría tener mejores resultados se le incorpora un elemento que genere el estímulo motor, el cual nosotros proponemos.

Es por ello por lo que lo que nosotros queremos innovar en el campo de la medicina es que mediante el uso de dispositivos automatizados y agregando medios digitales, logremos que los pacientes tengan una recuperación más completa y en un tiempo menor.

1.2 Justificación e importancia

En el aspecto médico se desenvuelve con sobrevivientes del ataque cerebral, con el fin de volver a aprender aptitudes que se han perdido cuando se daña parte del cerebro dentro de las cuales se encuentra la coordinación de los movimientos.

"Los pacientes con ataque cerebral que tienen hemiparesia o hemiplejía podrían tener dificultad con las actividades diarias como caminar y agarrar objetos"

El tratamiento de rehabilitación posterior al ataque cerebral se debe empezar una vez estabilizado el paciente, con la prioridad de evitar la pérdida de la motricidad del paciente, actualmente las enfermeras y terapeutas de rehabilitación ayudan a los pacientes que son capaces de realizar tareas progresivamente más complejas y exigentes, como bañarse, vestirse e ir al baño, y los alientan a comenzar a usar sus miembros afectados por el ataque cerebral mientras participan de esas tareas. Comenzar a recuperar la capacidad de llevar a cabo estas actividades básicas de la vida diaria representa la primera etapa del regreso del sobreviviente de un ataque cerebral a la independencia [4].

Cabe añadir, La enfermedad vascular cerebral (EVC) es un problema importante de salud pública. De acuerdo con la Organización Mundial de la Salud (OMS), es considerada la segunda causa global de muerte en el mundo, siendo los países de ingresos medios y bajos, los más afectados. México forma parte de este grupo de países, donde de cada 100,000 habitantes, 118 habitantes padecen de esta condición al año, principalmente en accidentes automovilísticos o dado por causas naturales, Otro ejemplo es, España las enfermedades vasculares cerebrales (EVC) son una causa muy frecuente de morbilidad y hospitalización, constituyendo la segunda causa de mortalidad en la población general, y la primera en las mujeres [5].

1.3 Objetivos

Hacer un equipo enfocado a la rehabilitación de los brazos de pacientes con EVC. Moviendo el brazo y hombro mediante un plano automatizado acompañado de un panorama de realidad virtual se pueda obtener una recuperación motora en menor tiempo que con los procedimientos actuales.

1.4 Objetivos específicos

Elaboración del equipo automatizado

Equipo de desplazamiento para brazo. Este equipo se distingue es el uso de dos bandas transportadoras que trabajan en un plano, es decir sobre el eje "X" y "Z", estas serán las encargadas de mover el brazo de manera lineal (izquierda - derecha, adelante - atrás) formando así un plano.

Elaboración de panorama virtual

Se desarrollará un entorno para el paciente a modo de escenario, en el cual tendrá que pasar ciertas dinámicas las cuales desarrollará con ayuda del plano "X" y "Z".



Evaluación funcional

Como último plano se considera llevar al equipo a revisión en conjunto con autoridades médicas para que puedan evaluar el desempeño de este en conjunto con los pacientes. Encontramos que el departamento de Investigación del ISSSTE brinda apoyo a este tipo de proyectos.

2. Diseño del plano automatizado

2.1 Selección de materiales y construcción

Perfiles y láminas de aluminio: Se seleccionó que los materiales fueran de aluminio dado que cumple propiedades que es de nuestro interés ejemplo que el aluminio es altamente estable y resistente a impactos, es higiénico algo que era muy importante a considerar por el hecho que se plantea que se trabaje en hospitales, también que lograba soportar el peso de los motores y la del brazo sin deformarse y su facilidad que tiene el material para trabajar con él.

-Tela polyester: Esta tela posee propiedades como que es liviano y que tiene baja absorción a la humedad de tal manera que se evita la generación de malos olores [6].

Acrílico: Al igual que el aluminio los factores que más nos importan de este material era que su costo es accesible y que es muy liviano, aportando de esta manera que el dispositivo tuviera el menor peso posible.

Motor a pasos: se decidió la utilización del motor a pasos por la precisión que tiene y que es más fácil de controlar su velocidad; ya que era necesario llevar un control estricto de los tiempos para no causar daños al paciente.

