

Prototipo de mano robótica controlado mediante señales electromiográficas con un dispositivo comercial

Manuel Arias-Montiel¹, Asis Martínez-Miguel¹, Ester Lugo-González¹,
Rosebet Miranda-Luna¹, Ricardo Tapia-Herrera²

¹ Universidad Tecnológica de la Mixteca,
Insitute of Electronic and Mechatronics,
México

² CONACYT-Universidad Tecnológica de la Mixteca,
México

{mam, elugog,, rmiranda}@mixteco.utm.mx, ricardo.tahe@hotmail.com

Resumen. En este trabajo se presenta la aplicación del brazalete mioeléctrico comercial MYO[®] para el control de movimiento de una mano robótica. La adquisición y el procesamiento de los datos se realizan utilizando MATLAB[®] y la aplicación *Classification Learner*, los datos procesados son utilizados para controlar los servomotores que accionan los dedos de la prótesis mediante una tarjeta Arduino[®]. La mano robótica se imprime en 3D a partir de código libre. Los resultados experimentales muestran que es posible identificar movimientos de la mano a partir de señales electromiográficas (EMG) del antebrazo para ejecutar movimientos básicos con la mano robótica.

Palabras clave. Brazalete MYO[®], mano robótica, electromiografía.

Prototype of Robotic Hand Controlled by Electromyographic Signals with a Commercial Device

Resumen. In this paper, the application of myoelectric commercial armband MYO[®] to control the motion of a robotic hand is presented. Data acquisition and processing is carried out by MATLAB[®] and the *Classification Learner* application. The processed data are used to control the servomotor which actuates the prosthesis fingers by an Arduino[®] card. The robotic hand is printed in 3D from free code. The experimental results show it is possible to identify hand movements from forearm electromyographic (EMG) signals in order to execute basic movements with the robotic hand.

Palabras clave. MYO[®] armband, robotic hand, electromyography.

1. Introducción

Las señales electromiográficas (EMG) proporcionan información acerca del funcionamiento de los músculos y nervios midiendo señales eléctricas generadas por los músculos durante su movimiento. Las señales EMG representan una medida de la actividad muscular y pueden ser detectadas en la superficie de la piel usando electrodos Ag/AgCl [7]. En los últimos años, las señales EMG han encontrado diversas aplicaciones en el campo de la ingeniería biomédica: exoesqueletos y dispositivos robóticos para rehabilitación física [6, 9, 10]; electroestimulación funcional [5, 8]; teleoperación de dispositivos [15] y prótesis [1, 2, 4, 12, 13, 14].

Debido a que las señales EMG son el resultado de muchos factores fisiológicos, anatómicos y técnicos, es necesario su adecuado procesamiento. En particular, en las prótesis controladas mediante señales EMG, su éxito es medido por el desempeño en la ejecución de movimientos prácticos con la naturalidad de los ejecutados por la mano humana, para esto es esencial un proceso de análisis de las señales motoras EMG.

Las señales EMG superficiales (EMGS), son esencialmente un patrón unidimensional, por lo que cualquier técnica de procesamiento de señales para extracción de características y reconocimiento de patrones se puede aplicar a este tipo de señales. La tendencia en el control de prótesis a partir de señales EMGS obedece a que se constituye en la técnica más sencilla de implementar por su facilidad en la recolección sin intromisión directa sobre el organismo del usuario, remoción de electrodos y equipo para efectos de mantenimiento y/o calibración y su reutilización de una persona a otra [3].

Con respecto a las técnicas implementadas para la caracterización de las señales EMG y el reconocimiento de patrones, se ha reportado el uso de redes neuronales [2], transformada de Fourier y wavelets [4], algoritmos de machine learning [7], memorias asociativas extendidas [13], algoritmos de optimización paramétrica [14], métodos estadísticos para el reconocimiento de patrones [15], entre otros métodos tanto en el dominio del tiempo como de la frecuencia.

