



UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE ELECTROTECNIA Y COMPUTACIÓN

PROTOCOLO DE TRABAJO MONOGRÁFICO

**SISTEMA OPERATIVO PARA ROBOTS(ROS): APLICACIÓN EN EL
DESARROLLO DE UN LABORATORIO VIRTUAL PARA EL ESTUDIO DE LOS
FUNDAMENTOS DE LA ROBÓTICA INDUSTRIAL.**

Presentado por: Br. Yeser Alfredo Morales Calero

Carnet: 2013-62290

Tutor: MSc. Alejandro Alberto Méndez Talavera

Octubre 01, 2018

Índice de Contenido

INTRODUCCIÓN.....	3
ANTECEDENTES.....	5
JUSTIFICACIÓN.....	9
1. OBJETIVOS.....	11
1.1. Objetivo general.....	11
1.2. Objetivos específicos.....	11
2. MARCO TEÓRICO.....	12
2.1. Tipos de Laboratorios.....	12
2.1.1. Laboratorios Físico.....	12
2.1.2. Laboratorios Virtual.....	13
2.1.3. Laboratorio Remoto online.....	13
2.1.4. Realidad Virtual 3D.....	13
2.2. Sistema operativo para robot (ROS).....	15
2.2.1. Conceptos básicos de ROS.....	15
2.2.2. Librerías de ROS.....	18
2.2.3. Herramientas de visualización de datos en ROS.....	19
2.2.4. Lenguajes de programación soportados.....	21
2.3. Laboratorio virtual para robótica industrial.....	22
2.3.1. Interfaz gráfica de usuario (GUI)	23
2.3.2. Fundamentos de la robótica industrial.....	24
2.3.2.1. Morfología del Robot.....	24
2.3.2.2. Herramientas matemáticas.....	24
2.3.2.3. Cinemática directa e inversa.....	25
2.3.2.4. Programación del robot.....	26
2.4. Uso del laboratorio virtual de Robótica Industrial.....	29
3. HIPÓTESIS Y VARIABLES.....	30
4. DISEÑO METODOLÓGICO.....	31
5. CRONOGRAMA DE EJECUCIÓN.....	32
6. BIBLIOGRAFÍA.....	33
ANEXOS.....	36

Introducción

La robótica es uno de los campos de la tecnología que se desarrolla velozmente y encuentra cada día nuevas áreas de aplicación de tal forma que la clasificación de los robots debe ser ajustada constantemente. En la actualidad encontramos aplicaciones de la robótica en el campo personal y doméstico, servicios profesionales, investigación y desarrollo. Unos de los primeros campos donde los robots entraron en acción fue la industria en la cual destaca el uso de los mismos en el sector automotriz (soldadura y/o manejo de materiales). A pesar de que el uso de los robots en la industria data de la década de los 60s, en Nicaragua el uso de los mismos es relativamente bajo. El hecho mencionado tiene un impacto negativo en el desarrollo de la industria nicaragüense. Son pocas las industrias que poseen robots industriales y, de igual forma, son pocas las instituciones de educación superior que cuentan con la infraestructura adecuada, y el recurso humano, para la enseñanza e investigación de la robótica. Una de las razones es que un laboratorio físico, para experimentar en los diferentes aspectos asociados con los robots industriales, requiere de una inversión relativamente alta dado que es necesario garantizar el espacio físico, adquirir los robots y su mantenimiento.

La falta de infraestructura para el desarrollo e investigación de la robótica ha sido experimentada en la mayoría de las instituciones de educación superior latinoamericana muchas de las cuales han recurrido a la utilización laboratorios virtuales, diseñados e implementados por dichas instituciones o adquiridos en el mercado internacional. MATLAB, software ampliamente utilizado en el campo de las ingenierías, cuenta con un toolbox para robótica (Corke, 1996) mediante el cual se puede modelar y simular el comportamiento de un manipulador, sin embargo, el mismo tiene limitaciones. Simscape Multibody (antes llamado SimMechanics), de MATLAB y Simulink, permite la simulación de sistemas mecánicos en 3D, tales como robots. El software mencionado requiere de una licencia y el costo de la misma aumenta en dependencia del número de usuarios. También se puede utilizar el software de programación de robots, versión evaluación, de algunos fabricantes, tales como Robot Studio de ABB. Dicho software, además de limitaciones temporales, solo puede ser utilizado en los robots fabricados por la empresa. Para trabajar con diferentes robots, es necesario adquirir y aprender los lenguajes de programación de cada uno de ellos.