Arduino: para la programación se escogió una tarjeta Arduino que su plataforma de programación es muy sencilla de manejar.

2.2 Diseño del prototipo y planteamiento de la elaboración

La estructura mecánica del proyecto es mayormente de aluminio y acrílico por lo fácil de manipular trabajar con estos materiales, además por lo ligeros que son.

El sistema mecánico está conformado por dos motores uno que se mueve para el eje Z y otro en el eje X.

Componentes del diseño

- Láminas de doble cara
- Perfiles metálicos
- Acrílico
- Motores
- Bandas
- Rodillos
- Gabinetes

Según lo mencionado anteriormente, cada una de las piezas fue analizada para tomar la mejor decisión para el dispositivo.

En el primer prototipo se utilizaron los perfiles metálicos como soportes y el acrílico fue usado como base para el láser. Los motores a pasos fueron montados de tal manera que no fueran visibles,

en la parte inferior del acrílico. Para poder de manera sencilla el prototipo se dividió en dos partes, La primera es para el eje X y el segundo para el eje Y. Figura 1 y 2.

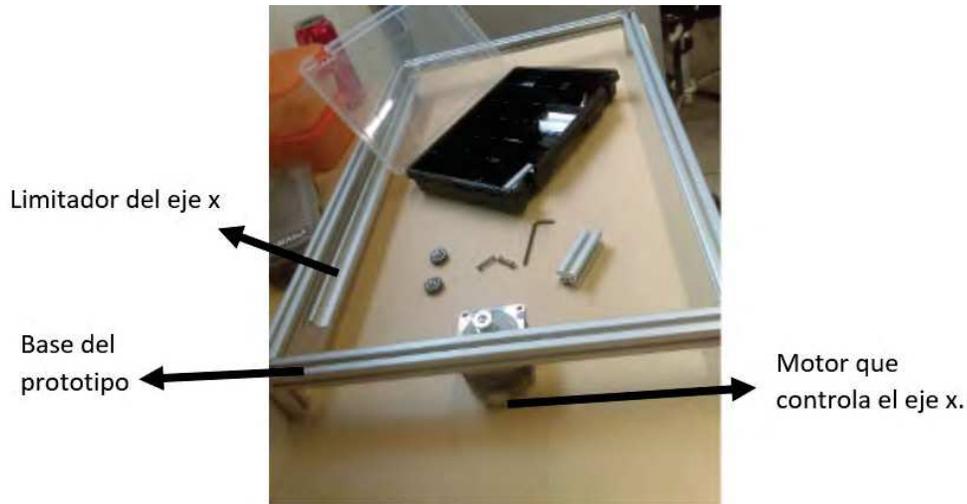


Figura 1. Primer prototipo de plano X en proceso de armado.

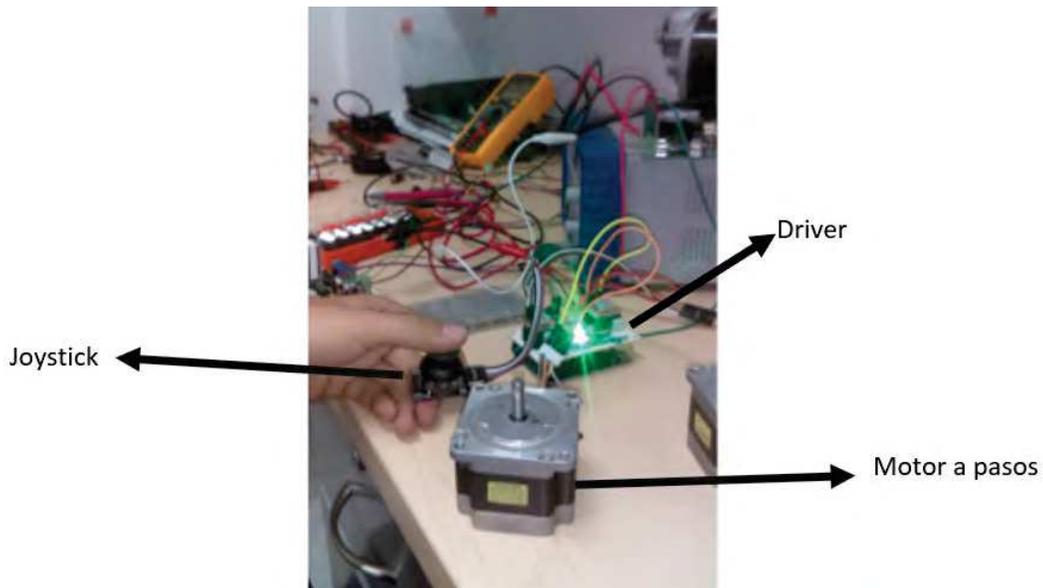


Figura 2. Control motor a pasos con Joystick.

