

Jornadas de Automática

Control y virtualización de un mano robótica RH8D mediante ROS

Martínez-García, Ana María^a, Sánchez, Daniel^a, Úbeda, Andrés^{a,*}, Jara, Carlos Alberto^a, Morell, Vicente^a

^aHuman Robotics Group, Universidad de Alicante, Crta. San Vicente del Raspeig S/N, 03690, Alicante, España.

To cite this article: Martínez-García, A.M., Sánchez, D., Úbeda, A., Jara, C.A., Morell, V. 2025. Control and virtualization of a RH8D robotic hand using ROS. *Jornadas de Automática*, 46.
<https://doi.org/10.17979/ja-cea.2025.46.12233>

Resumen

Este trabajo presenta el desarrollo de un sistema de control híbrido para la mano robótica RH8D (Seed Hand), integrando distintas formas de interacción humano-máquina. El sistema incluye un control manual desde una interfaz gráfica creada en MATLAB combinado con un sistema de control mioeléctrico. Para ello, se emplea una arquitectura distribuida soportada por ROS, lo que permite una comunicación eficiente y flexible entre dispositivos y plataformas. Además, se ha implementado un gemelo digital de la mano en Unity que replica en tiempo real los movimientos de la mano física a partir de los datos recibidos mediante ROS. Esta virtualización proporciona un valor añadido al sistema, ofreciendo capacidades de supervisión remota y simulación previa. La propuesta sirve de base a futuros desarrollos en el ámbito del entrenamiento virtual del control mioeléctrico de prótesis robóticas.

Palabras clave: Robótica y Control, Sistemas Biológicos y Médicos, Sistemas Mecatrónicos

Control and virtualization of a RH8D robotic hand using ROS

Abstract

This work presents the development of a hybrid control system for the RH8D robotic hand (Seed Hand), integrating different forms of human-machine interaction. The system includes hand control from a graphical interface created in MATLAB combined with a myoelectric control system. To achieve this, a distributed architecture supported by ROS is used, enabling efficient and flexible communication between devices and platforms. Furthermore, a digital twin of the hand has been implemented in Unity that replicates the movements of the physical hand in real time based on data received via ROS. This virtualization provides added value to the system, offering remote monitoring and pre-simulation capabilities. The proposal serves as a basis for future developments in the field of virtual training of myoelectric control of robotic prostheses.

Keywords: Robotics and Control, Biological and Medical Systems, Mechatronic Systems.

1. Introducción

El uso de manos robóticas en numerosos campos de la ingeniería ha supuesto un avance muy importante en aspectos como la manipulación diestra en el ámbito industrial (Weinberg et al., 2024) pero también en el ámbito médico en forma de prótesis sustitutivas de amputaciones de miembro superior (Mendez et al., 2021). En este contexto, la prótesis robótica en general requiere de herramientas que permitan mejorar la adherencia del usuario final al dispositivo, ya que en muchos

casos, las personas con amputación prefieren una alternativa pasiva frente a dispositivos actuados.

La realidad virtual es una alternativa especialmente útil en el contexto de la prótesis robótica (Joyner et al., 2021). En este caso, algunos de los dispositivos actuados utilizan alternativas de control basadas en el uso de electromiografía (Geethanjali, 2016). Muchos de estos algoritmos requieren de bastante entrenamiento para poder ser empleados con eficacia, por lo que la virtualización de la prótesis para generar herramientas

*Autor para correspondencia: andres.ubeda@ua.es
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

virtuales que puedan incorporar elementos realistas o, incluso, componentes gamificados, es un complemento ideal para ese entrenamiento, especialmente en el caso de menores de edad (Carro et al., 2022).

Existen varios estudios que han explorado la virtualización de manos robóticas en el contexto del entrenamiento protésico. Por ejemplo, en (Chappell et al., 2022), se evalúa el uso de una prótesis virtual en la realización de tareas de *pick and place* como el *Block and Box Test* o con objetos de la vida diaria, incorporando un control mioeléctrico muy sencillo. En este aspecto, el entrenamiento del control mioeléctrico ha sido explorado en otros trabajos en más detalle incorporando entornos virtuales, en este caso usando un brazo robot en la manipulación de objetos cotidianos (Polo-Hortigüela et al., 2024). Otro estudio analiza el uso de digital twins en el ámbito protésico para evaluar distintos estándares de validación como el Southampton Hand Assessment Procedure (SHAP) (Cellupica et al., 2024). Además, el uso en robótica de herramientas como Robot Operating System (ROS) para generar frameworks escalables y compatibles con numerosos tipos de dispositivos, permite disponer de entornos virtuales mucho más personalizables y flexibles (Singh et al., 2024).

