

Innovación, Control y Acceso en el ROS Meetup

Introducción: La Robótica como Puente entre Investigación y Comunidad

El ROS Meetup Bogotá 2025 se consolidó como un punto neurálgico para la convergencia de la academia, la industria y la comunidad en torno al Robot Operating System (ROS y ROS 2), reflejando las tendencias globales y abordando los desafíos locales en el ecosistema robótico latinoamericano. Las ponencias presentadas en el evento trazaron una hoja de ruta clara, demostrando que la innovación no se limita únicamente a la investigación de vanguardia, como la robótica médica y el control de humanoides, si no que se extiende fundamentalmente a la superación de barreras de acceso y al fomento de una cultura de desarrollo abierto. El análisis de las charlas revela tres ejes temáticos interconectados: la aplicación de sistemas robóticos en entornos críticos (salud), los fundamentos de la ingeniería de control para la autonomía avanzada (humanoides), y la imperante necesidad de democratizar el conocimiento de ROS para impulsar el crecimiento regional. Este ensayo examina cómo las contribuciones del Meetup abordan estos pilares, redefiniendo el papel de la robótica en la transformación tecnológica de la región.

I. La Frontera de la Robótica Médica: Integración y Seguridad

La presentación de Arvind Kumar sobre la integración de 3D Slicer y ROS 2 (Slicer ROS 2) puso en relieve uno de los campos más exigentes y de mayor impacto social de la robótica: las cirugías mínimamente invasivas. Este trabajo representa un avance crucial al fusionar la precisión de la planificación quirúrgica con la capacidad de ejecución de los sistemas robóticos.

3D Slicer, una plataforma de visualización y procesamiento de imágenes médicas, es fundamental para la segmentación de órganos, la planificación de trayectorias y la definición de las zonas de intervención. Su integración con ROS 2 —la versión moderna y robusta del *middleware* robótico— permite que los planes generados en el ámbito virtual se traduzcan en comandos de movimiento y control en tiempo real para el robot quirúrgico. ROS 2, con su arquitectura de comunicación distribuida y enfocado en la latencia baja, es el sistema operativo idóneo para manejar los estrictos requisitos de tiempo real y seguridad que exige un quirófano.

El componente más innovador y críticamente relevante de esta ponencia fue la implementación de retroalimentación háptica basada en el concepto de "regiones prohibidas". En un procedimiento, se define una región de seguridad alrededor de estructuras críticas (como un tumor o vasos sanguíneos vitales). Si el efector final del robot (la herramienta quirúrgica) se acerca demasiado a esta zona de peligro —una violación detectada por el *software* de planificación en 3D Slicer—, el sistema activa una señal háptica. Esta retroalimentación no solo actúa como una advertencia visual o sonora, sino que genera una resistencia física en los controles que manipula el cirujano. Esta interacción hombre-máquina, mediada por ROS, dota al cirujano de un sentido de "freno virtual", previniendo errores de alta consecuencia. Esta solución subraya que la robótica médica del futuro no solo busca la autonomía total, sino la mejora continua de la colaboración humano-robot, donde la máquina complementa la pericia humana con capacidades de precisión y monitoreo imposibles de mantener por la atención humana continua. El éxito de Slicer ROS 2 radica, por lo tanto, en su capacidad para ofrecer un *framework* abierto que cumple con la promesa de procedimientos más seguros, controlados y, en última instancia, menos invasivos para el paciente.

II. El Desafío de Control en Sistemas Humanoides

La presentación de Gabriel Díaz sobre los Desafíos del Control de Posición en Robots Humanoides Simulados trasladó el foco de la aplicación crítica a los cimientos de la teoría de control y la dinámica compleja. Los robots humanoides, en particular los bípedos, son considerados uno de los desafíos más grandes de la ingeniería robótica debido a su morfología inherentemente inestable, imitando la locomoción humana que requiere un equilibrio dinámico constante.

Gabriel destacó que la simulación es una etapa fundamental, no opcional, en el desarrollo de estos sistemas. Plataformas como Gazebo permiten la experimentación con modelos cinemáticos y dinámicos complejos sin el riesgo de daños físicos al hardware, una consideración económica y práctica crucial dada la alta complejidad y costo de los robots humanoides. La simulación no es solo una representación visual; es un campo de pruebas para estrategias de control.