Para poder tener movilidad en el eje X, se utilizó las bandas pegadas en la parte de los perfiles metálicos y con el motor en un extremo. El cual lograba mover la parte con la que se movía el segundo eje.

La segunda parte tenía dos perfiles metálicos y un acrílico, que ayudaban con el soporte y la unión de los dos perfiles. En esta parte se incorporó el segundo motor para mover en el eje y, al igual que en el primero, se colocó por debajo de la base por cuestiones estéticas.

En el área de la protección de los motores se agregó goma para proteger de daños al prototipo, en el contorno de los perfiles se debía de evitar las esquinas por cuestiones de seguridad de los pacientes, entonces se agregó acrílico y goma en todo el perímetro.

Para soporte del brazo se utilizó velcro, argollas y lámina de PVC para estar sólido que fue termoformado para adaptarlo al brazo, además de lograr que pueda adaptarse para diferentes tamaños. Por cuestiones de comodidad y estética se agregó tela tipo polyester. Figura 3.



Figura 3. Primer prototipo del plano X,Z completo.

2.3 Programación del prototipo

En el área de la programación se utilizó un Arduino para controlar los motores de X y Z al mismo tiempo que fuera sencillo de verse en el entorno virtual creado. Dado la facilidad que tiene Arduino para crear la comunicación de control de los tiempos y es fácil conectar con los dispositivos visuales. Con el Arduino logramos que fuera casi instantánea la relación del movimiento del láser con lo que aparece en la pantalla.

Para la programación de los motores en algunos casos se buscó que el movimiento fuera lento y controlado el movimiento y de manera sincronizada ambos ejes para evitar causar lesiones al usuario. La programación lograda por el dispositivo Arduino facilitó de gran manera este hecho. Figura 4.

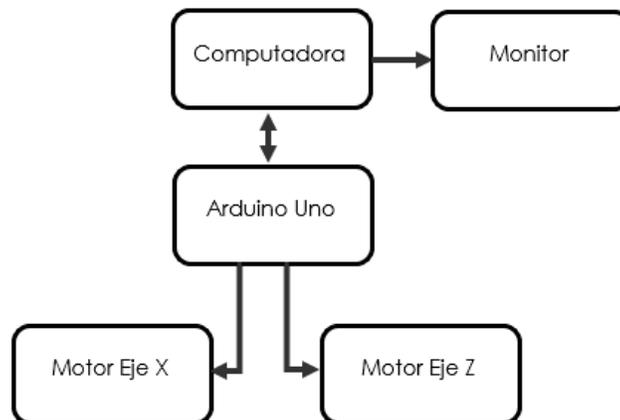


Figura 4. Diagrama de conexión del dispositivo



3. Programación del entorno virtual

El entorno virtual se establece para resolver múltiples complicaciones que otros proyectos similares han sufrido, estos problemas van desde abandono temprano de la rehabilitación por resultados poco notables, hasta por aburrimiento debido a las actividades realizadas en la misma rehabilitación.

La solución planteada es incorporar juegos en el entorno virtual para incentivar a los pacientes a seguir avanzando con su rehabilitación. La idea de usar juegos llamativos viene de la “adicción” que estos causan a los jugadores, donde se les incentiva a seguir jugando para lograr diferentes objetivos, conseguir retos más complicados, etc. Estos juegos fueron programados con Unity ya que es muy accesible y sencillo el programa. Diagrama del funcionamiento del proyecto figura 5.