El objetivo de este trabajo es usar herramientas de software y hardware comerciales para adquirir señales EMG, realizar su procesamiento y clasificación, y usar dicha información para realizar el control de movimiento de una mano robótica con potenciales aplicaciones en prótesis activas de mano sin requerir demasiado desarrollo matemático o analítico, ofreciendo la posibilidad de que cualquier persona pueda reproducir éste tipo de proyectos. En particular, se usará el brazalete MYO[®], para la adquisición de las señales EMG, MATLAB[®] para el procesamiento de los datos, una tarjeta Arduino[®] para el control de los servomotores y código libre para la manufactura en impresión 3D de la mano robótica.

2. Metodología

La metodología usada para el desarrollo de este trabajo es la propuesta para el diseño universal (Fig. 1) [11].

Con base en esta metodología, la secuencia de actividades es la siguiente:

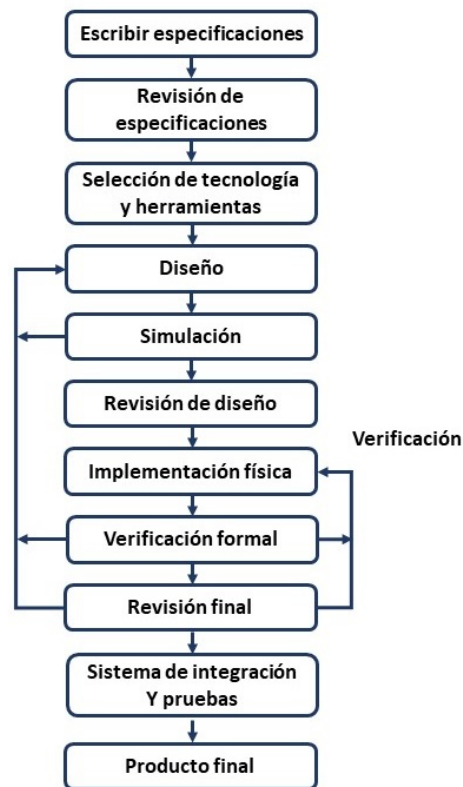


Fig. 1. Metodología del Diseño Universal

- Definición del problema: Se tiene una mano robótica impresa en 3D totalmente mecánica y se requiere que realice los movimientos de flexión-extensión de los dedos y agarre grueso, similares a los de una mano humana utilizando señales mioeléctricas para generar los movimientos.
- Identificar los usuarios y las necesidades: Se requiere generar movimientos de flexión-extensión de los dedos de una mano robótica a partir de señales EMG adquiridas con el brazalete MYO[®].
- Describir los requerimientos del producto y las especificaciones técnica: Implementar movimientos básicos en una mano robótica mediante software y hardware comerciales.

- Generación y selección de concepto: Considerar diferentes algoritmos para el reconocimiento de patrones en las señales EMG y distintas configuraciones y modos de actuación para la mano robótica.
- Desarrollo del concepto: Realizar adquisición en tiempo real de las señales EMG para el control de movimiento de la mano robótica. Clasificación de las características de las señales EMG para el entrenamiento del algoritmo de reconocimiento de patrones que permita la identificación de movimientos de la mano, interpretación y uso de señales para control.
- Diseño final: Integrar mano robótica impresa, tarjeta Arduino® y brazaletes MYO®.
- Evaluación del producto: La mano robótica y el brazo generan los movimientos seleccionados.

Para realizar la actividad de desarrollo de concepto que corresponde a la primera fase del diseño, se inicia por la adquisición y tratamiento de señales utilizando el brazaletes MYO® y MATLAB®.

2.1. Adquisición, clasificación y reconocimiento de patrones

Como se menciona en las hojas de especificaciones del brazaletes MYO®, éste contiene 3 acelerómetros y 3 giroscopios para la captura de los datos de espaciales de orientación y posición, además de 8 sensores para adquirir las señales EMG.

El sistema funciona leyendo las señales electromiográficas tomadas del antebrazo mientras se realiza cada uno de los gestos que se desee detectar. En este caso se reconocerán 6: movimiento de palma izquierda y derecha, puño, mano abierta, descanso y un gesto propuesto.