En la actualidad se busca estandarizar el trabajo con los robots y una de las herramientas disponibles para tal fin es el middleware “Sistema Operativo para Robot (ROS, por sus siglas en inglés)” el cual se ha convertido en el estándar de facto a nivel industrial tanto así que ya existe una versión denominada ROS Industrial (2017). ROS (s.f.) es un middleware, open-source, para el desarrollo a gran escala de complejos sistemas robóticos. Se trata de una colección de herramientas, librerías y convenciones que tienen como objetivo simplificar la tarea de crear un

comportamiento complejo y robusto en una amplia variedad de plataformas robóticas.

En este documento proponemos, como trabajo de monografía, dar a conocer el middleware ROS y mostrar sus posibilidades en el campo de la robótica mediante el desarrollo de un laboratorio virtual de robótica (LVR) para el estudio de los fundamentos de los manipuladores industriales tipo serie. ROS permite la programación de robots de diferentes fabricantes siempre y cuando se cuenta con las librerías apropiadas las cuales aumentan cada día dada que ROS es una comunidad y permite la reutilización de software. Para el desarrollo del laboratorio virtual, además de ROS, se utilizarán herramientas de simulación tales como GAZEBO, visualizadores como RVIZ y RQT. La integración de los diferentes nodos será realizada mediante el lenguaje de programación C++, la modelación de los robots será desarrollado utilizando el formato de descripción unificado de robots (URDF, por sus siglas en inglés), y para la creación de ambientes y escenarios utilizaremos el formato de descripción de la simulación (SDF, por sus siglas en inglés).

El laboratorio virtual contará con los recursos necesarios para que los estudiantes estudien y realicen prácticas relacionadas a la morfología de los robots, herramientas matemáticas necesarias para el estudio de los robots, cinemática de los robots y programación de manipuladores industriales. Un elemento de mucha importancia en el LVR, es el simulador que incorporará el cual permitirá manipular el modelo de un robot en un escenario particular, en respuesta a un programa utilizando instrucciones parecidas a las que incorporan los lenguajes de programación de fabricantes tales como ABB.

El laboratorio virtual de robótica contará con un manual de usuario en el cual será presentada la información de los recursos disponibles en el LVR de forma tal que los docentes puedan diseñar prácticas que permitan a los estudiantes afianzar los fundamentos de la robótica industrial. Por ejemplo, el profesor podrá elaborar una práctica de laboratorio, a partir de los modelos de robots presentes en el LVR, para que los estudiantes apliquen el procedimiento de Denavit y Hartenberg para determinar la matriz de transformación homogénea. En su versión inicial, el LVR contará con dos guías para realizar algunas actividades que permitan a los futuros usuarios apropiarse del procedimiento necesario para utilizar efectivamente los recursos de este.

Antecedentes

Con el objetivo de identificar trabajos relacionados con el propuesto en este protocolo de trabajo monográfico, se buscó información en el centro de documentación de la facultad de electrotecnia y computación, así como en el repositorio del consejo nacional de universidades, el cual presenta información sobre trabajos de investigación realizados en las universidades miembros de dicho organismo. En cuanto al Sistema Operativo para Robot (ROS), cuya primera versión fue lanzada en el 2007, dichos lugares no cuentan con información sobre trabajos en los cuales se haya utilizado dicho sistema. De igual forma, documentación de laboratorios virtuales, desarrollados en el seno de las universidades del CNU (2018), es poca y destaca el trabajo monográfico “Laboratorio virtual para el desarrollo de prácticas en la disciplina de Control automático y Automatización industrial en la carrera de Ingeniería Electrónica” desarrollado por Pérez et al (2016a). El sistema desarrollado cuenta con cuatro plantas virtuales el comportamiento de las cuales puede ser controlado mediante programas escritos por los estudiantes utilizando el TIA Portal de Siemens.