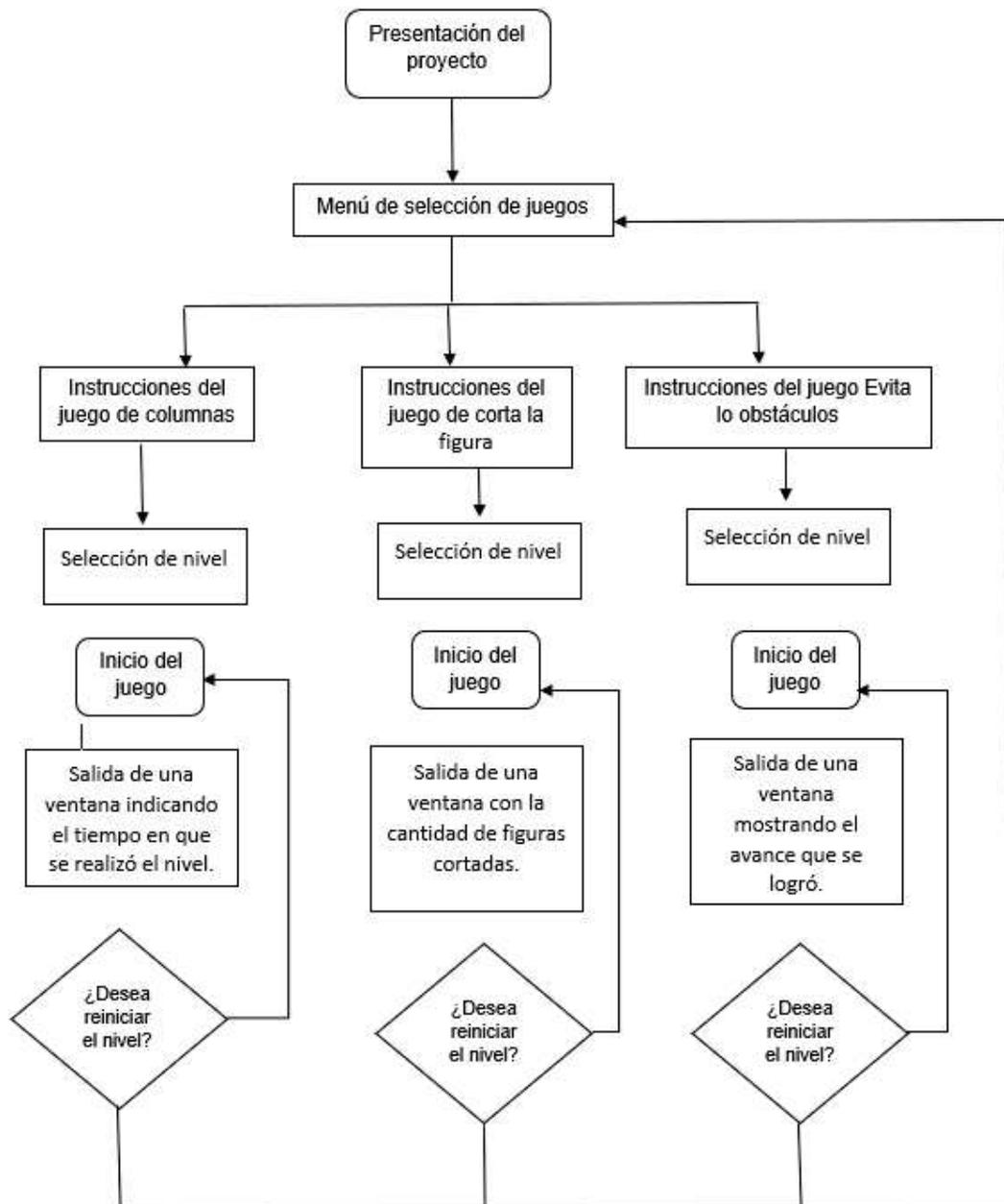


Figura 5. Diagrama de flujo de juegos



3.1 Selección de juegos

Para seleccionar los juegos nos centramos en varias características clave que estos deben de tener, empezando con que estos juegos de ser repetitivos, esto se debe a que el paciente debe de realizar muchas series de movimientos para su rehabilitación. Deben ser fáciles de entender y no muy complicados de realizar.

Se seleccionaron tres juegos que cumplen con las características buscadas por el equipo. Los juegos seleccionados consisten en lo siguiente:

1. En una malla se tienen múltiples cuadros de colores, con cada color teniendo un par de cuadros, el objetivo del juego es llenar la malla conectando los colores con su par, sin que los colores se crucen. Este juego es para los pacientes con mayor control de sus brazos ya que requiere movimientos en los ejes vertical y horizontal relativamente precisos. Para seleccionar el color para conectar, el paciente tiene que colocar el puntero sobre el color y esperar 3 segundos para que el color se seleccione.
2. El juego consiste en esquivar obstáculos que aparecen frente al jugador, este debe de mover su brazo verticalmente sobre el plano para mover su personaje y así evitar chocar con muros, mientras más muros esquive mayor será la puntuación.
3. El paciente debe de atrapar frutas que caen desde la parte superior de la pantalla, este tiene que mover el brazo de forma horizontal al plano para así poder atrapar las frutas antes de que caigan. Este juego es más sencillo que los otros dos ya que solo requiere un movimiento de brazo que puede venir desde el hombro.