Como condiciones iniciales la duración del muestreo es de 20 segundos para cada gesto y se adquieren en total 4000 muestras para cada posición realizada. En la Fig. 2 se presentan las acciones realizadas para la identificación de movimiento de la mano a partir de las señales EMG del brazaletes MYO®.

Para la primera acción de la Fig.2, se usa el SDK de ThalmicLabsTM, que permite obtener datos del gesto para una o varias poses.

La adquisición de los datos de cada uno de los sensores de EMG del brazaletes se realiza a una frecuencia de 200 Hz durante el periodo que dure la prueba. Debido a que no se pueden enviar suficientes datos sobre una única característica, se separó en cuatro segmentos y se envían dos lecturas secuenciales en cada actualización. Eso significa que la forma de interpretar los datos es la siguiente:

```
EmgData0Characteristic
  Sample1
  Sample2
EmgData1Characteristic
  Sample3
  Sample4
```

En la Fig. 3 se muestra un ejemplo de datos adquiridos para un gesto de cerrar el puño. Para el movimiento que se desea reconocer se almacenan las señales de cada uno de los sensores EMG del brazaletes durante un tiempo de 20 segundos (para evitar sobre ajustes). Si existen pérdidas de datos en la transmisión, solamente existirán huecos en las matrices y se tendrán que adquirir nuevos datos. Posteriormente se reordena la información almacenada de tal forma que se tengan 333 muestras de 12 elementos (60 ms) para cada sensor.

Del vector que almacena la información ordenada se extraen las características que permiten diferenciar entre los datos obtenidos cada gesto realizado, utilizando el valor medio, la desviación estándar y un análisis de componentes principales, aunque para mayor precisión también puede usarse un análisis frecuencial de las señales o mediciones en su amplitud. Estos cálculos son realizados con código propio y la aplicación *Classification Learner* de MATLAB®.

Una vez establecido el vector de características, se construye el clasificador que involucra etapas como: la elección del modelo, el entrenamiento y verificación de los resultados.

Para tener la validación del sistema de reconocimiento de patrones se utiliza el método de

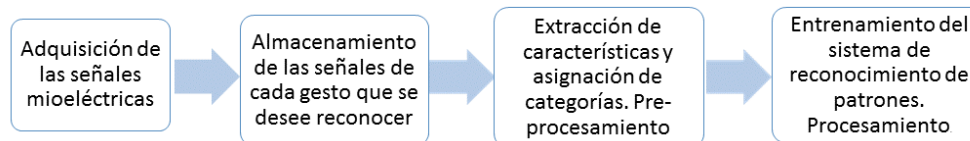


Fig. 2. Diagrama de flujo para la clasificación y el reconocimiento de patrones de las señales EMG

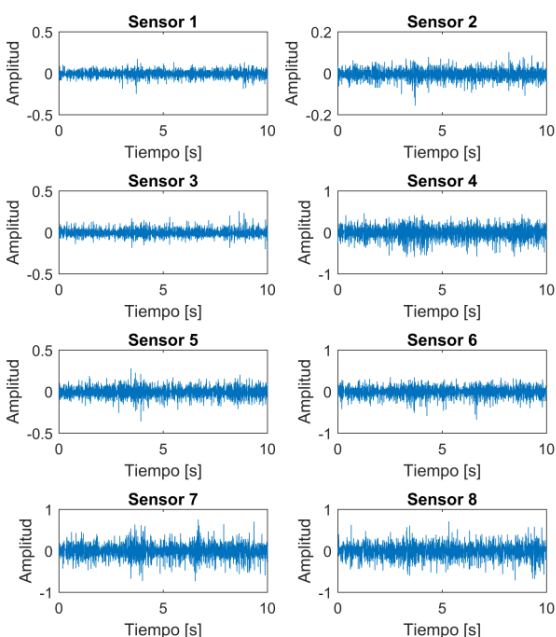


Fig. 3. Señales EMG adquiridas del brazalete MYO® para un movimiento de cerrar el puño

Holdout Validation, el cual consiste en designar un porcentaje de los datos para el entrenamiento del sistema y otra parte para validar su desempeño.

En la pantalla de *Classification Learner* de MATLAB® se hace la selección de predictores, respuestas y método de validación del clasificador.