La promoción y uso de ROS muestra algunos indicios en Centroamérica, destacando el curso “Robótica,” desarrollado por la Universidad de Costa Rica, en el cual una de las unidades es dedicada al sistema operativo para robot (Ramírez, 2018). En el blog COSTARICAMAKERS (2018), utilizan ROS para promover la robótica y presentan la utilización de este para el reconocimiento de patrones usando una cámara web. En un trabajo en conjunto con la Universidad de Sao Paulo, Brasil, se realizó en el Tecnológico de Costa Rica, un trabajo monográfico en el cual se utilizó ROS para la obtención de datos de sensores 3D para vehículos autónomos. El trabajo fue desarrollado por Valverde (2015).

En la Universidad Tecnológica de Panamá se elaboró el diseño e implementación de un sistema de control para un Mini carro tele operado usando ROS. El trabajo lleva por nombre MODELADO Y CONTROL DEL ROBOT MÓVIL ROBOTNIK SUMMIT XL desarrollador por Rosas et al (2013).

A diferencia de Centroamérica, el sistema operativo para robot ha tenido una amplia aceptación en América del Sur siendo el mismo utilizado en cursos robótica, así como en trabajos de tesis en las universidades de mayor prestigio. Los países de América del Sur con mayor grado de adopción de ROS son Chile, Brasil, Ecuador, Argentina y Colombia.

En América del Norte, México presenta un movimiento muy importante al desarrollar Robots utilizando ROS. Se organizan competencias, poseen trabajos académicos tanto de desarrollo como de investigación. Es destacable el trabajo de Estudiantes del Instituto Tecnológico y de Estudios Superiores de Monterrey que participarán en la competición para estudiantes University Rover Challenge (URC) (Eagle X 2018). Es tan importante el trabajo desarrollado en México que han ganado premios a nivel

internacional tal como el 4to lugar obtenido en la competencia RoboCup del año 2017 con el robot humanoide de servicio JUSTINA el cual también está construido bajo ROS para controlar cada una de las funcionalidades de las cuales fue provista (Justina, 2017).

Es importante, dada el impacto que ha tenido en el desarrollo de la robótica, conocer el sistema operativo para robot, ROS en general y ROS industrial en particular, sus aplicaciones y tendencias.

En esta propuesta de trabajo monográfico se plantea mostrar la importancia de ROS elaborando los medios requeridos para tal fin, así como el desarrollo de un laboratorio virtual, teniendo a ROS como soporte, para el estudio de los fundamentos de la robótica industrial. Considerando lo limitado de los recursos disponibles para el estudio de la robótica en nuestro país, el laboratorio virtual sería de mucha ayuda en el proceso de formación de los estudiantes.

El desarrollo de laboratorios virtuales es una alternativa que ha sido asumida a nivel mundial y es así que encontramos instituciones académicas que los diseñan e implementan para uso propio o que los adquieren de empresas que se dedican al desarrollo de estos con fines comerciales como es el caso de la empresa TheConstructSim (Tellez, 2016), que proveen servicios para simular robots. Su estrategia de inmersión en el campo de la educación es garantizar todos los recursos vía web (Simulaciones de Robots en la Nube (Cloud)) para enseñar robótica y aunque esta empresa usa librerías nativas de software libre de ROS, el uso de sus plataformas tiene un costo para el paquete principiante de €15 y una cuota mensual de €39.97.