La interfaz consta de los siguientes elementos:

Pantalla principal: Consta de un menú donde se puede seleccionar uno de los tres juegos o salir de la aplicación, la selección se realiza colocando el puntero sobre el ícono de los juegos o salir durante tres segundos.

Interfaz de juego: Cada juego consta de una interfaz propia, donde los puntos que comparten son el nivel, temporizador y el puntaje.

Pantalla de fin de partida: Al terminar cada nivel el juego da la opción de avanzar al siguiente nivel, repetir el nivel actual o salir, además de esto también muestra el puntaje y el tiempo total jugado durante la sesión.

3.2 Función de los juegos

La función principal por la que se escogió utilizar juegos es para motivar y entretener a los pacientes mientras llevan su tratamiento de rehabilitación.

Los juegos cuentan con diferentes niveles de ayuda, en los cuales, dependiendo de la movilidad de los pacientes, los motores del sistema ayudarán a controlar el movimiento ejercido por los pacientes para así estimular el sistema nervioso del paciente y hacerlo creer que él está moviendo el brazo por su propia cuenta.

3.3 Emparejamiento con el plano automatizado

El monitor es del mismo tamaño que el plano automatizado, esto permite que la relación del movimiento del plano con el cursor del monitor sea uno a uno. Esto es de gran ayuda ya que permite una mejor inmersión al juego de parte del paciente.

Para los niveles que requieren que el motor a pasos se mueva se utiliza un programa de Arduino, el cual recibe instrucciones de la aplicación de los juegos, la cual le indica en qué dirección moverse y la cantidad de pasos que debe de realizar, esta opción actualmente solo funciona para el juego de atrapar frutas, donde solo se requiere movimiento horizontal.

4. Implementación, prueba del funcionamiento y ajustes finales

4.1 Implementación

Se realizaron dos pruebas para verificar el correcto funcionamiento del dispositivo completo:

1. La prueba se realizó con los integrantes del equipo médico, Se autorizó a realizar la incorporar paciente con EVC bajo la supervisión médica, en esta parte se revisó que los movimientos de ambos ejes estuvieran a velocidades adecuadas para trabajar.
2. Se llevó el prototipo para que un paciente con EVC en recuperación lo probara. Con el paciente se tuvo dificultades para colocar el dispositivo dado por esto se decidió mejoras en el diseño en la funcionalidad del proyecto no se tuvo ningún conflicto, ya que logró cumplir su objetivo. Figura 6.



Figura 6. Primera prueba con paciente de EVC.

Posteriormente de realizar una visita al Hospital General de Querétaro y realizar pruebas con un paciente de EVC se observaron algunos defectos con el primer diseño del plano automatizado, por lo que se decidió integrar el plano directamente en la mesa, en vez de ser un dispositivo aparte. Figura 7 y 8.



Figura 7. Segundo prototipo del plano X,Z (vista superior).



Figura 8. Segundo prototipo del plano X,Z (vista frontal).

4.2 Ajustes finales

Se modificó el diseño del proyecto, dado que el eje X y Z limitaba la movilidad del paciente se cambió a solo trabajar con motor en un solo con el eje X y dejar que el mismo paciente pueda mover en el eje Z. También se construyó una mesa que pudiera cambiar su altura para que se adaptara mejor a las camas del hospital. Igualmente se cambió la base y se implementaron los perfiles a la mesa siendo esta la nueva base, conectando por debajo el Arduino y los motores.

Debido a que se tenían los ejes X y Z con motores, la movilidad del paciente se veía ligeramente limitada en el eje Z, por lo que se optó por dejar este eje libre y el eje X con motor. Se instaló el plano directamente sobre la mesa para reducir la altura a la que se tiene que suspender el brazo del paciente, esto se debe a que no todas las camas hospitalarias tienen la opción de elevarse o inclinarse al paciente a una posición adecuada para la rehabilitación. Figura 9.