Este trabajo propone el desarrollo de un sistema de control híbrido para la mano robótica RH8D (Seed Hand), integrando distintas formas de interacción humano-máquina. La propuesta combina el control manual desde una interfaz gráfica creada en MATLAB con un sistema de control mioeléctrico. Los dos métodos están conectados a través de una arquitectura distribuida soportada por ROS, lo que permite una comunicación eficiente y flexible entre dispositivos y plataformas. Además, en Unity se ha implementado un gemelo digital de la mano, el cual replica en tiempo real los movimientos de la mano física a partir de los datos recibidos mediante ROS. Esta virtualización proporciona un valor añadido al sistema, ofreciendo capacidades de supervisión remota y simulación previa. Esta propuesta se ha diseñado con el fin de establecer una base de desarrollo a futuros entornos eficaces de entrenamiento del control mioeléctrico de prótesis robóticas.

2. Framework de virtualización y control

2.1. Mano robótica RH8D Seed Hand

La RH8D es una mano robótica diestra de tamaño adulto, inspirada en la mano humana, desarrollada por Seed Robotics (Figura 1).

Tiene 19 grados de libertad, entre los que se incluyen una articulación de muñeca completamente esférica y un pulgar oponible, junto con 8 mecanismos inteligentes para un control preciso. Se basa en la mano humana para realizar los agares de mayor importancia, y sus dedos, que constan de tres segmentos, están gestionados por actuadores inteligentes. El objetivo del diseño subaccionado es lograr el equilibrio adecuado entre el control exacto y la conformidad con la forma de los objetos.

Todos los actuadores proporcionan control y retroalimentación en tiempo real acerca de la posición, velocidad y medición de la corriente (con dirección), lo que simplifica la deducción de la fuerza aplicada. Además, cuenta con un grupo

de sensores que proporcionan datos adicionales como la temperatura del actuador, la distancia de los objetos a la palma de la mano y el estado de carga y PWM.

Cuenta con una capacidad de carga útil de hasta 2.5 kilogramos (kg) cuando realiza una tracción vertical, y puede manipular objetos de hasta 1 kg en movimientos dentro del espacio tridimensional. A pesar de su capacidad, mantiene un diseño compacto con un peso total de 620 gramos.

Además, cuenta con modelos URDF para su uso en simulación, protocolo serie de bajo nivel (UART) y paquetes ROS de código abierto.



Figura 1: RH8D Adult Hand.

2.2. Interfaz de usuario

Con el objetivo de facilitar la interacción con la mano robótica, se ha desarrollado una interfaz gráfica en MATLAB que permite al usuario controlar los movimientos de la mano de forma manual e intuitiva.

Tal y como puede verse en la figura 2, esta interfaz incluye distintos botones:

- Por un lado, en color rosa se encuentran los botones que muestran por terminal información sobre el estado de las articulaciones y de los sensores de la mano. Tanto de los de fuerza de los dedos como del sensor de distancia de la palma.
- Por otro lado, en naranja se ubican aquellos que se emplean para activar el control mioeléctrico y que muestran información de los sensores pero mediante un gráfico de barras.
- En azul están aquellos que tienen unas acciones predefinidas. Los que se encuentran en la zona de arriba a la derecha hacen que la mano realice un determinado gesto, que viene definido por el nombre del botón. Más abajo, se encuentra el de send, que se encarga de enviar las posiciones y velocidades objetivo que se hayan definido en la tabla. Finalmente, los botones de grasp y drop permiten agarrar y soltar objetos.
- Por último, en verde queda definido el botón de start, que será el que inicia la comunicación.

Además de estos botones, se incluye un gráfico de barras que muestra el valor de los sensores de fuerza de los dedos de la mano durante la operación de agarre o si el botón “force graph” es pulsado.

Así, MATLAB se establece como un nodo cliente en la red ROS, lo que permite enviar comandos directamente a los topics correspondientes y recibir información del estado de las articulaciones.



Figura 2: Interfaz para un control manual en MATLAB

2.3. Virtualización de la mano

Para complementar el control físico de la mano robótica RH8D, se ha creado una representación virtual de esta a través de Unity. Este gemelo digital muestra en tiempo real los movimientos y configuraciones de la mano física.

El primer paso consiste en integrar ROS con Unity. Para ello se hace uso de los paquetes “ROS TCP Connector” y “ROS TCP Endpoint” que van a permitir que Unity pueda suscribirse y publicar en los topics ROS. Posteriormente, se lleva a cabo la generación de mensajes en Unity y la exportación del modelo URDF de la mano.