Los retos principales identificados son:

1. **Morfología Inestable:** A diferencia de los robots con una base estática o de los cuadrúpedos, el centro de masa de un bípedo se encuentra constantemente fuera de la base de soporte durante el movimiento, requiriendo un ajuste continuo para mantener el equilibrio dinámico.
2. **Independencia y Coordinación Articular:** Los movimientos de una pierna o brazo deben coordinarse con el resto del cuerpo en tiempo real para preservar el balance, lo que exige un sistema de control distribuido y altamente eficiente.
3. **Control Robusto ante el Error:** El ponente enfatizó la necesidad de un "control de error de error" que pueda manejar perturbaciones y fallos inesperados.

Las estrategias de control discutidas (control directo y PID simple) son puntos de partida, pero la conclusión del ponente resuena como un principio fundamental: "un control estable no solo mueve un robot, sino que lo mantiene de pie". Esto va más allá de la simple cinemática; implica la implementación de modelos de dinámica de cuerpo rígido y la estimación de estados avanzados para lograr un control compuesto e integrado que maneje la inercia, la fricción y las fuerzas externas. El trabajo en simulación, por tanto, se convierte en la herramienta indispensable para refinar estos algoritmos de control antes de la implementación en hardware real, asegurando la robustez y la capacidad de adaptación en entornos no estructurados.

III. Rossblocks

El análisis de la industria robótica en América Latina reveló una brecha significativa: un crecimiento regional del 1.11% frente a la proyección global del 14.7%. Esta disparidad puso de manifiesto que las barreras de entrada al desarrollo en ROS son demasiado altas, especialmente para estudiantes y desarrolladores jóvenes. La propuesta de Rossblocks emerge como un pilar en la estrategia de democratización.

La complejidad de la configuración inicial de un entorno ROS 2 (instalación de Ubuntu, manejo de contenedores Docker, configuración de paquetes) es un obstáculo conocido que desanima a los recién llegados. Rossblocks aborda esto mediante una propuesta enfocada en la facilidad de uso y la accesibilidad web.

Rossblocks opera en dos niveles cruciales:

1. Reducción de la Fricción Técnica: Mediante una arquitectura basada en AWS (Cloudfront, Enginex) y una interfaz frontend desarrollada con Angular, el sistema despliega contenedores serverless personalizados de ROS 2 para cada usuario. Esto elimina la necesidad de que los usuarios realicen complejas instalaciones locales, permitiendo que la programación comience inmediatamente desde cualquier navegador web. El acceso instantáneo al entorno de desarrollo mitiga el problema de la configuración y estandariza la experiencia de aprendizaje.
2. Accesibilidad Educativa con Programación de Bloques: Rossblocks adopta el paradigma de programación visual, similar a Scratch o Blockly, permitiendo a los usuarios juveniles y novatos construir lógica robótica arrastrando y soltando bloques. Esta aproximación abstracta simplifica la sintaxis y permite a los usuarios centrarse en la lógica algorítmica y los conceptos de ROS (nodos, tópicos, mensajes) antes de enfrentarse a la complejidad del código C++ o Python.

El proceso de desarrollo culmina en la capacidad de exportar el *workspace* de bloques a código ROS 2 local, cerrando el ciclo entre el aprendizaje visual simplificado y el desarrollo profesional. Esta iniciativa no solo busca aumentar la cantidad de programadores en la comunidad, sino también fomentar la transición fluida del concepto visual a la codificación tradicional, sentando las bases para el crecimiento futuro de la robótica en la región y acelerando la curva de aprendizaje de la próxima generación de ingenieros.

IV. Robot

Las ponencias y la demostración de interacción humana con el robot Pepper ilustraron la profunda integración entre la Inteligencia Artificial (y la robótica, enfocándose en la autonomía y la interacción natural).

Por un lado, se abordó el Aprendizaje por Refuerzo (Reinforcement Learning - RL), destacándolo como una técnica crucial para entrenar modelos robóticos complejos, especialmente en escenarios donde los datos etiquetados son escasos (a diferencia del aprendizaje supervisado). El RL, al utilizar funciones de recompensa y penalizaciones, permite que el robot aprenda a realizar tareas complejas (como la locomoción estable de un cuadrúpedo en diversos entornos) a través de ensayo y error en un entorno simulado (utilizando herramientas como Isaac Sim e Isaac Lab). El enfoque biológico, donde el cuadrúpedo se inspira en el comportamiento y la dinámica de un perro, resalta la búsqueda de soluciones robustas y versátiles derivadas de la naturaleza. La implementación exitosa de RL es esencial para que los robots desarrollen comportamientos adaptativos sin una programación explícita para cada posible escenario, llevándonos a una verdadera autonomía en entornos no estructurados.