Posteriormente se utiliza una máquina de soporte vectorial (SVM, por sus siglas en inglés) [1,2] para hacer la clasificación y el entrenamiento de los distintos núcleos, seleccionando el tipo específico como el Medium Gaussian, quadratic, cubic, etc. con base en restricciones que se dan por factores como el porcentaje de grasa en la piel, vello corporal o humedad. Por la información utilizada en este trabajo, el núcleo con mejor

desempeño fue Quadratic SVM, el cual tuvo una precisión de 99.7 por ciento (Fig. 4).

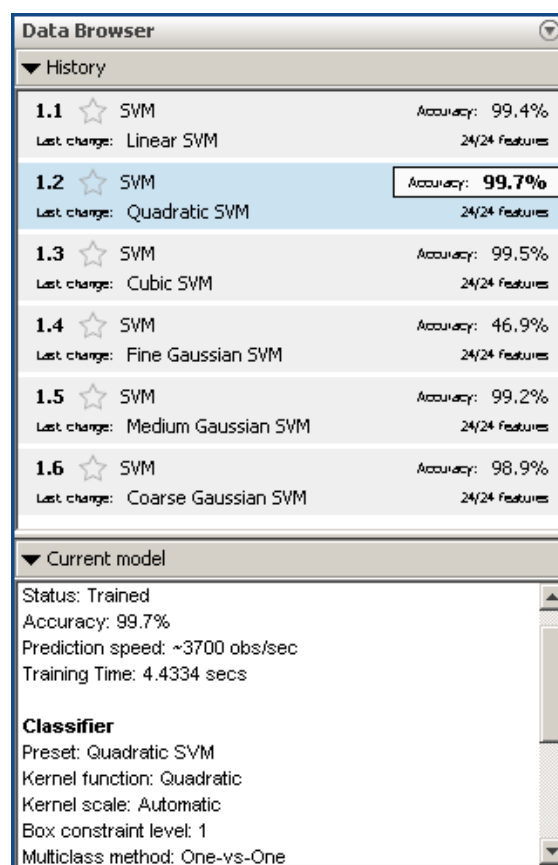


Fig. 4. Resultado del entrenamiento de distintos tipos de Kernels de máquinas de soporte vectorial y su precisión

2.2. Control de los servomotores en tiempo real

Una vez que se determina el gesto, se decide el número de grados que ha de girar cada actuador y, utilizando la tarjeta Arduino®, se envían las

señales a cada uno de los servomotores para que alcancen la posición deseada, usando una señal PWM.

En la Fig. 5 se muestra el diagrama de flujo para realizar el control en tiempo real de los servomotores mediante las señales EMG del brazalete MYO®.

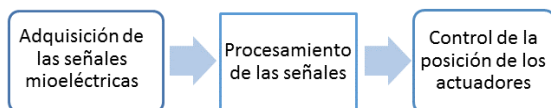


Fig. 5. Diagrama de flujo para el control de los servomotores aplicando señales EMG

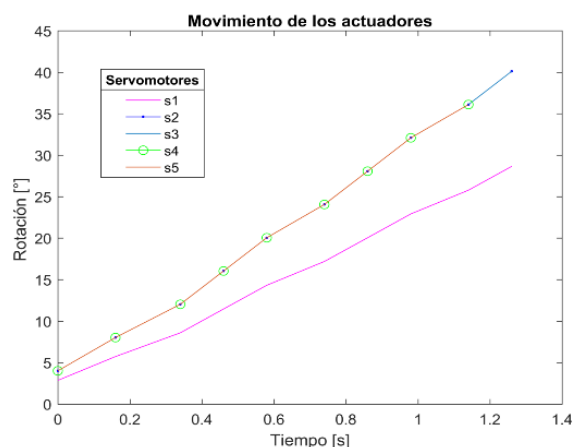


Fig. 6. Representación del movimiento de los actuadores durante la transición de la mano extendida al puño cerrado

2.3. Cuaterniones

Como se mencionó anteriormente, el brazalete MYO® tiene 3 acelerómetros y 3 giroscopios para obtener su posición y orientación en el espacio. Para los datos espaciales el SDK tiene:

1. *Datos de orientación:* En forma de cuaterniones y de ángulos de Euler.
2. *Datos puros de lectura del acelerómetro:* en forma de vector tridimensional.