Finalmente, se genera un script que permite controlar el modelo del robot en Unity usando la información recibida desde ROS. Para ello, se establece una conexión mediante ROS-Connection y se suscribe al topic R.Joints, que recibe mensajes del tipo AllJointsMsg con las posiciones de todas las articulaciones. Dado que los nombres de las articulaciones en el URDF y en ROS no coinciden, se implementa un diccionario de mapeo que asocia cada nombre recibido con su correspondiente objeto en la jerarquía del modelo 3D. Al recibir un mensaje, el script localiza cada articulación, convierte los valores de posición (en el rango 0-4095) a ángulos en radianes adecuados para Unity, y los aplica al modelo físico, permitiendo una simulación en tiempo real sincronizada con el estado de la mano física.

Lo que se busca con esta virtualización es conseguir una herramienta que permita supervisar el comportamiento de la mano a distancia, validar comandos antes de su ejecución en el hardware real, detectar posibles errores en la transmisión de comandos y proporcionar entornos de simulación seguros para pruebas y demostraciones.

En la figura 3 se muestra la integración con Unity reflejando la capacidad de replicar con precisión los gestos y comandos ejecutados por la mano robótica.



Figura 3: Visualización en Unity

2.4. Comunicación entre sistemas

El sistema creado se basa en una arquitectura de comunicación distribuida entre varios dispositivos y plataformas de software, que se conectan mediante el middleware ROS, con la finalidad de controlar la mano robótica R8HD.

La mano robótica se conecta al ordenador que ejecuta el maestro ROS, gestionando la comunicación de bajo nivel y los nodos necesarios para el control de la mano. Desde MATLAB, mediante su toolbox de robótica, se establece la conexión como cliente ROS, lo que posibilita enviar y recibir comandos y estados de la mano.

En el proceso de virtualización, Unity desempeña el papel de suscriptor ROS. De este modo, la información de posición de las articulaciones publicada por los nodos de ROS se recibe en Unity, permitiendo representar en tiempo real los movimientos físicos de la mano en un entorno virtual.

Este enfoque de comunicación permite que cada sistema opere en su entorno óptimo y mantenga una sincronización robusta a través de una arquitectura híbrida de ROS.

En la figura 4 se presenta un esquema de dicha comunicación.

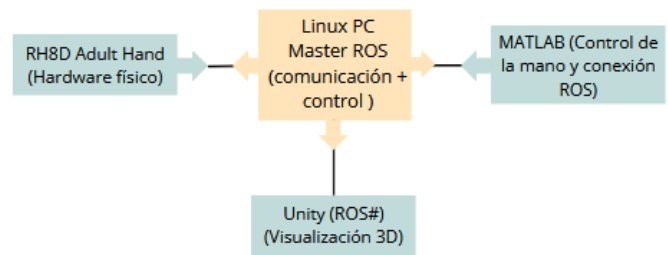


Figura 4: Esquema de comunicación

3. Control mioeléctrico

La implementación del control mioeléctrico se llevó a cabo utilizando el sistema Noraxon MiniDTS que dispone de 4 sensores bipolares y permite la adquisición de señales electromiográficas a 1500 Hz de frecuencia. Este sistema dispone del software MR3, el cual facilita tanto el registro como el filtrado eficiente de las señales captadas por los electrodos.

Se trata de un control mioeléctrico de tipo on/off en el que, mediante el establecimiento de rangos de señal EMG, fue posible distinguir entre tres estados musculares: flexión, extensión y reposo. Para ello, desde el software MR3 se rectifica la

señal EMG y se obtiene la envolvente. Esta señal es enviada al software diseñado mediante un protocolo HTTP Streaming en local.

Con el objetivo de simplificar el uso de este control, se colocaron electrodos en el antebrazo derecho para captar el estado de flexión y en el antebrazo izquierdo para detectar el estado de extensión. La condición de reposo se definió como la ausencia de contracción muscular voluntaria en ambos brazos.

Durante las pruebas, los usuarios se encontraban de pie, con los brazos suspendidos en el aire, sin estar apoyados. Cada gesto se mantenía durante un período de entre 1 y 2 segundos, lo que permitía registrar señales consistentes para establecer los umbrales de activación en el software MR3.

Esta estrategia permitió desarrollar un control sencillo pero funcional que habilita la ejecución de diferentes comportamientos en la mano robótica. En particular, se estableció una correspondencia entre cada estado muscular y un gesto específico realizado por la mano:

- El estado de flexión activa el gesto "perfect", como se observa en la figura 5.



Figura 5: Gesto asociado al estado de flexión

- El estado de extensión activa el gesto "peace", tal como se muestra en la figura 6.