Figura 9. Prueba del dispositivo con el monitor.

4.3 Elaboración del manual

Se elaboró un manual para el uso del operario, donde se explica el cuidado del dispositivo, explicación de cómo usarlo correctamente y el cómo conectarlo. También el cómo se debe de colocar al paciente para que esté cómodo y pueda llevar la rehabilitación de manera adecuada.

4.4 Entrega de productos

Prototipo: El prototipo entregado se compone de los elementos señalados a continuación.

- Soporte del paciente
- Mesa de trabajo
- Monitor
- Carro deslizante
- Compartimento para el equipo electrónico
- Contenedor de elementos desmontables
- Soporte de altura variable
- Ruedas libres con seguro
- Eliminador y cables del monitor
- Manual

El equipo se estructura de tal manera que sea modular, es decir, que no requiera de otros elementos (con excepción de una conexión de corriente) para su funcionamiento. Es por esto por lo que añadimos un contenedor en el cual se guardan todos los elementos móviles que corren riesgo durante el transporte del prototipo. Figura 10.



Figura 10. Partes del dispositivo final.

5. Resultados

En el proyecto se obtuvieron los siguientes resultados.

1. Dispositivo calificado para la rehabilitación de pacientes con EVC; conformado por un plano automatizado de 2 ejes, entorno virtual y soporte para el brazo. Dando con el resultado un equipo que cumple con los movimientos requeridos para la rehabilitación y seguro para el usuario.
2. Coordinación entre el plano automatizado y el entorno digital, donde los movimientos llevados a cabo por el usuario se ven reflejados en el entorno digital de manera instantánea.
3. El entorno digital cuenta con tres aplicaciones con las cuales interactúa el usuario de manera de juego. Cada aplicación sirve para estimular distintas habilidades motoras donde se contará con varios niveles de dificultad y asistencia mecánica con la finalidad de la recuperación del paciente.



4. Que el equipo cumpla con las normas médicas básicas que contemplan a un equipo médico para su posible incorporación al sector salud, siendo así estéticamente presentable y ergonómico, listo para ser empleado para los pacientes para los cuales fue diseñado.
5. Se llevó ante el departamento de enseñanza del hospital General de Querétaro para así determinar la factibilidad y los requerimientos necesarios para desarrollar una etapa de investigación y llevarlo al sector salud, con ello se logró recibir retroalimentación por parte de los doctores (traumatólogos en su mayoría) sobre las modificaciones pertinentes.

Impacto obtenido.

1. Investigación aplicada en el ramo del entorno digital y la automatización para fines médicos como nuevas opciones de reintegración del paciente.
2. Médico - Desarrollo del equipo, diseñado para realizar terapias que involucran tanto la mente como el cuerpo del paciente, permitirá obtener resultados en una menor cantidad de tiempo, y una mayor cantidad de personas que podrán ser beneficiadas.
3. Social - Lograr la reintegración de un paciente a sus labores diarias después de haber sufrido un evento cerebrovascular.

¿Qué es lo que sigue?

Al presentar esta propuesta en el hospital, se encontró gran interés en este proyecto, desde el área de traumatología, y por ello se nos invitó a participar en el congreso de lesionado medular, mismo que busca exponer los avances que se logran para beneficio del paciente.

Por otra parte, el coordinador de traumatología Dr. Tovar, nos ofreció llevar a cabo un protocolo de investigación aplicada para nuestro proyecto, es decir llevar el equipo a determinados pacientes para poder evaluar el desempeño real y tener un control del progreso comprendido para la rehabilitación del paciente. Esto aunado a previos estudios éticos y consentimiento por parte de los superiores.