Por otro lado, la demostración con el robot Pepper y los esfuerzos en el proyecto Orion mostraron el avance en la Interacción Humano-Robot (HRI). La utilización de modelos de Procesamiento de Lenguaje Natural (NLP) dota al robot de:

1. Comprensión Contextual: El robot puede entender comandos de voz complejos y mantener un contexto conversacional, recordando el nombre del usuario o el entorno en el que se encuentra.
2. Respuestas Comportamentales: El sistema puede traducir peticiones verbales a acciones físicas o de interfaz (como cambiar el color de sus ojos o realizar un gesto).

La HRI natural y fluida es la clave para la aceptación social y la utilidad práctica de los robots de servicio y asistencia. La capacidad de los robots, incluso en entornos de laboratorio, de reaccionar a

la voz con matices contextuales y de ejecutar tareas complejas (como la navegación o el control de *hardware* a través de un simple mensaje de WhatsApp, como se vio en la ponencia de **Rosline** de Miguel González) marca una transición del control de máquina rígido a una interfaz de colaboración intuitiva y accesible.

V. Fomento de la Comunidad y Desarrollo de Hardware Abierto

Finalmente, el Meetup sirvió como plataforma para destacar la importancia del desarrollo de hardware abierto y la conectividad comunitaria, elementos esenciales para sostener el crecimiento de la robótica en países con recursos limitados para la adquisición de equipos comerciales de alto costo.

El proyecto Orion encarna esta filosofía. Diseñado con el objetivo explícito de ser un robot de bajo costo y con licencias abiertas, Orion busca proporcionar una alternativa a plataformas costosas como Pepper o Nao. Al compartir planos, componentes y software, el proyecto fomenta la experimentación y la investigación en las universidades y la industria, reduciendo la barrera económica para acceder a plataformas de robótica avanzada. La integración de componentes accesibles, como la Raspberry Pi 4/5 y el uso de baterías estándar (18650), demuestra un enfoque pragmático para el desarrollo en contextos de recursos limitados, donde la optimización del costo es tan importante como la capacidad técnica.

En el plano de la interacción, la iniciativa Rosline de Miguel González ilustra la convergencia de la robótica con herramientas de comunicación masiva. Al crear un asistente de WhatsApp para ROS, se supera la barrera de las interfaces de depuración y control especializadas, permitiendo que cualquier persona con acceso a un *smartphone* pueda interactuar con un robot. Este tipo de innovación en la interfaz es crucial para la adopción masiva y la creación de un ecosistema que no dependa de un conocimiento técnico profundo para las interacciones básicas.

El evento, en sí mismo, representó un esfuerzo coordinado de múltiples ramas estudiantiles y organizaciones (Ecumen, RAS Colombia, Sinfonía Uniandes, etc.). La alta participación y el enfoque en compartir proyectos con licencia abierta reafirman el compromiso de la comunidad robótica colombiana con la colaboración como motor de innovación. Este espíritu comunitario es vital para el desarrollo de soluciones adaptadas a los contextos regionales y para asegurar que la robótica en Latinoamérica se desarrolle sobre cimientos de conocimiento compartido y accesibilidad.

Conclusión: La Robótica Latina en el Cruce de Caminos

El ROS Meetup Bogotá 2025 demostró que la robótica en América Latina se encuentra en un cruce de caminos dinámico, impulsada por la ambición de abordar problemas de alto impacto y la conciencia de la necesidad de democratizar las herramientas. Desde la vanguardia de la cirugía asistida por robot con Slicer ROS 2, pasando por los desafíos fundamentales de la dinámica y el control de humanoides, hasta las iniciativas de acceso masivo como Rossblocks y Rosline, el panorama es vibrante y multifacético.

La clave para el crecimiento sostenido, tal como lo indican las ponencias, reside en la colaboración abierta y en el desarrollo de soluciones contextualizadas. Es imperativo que la comunidad continúe reduciendo las barreras de entrada (como propone Rossblocks) para asegurar la masa crítica de desarrolladores necesaria para capitalizar el potencial global de crecimiento. Además, el enfoque en la seguridad (retroalimentación háptica) y la intuitividad (HRI con NLP) garantiza que los sistemas

robóticos avanzados sean adoptados de manera efectiva tanto en la industria como en la vida cotidiana. El camino hacia adelante implica la integración continua de la IA (RL) en el control, y la expansión de plataformas de hardware abierto como Orion, asegurando que la próxima ola de innovación robótica sea forjada por una comunidad amplia, competente y accesible