3. *Datos puros de lectura del giroscopio:* en forma de vector tridimensional.

Los cuaterniones y los ángulos de Euler son diferentes representaciones para expresar la orientación de un cuerpo rígido en el espacio. Los tres componentes de los ángulos de Euler corresponden al ángulo vertical, al horizontal y al rotacional. Los cuaterniones representan la trayectoria más corta para llegar de una trayectoria a otra.

La posición (Yaw = 0, Pitch = 0, Roll = 0) es representada en cuaterniones por la forma $q = [0, 0, 0, 1]$.

Para convertir de cuaterniones a ángulos de Euler siguiendo la secuencia (yaw, pitch, roll) se tiene:

$$\begin{aligned}
 yaw &= \tan^{-1} \left(\frac{2(q_0q_1 + q_3q_2)}{q_3^2 - q_2^2 - q_1^2 - q_0^2} \right), \\
 pitch &= \sin^{-1} (-2(q_0q_1 - q_1q_3)) \\
 roll &= \tan^{-1} \left(\frac{2(q_1q_2 + q_0q_3)}{q_3^2 - q_2^2 - q_1^2 - q_0^2} \right). \quad (1)
 \end{aligned}$$

Si se quiere realizar esta conversión de manera gráfica, se puede utilizar la instrucción `quaternion2euler` de MATLAB®.

2.4. Integración de sistema de actuación con mano robótica

Para este caso de aplicación la mano robótica sujeta objetos a partir de 12cm x 12cm con masa no mayor a 500 gramos, los dedos se adaptan a las diferentes formas de los objetos para una mejor sujeción y poseen una rugosidad que permite sujetar objetos sin resbalse.

Las piezas de la mano son parte de un proyecto de robótica de código abierto denominado InMoov, creado por Gael Langevin. Los archivos `.stl` se encuentran disponibles en: <http://inmoov.fr/>.

Respecto a los requerimientos eléctrico-electrónicos se tiene que:

1. La prótesis debe ser accionada por señales mioeléctricas provenientes del músculo bíceps del brazo y del músculo Extensor Carpi Radialis Longus del antebrazo.

2. La fuente de alimentación debe ser externa.
3. Los actuadores deben ser servomotores con engranajes metálicos para una mejor transmisión de movimiento a los dedos de la prótesis.

Los requerimientos mínimos del sistema para generar un movimiento en la mano robótica son:

1. Una versión de MATLAB® 2015 o posterior, que incluya la aplicación *Classification Learner*.
2. Compilador de C++, se puede usar el que está incluido en Visual Studio® 2015.
3. MYO® SDK disponible¹ y MYO® Connect, para realizar la conexión entre el brazalete y la computadora.
4. Soporte para Arduino® en MATLAB®².

El código original de MATLAB® que se tomó como base para este trabajo se encuentra disponible³.

3. Resultados y discusión

Se reconocen 6 gestos de la mano del usuario: izquierda, derecha, puño, palma, descanso y propuesto. Procesando las señales adquiridas de estos gestos, clasificándolas y amplificándolas se obtienen las gráficas de las señales de control para los 5 servomotores que accionan los dedos de la mano robótica para generar movimientos específicos. Los servomotores s_1 , s_2 , s_3 , s_4 y s_5 mueven a los dedos pulgar, índice, medio, anular y meñique respectivamente. En ese orden se conectaron a los puertos digitales de la tarjeta Arduino® que tiene salidas PWM y permiten controlar su posición.