5.1 Presentación de trabajos en eventos académicos.

1. Presentación en el congreso “Manejo del lesionado Medular”. Fecha: 28 - octubre - 2018, Lugar: Hotel Plaza Camelinas. Objetivo: Se presentó en el congreso de “Manejo integral en el lesionado medular”, con la finalidad de dar a conocer el desarrollo en equipamiento para el tratamiento de la recuperación de la movilidad en extremidades inferiores.
2. Investigación interna en el Hospital General de Querétaro. Fecha: 20 agosto – 16 octubre, Lugar: Hospital General de Querétaro. Objetivo: Con la finalidad de preparar el prototipo para un futuro protocolo, se hicieron diversas simulaciones dentro del departamento de enseñanza para poder realizar las mejoras observadas. (en esta etapa existieron correcciones hacia el prototipo).
3. Difusión en la UAQ. Fecha: 15 - agosto - 2018, Lugar: UAQ – Radio. Objetivo: Por parte de la facultad en conjunto con el sistema FOPER, se llevará a cabo una entrevista, esto con la finalidad de invitar a los alumnos a participar en este tipo de proyectos, así como informar el estatus actual del proyecto y cómo ha surgido el avance de este.
4. Difusión en eventos educativos. Fecha: 23 a 27 - octubre - 2018, Lugar: Centro Cultural Manuel Gómez Morin. Objetivo: Por parte de la facultad de ingeniería en conjunto con CONCYTEQ, participamos en el evento número 32 de EXPOCYTEQ “Desastres Naturales: Terremotos y Huracanes”, en donde se mostraron los avances que la facultad realiza entorno tecnologías para recuperación de la salud.



6. Conclusiones

El dispositivo creado durante el desarrollo de este proyecto cumple con el propósito de ser libre en los ejes X y Z, pero no con el de tener asistencia de movimiento con motores en ambos ejes, por el hecho de que la longitud de los brazos varía de persona a persona y que se necesitaría tener varias secuencias de movimiento para diferentes rangos de longitudes de brazos, se optó por dejar el eje Z completamente libre, dejando la asistencia solo al eje X.

Debido a que el proyecto se realizó en los primeros semestres de la Universidad, muchos procesos mecánicos y electrónicos eran desconocidos para el equipo de trabajo, por lo que en un futuro el proyecto tiene mucho potencial para perfeccionar.

Actualmente el dispositivo se encuentra instalado y operando en el Hospital General de Querétaro, donde se utiliza con pacientes con EVC y se evalúa su desempeño con ellos.

Referencias

- [1] Desconocido. “*Enfermedad Vascolar Cerebral*”, <http://www.imss.gob.mx/salud-en-linea/enfermedad-vascular-cerebral>, Mexico, 26/11/15.
- [2] Fernández del Castillo Quintana M. I., Hernández-Franco J., Barragán Campos H. M., Sánchez-Villavicencio I., H. Pasaye E. (2012). “*Efecto del uso de realidad virtual como estrategia de manejo para la recuperación motora del miembro superior, en pacientes con enfermedad vascular cerebral crónica, evaluada mediante RMf*”, Consultado el 10 de octubre de 2020, <https://realidadfuturista.files.wordpress.com/2016/05/rv-y-recuperacic3b3n-motora.pdf>
- [3] Rubio Ballester B., Nirme J., Duarte E., Cuxart A., Rodríguez S., Verschure P., Duff A. (2015). “*The visual amplification of goal-oriented movements counteracts acquired non-use in hemiparetic stroke patients*”. JOURNAL OF NEUROENGINEERING AND REHABILITATION. <https://link.springer.com/article/10.1186/s12984-015-0039-z>
- [4] Office of Communications and Public Liaison National Institute of Neurological Disorders and Stroke National Institutes of Health Bethesda, MD 20892. (2016). Rehabilitación posterior al ataque cerebral. 10 de octubre del 2020, de National Institute of Neurological Disorders and Stroke Sitio web: https://espanol.ninds.nih.gov/trastornos/rehabilitacion_posterior_al_ataque_cerebral.html
- [5] Brea, A., Laclaustra, M., Martorell, E., & Pedragosa, À. (2013). Epidemiología de la enfermedad vascular cerebral en España. *Clínica e Investigación en Arteriosclerosis*, 25(5), 211-217.
- [6] Textilon,” *¿Qué es el poliéster? ¿Para qué se utiliza?*”, España, 2016, <https://textilon.es/2016/04/14/el-poliester-en-prendas-deportivas-y-merchandising/>

Desarrollos con Enfoque Mecatrónico

Asociación Mexicana de Mecatrónica A.C.
Prolongación Corregidora Norte No. 168, interior E
Col. El Cerrito, C.P. 76160, Querétaro, Qro.
Distribución en formato digital por internet,
tiraje: 1000, pp. 205, Tamaño: 47 Mb.

ISBN: 978-607-9394-22-6

ISBN: 978-607-9394-22-6



9 786079 394226