Figura 6: Gesto asociado al estado de extensión

- El estado de reposo mantiene la mano abierta, como se aprecia en la figura 7.



Figura 7: Gesto asociado al estado de reposo

4. Conclusion y trabajos futuros

Este proyecto desarrolla un sistema completo para controlar la mano robótica RH8D combinando distintos modos de interacción: desde una interfaz manual en MATLAB, hasta un control mioeléctrico, todo ello conectado mediante ROS. Además, la implementación de una representación digital en Unity facilita la visualización en tiempo real de los movimientos de la mano, lo que ofrece nuevas oportunidades en simulación y teleoperación.

Aunque la mano RH8D fue creada para propósitos de investigación, el resultado final es una solución versátil cuyo uso en contextos experimentales evidencia su capacidad para un extenso espectro de usos futuros, entre los que sobresalen:

- Desarrollo de tecnologías asistivas: MATLAB, con su interfaz intuitiva y control mioeléctrico, puede ajustarse a las necesidades de individuos con movilidad limitada. Esta tecnología posibilita la exploración de nuevas maneras de interactuar con dispositivos protésicos, simplificando la creación de soluciones a medida en el sector de la asistencia funcional.
- Formación y educación: se trata de un proyecto ideal para impartir conocimientos avanzados en robótica, el procesamiento de señales biomédicas y la comunicación multiplataforma. Su integración con ROS y el uso de control mioeléctrico lo hacen especialmente útil para prácticas en universidades o centros de formación tecnológica.

Para ello, se propone incorporar en el futuro algoritmos de control mioeléctrico más complejos que permitan manejar un mayor número de grados de libertad de la mano robótica. Además, el sistema debe someterse aún a un análisis riguroso de retardos de comunicación, que son claves en el control eficiente de la mano robótica virtual, sobre todo, si se incorporan los algoritmos de control avanzado citados.

Agradecimientos

Este trabajo ha sido apoyado por la ayuda de formación predoctoral en colaboración con empresas (UAIND21-06B) y financiado por el proyecto MYOREHAB (PCI2023-

143405), *Ministerio de Ciencia, Innovación y Universidades y la Agencia Estatal de Investigación* [MCI-N/AEI/10.13039/501100011033].

Referencias

- Carro, R., Costales, F., A., O., 2022. Serious games for training myoelectric prostheses through multi-contact devices. *Children* 9 (3), 423.
DOI: 10.3390/children9030423
- Cellupica, A., Cirelli, M., Saggio, G., Valentini, P., 2024. An interactive digital-twin model for virtual reality environments to train in the use of a sensorized upper-limb prosthesis. *Algorithm* 17, 35.
DOI: 10.3390/a17010035
- Chappell, D., Son, H., Clark, A., Yang, Z., Bello, F., Kormushev, P., Rojas, N., 2022. Virtual reality pre-prosthetic hand training with physics simulation and robotic force interaction. *IEEE Robotics and Automation Letters* 7 (2), 4550–4557.
DOI: 10.1109/LRA.2022.3151569
- Geethanjali, P., 2016. Myoelectric control of prosthetic hands: state-of-the-art review. *Medical Devices: Evidence and Research* 27 (9), 247–255.
DOI: 10.2147/MDER.S91102
- Joyner, J., Vaughn-Cooke, M., Benz, H., 2021. Comparison of dexterous task performance in virtual reality and real-world environments. *Frontiers in Virtual Reality* 2, 599274.
DOI: 10.3389/frvir.2021.599274
- Mendez, V., Iberite, F., Shokur, S., Micera, S., 2021. Current solutions and future trends for robotic prosthetic hands. *Annual Review of Control, Robotics, and Autonomous Systems* 4, 595–627.
DOI: 10.1146/annurev-control-071020-104336
- Polo-Hortigüela, C., Maximo, M., Jara, C., Ramon, J., Garcia, G., Ubeda, A., 2024. A comparison of myoelectric control modes for an assistive robotic virtual platform. *Bioengineering* 11, 473.
DOI: 10.3390/bioengineering11050473
- Singh, M., Kapukotuwa, J., Gouveia, E., Fuenmayor, E., Qiao, Y., Murry, N., Devine, D., 2024. Unity and ros as a digital and communication layer for digital twin application: Case study of robotic arm in a smart manufacturing cell. *Sensors* 24 (17), 5680.
DOI: 10.3390/s24175680
- Weinberg, A., Shirizly, A., Azulay, O., Shintov, S., 2024. Survey of learning-based approaches for robotic in-hand manipulation. *Frontiers in Robotics and AI* 11, 1455431.
DOI: 10.3389/frobt.2024.1455431