En la Fig. 6 se muestra la gráfica de rotación de cada motor con respecto al tiempo mientras se realiza el movimiento de cerrar el puño a partir

¹<https://developer.thalmic.com/downloads>

²<http://es.mathworks.com/hardware-support/arduino-matlab.html>

³<https://www.mathworks.com/matlabcentral/fileexchange/55817-myo-sdk-matlab>

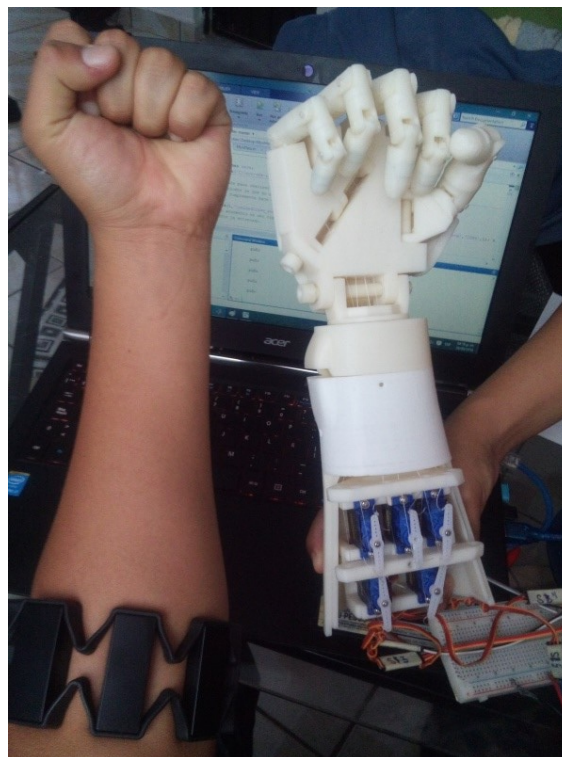


Fig. 7. La mano robótica imita el movimiento de cerrar el puño

de la mano completamente extendida. Aquí, los motores se mueven al mismo tiempo y el mismo número de grados, excepto que s_4 y s_5 que se detienen antes, y s_1 gira un menor número de grados.

En la Fig. 7 se muestra la mano robótica imitando el movimiento de cerrar el puño.

En la Fig. 8 se tiene la mano sosteniendo un multímetro para demostrar su capacidad de agarre y sujeción. Para visualizar el movimiento de la mano robótica con base en las señales EMG adquiridas, se puede consultar el video que se encuentra disponible⁴.

Además de las señales EMG, las mediciones de la orientación del brazalete, pueden ser también utilizadas como señales de control. En la Fig. 9, se muestran cuaterniones obtenidos de la pulsera MYO® así como las señales representadas por los ángulos de Euler.

⁴https://www.youtube.com/watch?v=PQqc_eikduQI



Fig. 8. Mano robótica sujetando un multímetro

Una aplicación de los cuaterniones en la mano robótica es el movimiento de pinza gruesa. Solo se mueven los dedos pulgar, índice y medio.

Para esta prueba, el movimiento de control considerado es la flexión-extensión del antebrazo (rotación alrededor del eje y), de tal forma que el movimiento de los dedos sea proporcional a la flexión del brazo para cerrar la pinza y a la extensión para abrirla.

En la parte inferior de la Fig. 10 se muestra la orientación del eje y llegando a 50 aproximadamente como máximo y el movimiento prácticamente nulo en los otros dos ejes, y en la parte superior se observa cómo los actuadores correspondientes a los dedos que generan el agarre se mueven indicando el inicio en 0 y su valor máximo en 130.

Los resultados de esta prueba se pueden visualizar en el video disponible en youtube⁵.

⁵https://www.youtube.com/watch?v=NMGF D_oo-ik&feature=youtu.be

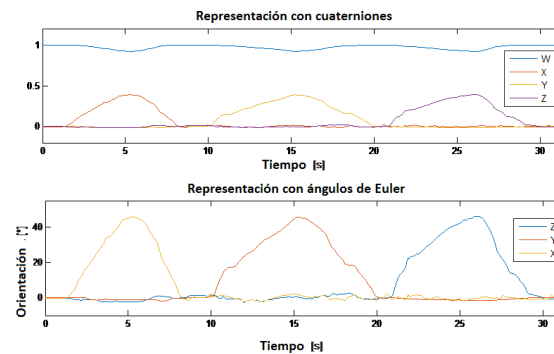


Fig. 9. Representación de señales de orientación en cuaterniones y en ángulos de Euler

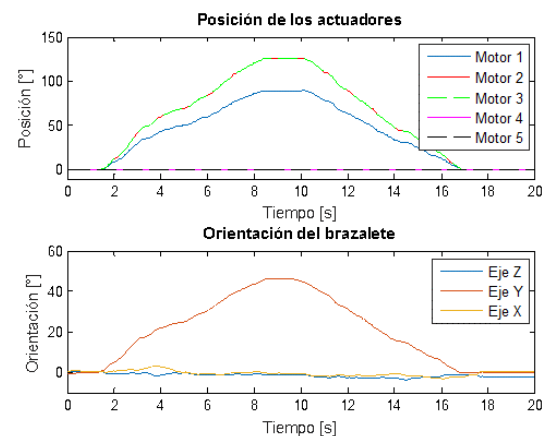


Fig. 10. Movimiento del brazo con el brazalete MYO® alrededor del eje y y movimiento de los actuadores correspondientes al dedo índice, medio y pulgar

La segunda prueba que se realizó con las señales de orientación del brazalete usando cuaterniones, fue que la mano robótica ejecutara la flexión consecutiva de los dedos a partir de la rotación del antebrazo alrededor del eje x .

En la parte inferior de la Fig. 11 se muestra la orientación del eje x llegando a 100 aproximadamente como máximo. En la parte superior de la misma figura se observa cómo los actuadores correspondientes a los cinco dedos se mueven de manera sucesiva en función del cambio en la orientación del brazo alrededor del

eje x . Los resultados de esta prueba se pueden observar en el video disponible en youtube⁶.

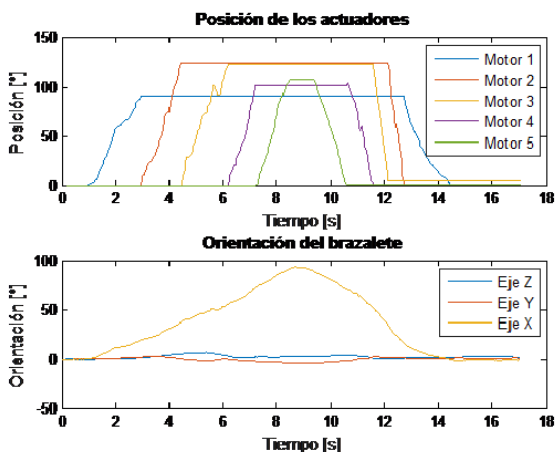


Fig. 11. Movimiento del brazo con el brazalete MYO[®] alrededor del eje x y movimiento de cada actuador para generar el movimiento de flexión sucesiva de cada uno de los dedos

4. Conclusiones

El desarrollo mecánico de la prótesis y del programa en MATLAB[®] para la adquisición e interpretación de las señales electromiográficas se desarrolló aplicando código abierto disponible en internet, lo que permite el diseño y construcción del prototipo en un tiempo corto. Al código para la adquisición y manipulación de señales se agregaron líneas de programación que permiten identificar diferentes gestos de los propuestos por los fabricantes del brazalete MYO[®]. Esta programación se subirá a la misma página de programación de MYO[®] para apoyo de los usuarios.

La unión de estos dos programas permitió el desarrollo de una mano robótica que imita los movimientos humanos generando las bases para desarrollar una prótesis real. Como trabajos futuros se plantea el rediseño para el incremento de fuerza en la prótesis y para generar los

movimientos finos de la mano. Éste proyecto es multidisciplinario y es necesaria la participación de un especialista para desarrollar prototipos personalizados cumpliendo con la ergonomía y las necesidades del paciente.

También se demostró que el prototipo presentado es de bajo costo y fácil adquisición, lo que permite a los usuarios generar e implementar diversos diseños a partir de código abierto modificado.

Agradecimientos

Agradecemos al Consejo Nacional de Ciencia y Tecnología (CONACyT) por el apoyo brindado a este proyecto a través del programa de Cátedras CONACyT.

Referencias

1. Al-Timemy, A., Khushaba, R., Bugmann, G., Escudero, J. (2016). Improving the performance against force variation of emg controlled multifunctional upper-limb prostheses for transradial amputees. *IEEE Transactions on Neural Systems and Rehabilitations Engineering*, Vol. 24, No. 6, pp. 650–661.
2. Amsuss, S., Goebel, P. M., Jiang, N., Graimann, B., Paredes, L., Farina, D. (2014). Self correcting pattern recognition system of surface emg signals for upper limb prosthesis control. *IEEE Transactions on Biomedical Engineering*, Vol. 61, No. 4, pp. 1167–1176.
3. de Luca, C. J. (1997). The use of surface electromyography in biomechanics. *Journal of Applied Biomechanics*, Vol. 13, No. 2, pp. 135–163.
4. Geethanjali, P., Ray, K. K. (2015). A low-cost real-time research platform for emg pattern recognition-based prosthetic hand. *IEEE/ASME Transactions on Mechatronics*, Vol. 20, No. 4.
5. Hara, Y. (2013). Rehabilitation with functional electrical stimulation in stroke patients. *International Journal of Physical Medicine and Rehabilitation*, Vol. 1, No. 6, pp. 1–6.
6. Hu, X.-L., Tong, R. K., Newmen, H., Xue, J., Rong, W., Li, L. (2015). Wrist rehabilitation assisted by and electromyography-driven neuromuscular electrical stimulation robot after stroke. *Neurorehabilitation and Neural Repair*, Vol. 29, No. 8, pp. 1–10.

⁶<https://www.youtube.com/watch?v=p7cj9dkQMyg&feature=youtu.be>

7. **Karlik, B. (2014).** Machine learning algorithms for characterization of emg signals. *International Journal of Information and Electronics Engineering*, Vol. 4, No. 3, pp. 189–194.
8. **Kim, Y. (2013).** The effects of emg-triggered functional electrical stimulation on upper extremity function in stroke patients. *Physical Therapy Rehabilitation Science*, Vol. 2, No. 1, pp. 1–6.
9. **Lu, Z., Tong, K.-y., Shin, H., Li, S., Zhou, P. (2017).** Advanced myoelectric control for robotic hand-assisted training: outcome from a stroke patient. *Frontiers in Neurology*, Vol. 8, No. 107, pp. 1–5.
10. **Lyu, M., Chen, W.-H., Ding, X., Wang, J., Pei, Z., Zhang, B. (2019).** Development of an emg-controlled knee exoskeleton to assist home rehabilitation in a game context. *Frontiers in Neurorobotics*, Vol. 13, No. 67, pp. 1–15.
11. **Martin, B., Hanington, B., Hanington, B. (2012).** *Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions.* Rockport Publishers.
12. **Parajuli, N., Sreenivasan, N., Bifulco, P., Cesarelli, M., Savino, S., Niola, V., Esposito, D., Hamilton, T. J., Naik, G. R., Gunawardana, U., Gargiulo, G. D. (2019).** Real-time emg based pattern recognition control for hand prostheses: a review on existing methods, challenges and future implementation. *Sensors*, Vol. 19, No. 4596, pp. 1–30.
13. **Sanchez-Velasco, L. E., Arias-Montiel, M., Guzman-Ramirez, E., Lugo-Gonzalez, E. (2020).** A low-cost emg-controlled anthropomorphic robotic hand for power and precision grasp. *Biocybernetics and Biomedical Engineering*, Vol. 40, No. 1, pp. 221–237.
14. **Vidovic, M. M., Hwang, H.-J., Amsuss, S., Hahne, J. M., Farino, D., Muller, K.-R. (2016).** Improving the robustness of myoelectric pattern recognition for upper limb prostheses by covariate shift adaptation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 24, No. 9, pp. 961–970.
15. **Vogel, J., Bayer, J., van der Smagt, P. (2013).** Continuous robot control using surface electromyography of atrophic muscles. *Proceedings of 2013 IEEE/RJS International Conference on Intelligent Robots and Systems*, IEEE, pp. 845–850.

*Article received on 12/08/2020; accepted on 25/03/2021.
Corresponding author is Manuel Arias-Montiel.*