En la Universidad Nacional de Ingeniera, en Nicaragua, Pérez et al (2016b) desarrollaron el “Laboratorio virtual para el desarrollo de prácticas en la disciplina de Control automático y Automatización industrial en la carrera de Ingeniería Electrónica” cuya finalidad es verificar la efectividad de programas para PLC escritos por los estudiantes. El sistema desarrollado cuenta con cuatro plantas virtuales las cuales contienen elementos comúnmente encontrados en una planta real tales como sensores, bombas, calentadores, etc., teniendo el docente la posibilidad de decidir el comportamiento deseado de la planta y solicitar a los estudiantes el programa para lograrlo. Al correr el programa, se puede apreciar la animación de las partes correspondientes tales como activaciones de sensores, cambios de nivel en un tanque, activación de un calentador, etc., lo cual brinda al estudiante información visual que le permite verificar la efectividad de su programa.

En Latinoamérica se ha realizado un trabajo importante para desarrollar laboratorios virtuales en diferentes campos de la ingeniería. En cuba, se propuso un laboratorio para procesos de control e instrumentación (Ayala, et al, 2016). También destaca el trabajo desarrollado en Guatemala para ejecución de prácticas de circuitos eléctricos por Juárez (2008) quien establece que “se considera que un laboratorio virtual será de gran utilidad para los estudiantes y los docentes de la universidad, porque el

objetivo es facilitar la enseñanza y la comprensión de los conceptos teóricos vistos en clase, mediante las tecnologías de la información y la comunicación”, Diseño de un compilador para programación en MOODLE (Merino, 2015) desarrollado en Ecuador, un laboratorio virtual llamado SISMLAB para ingeniería sísmica en Colombia (Guerrero, et al, 2014). Todas estas propuestas de laboratorios virtuales poseen un propósito compartido de acercar al estudiante a la experimentación virtual.

En el campo de la robótica, asunto que nos ocupa en la presente propuesta de trabajo monográfico, México ha incursionado en la enseñanza de la robótica utilizando laboratorios virtuales. (Prieto, et al, 2010; Ortega, et al, 2016a), plantean que la enseñanza de la robótica a menudo requiere de costosos laboratorios que están más allá del alcance de la mayoría de las universidades públicas. Son pocos los trabajos de laboratorios virtuales de robótica que tengan en consideración la importancia de modelar un ambiente de trabajo virtual robótico lo más real posible involucrando gravedad, interacción entre cuerpos, velocidad de movimiento, dinámica de los cuerpos etc. Ortega, et al (2016b) analizan estas situaciones y proponen en su trabajo la integración de motores de física virtual tales como PhysX y el entorno de desarrollo Unity 3D para el desarrollo de una solución enfocada a un laboratorio virtual completo con tres niveles de desarrollo, Nivel de Ambientes 3D usando herramientas de CAD y modelado 3D, Nivel de instrumentación y control con el análisis matemático del modelo del sistema para introducir etapas de control PID en cada eslabón del Robot con configuración de operación paralela (Robot Delta) y conseguir una aproximación de respuesta del diseño virtual a un diseño real, además del Nivel de HMI que involucra al estudiante con la práctica permitiendo ajustar parámetros físicos y dinámicos del modelo virtual tales como la masa de cada elemento del robot, características mecánicas y eléctricas de los motores virtuales, parámetros del controlador PID, etc.

Laboratorios Virtuales remotos, tales como el Laboratorio Remoto usando el robot PR2 propuesto por Pitzer et al (2012), permiten a un usuario observar, desde una página WEB con GUI's y modelos virtuales, el comportamiento bajo cámaras WEB de un robot físico al experimentar con este. Otro trabajo bajo el entorno ROS es el desarrollado por Casañ, et al (2015) quienes proponen el uso de las funcionalidades de ROS, pero agregando una plusvalía de carácter open-source como lo es la plataforma Moodle, para centralizar el acceso de los estudiantes del laboratorio remoto, todo esto bajo un mismo servidor donde se ejecutan los nodos de ROS y la base de datos de Moodle que son accedidos desde una página WEB.

Los resultados de la revisión de literatura muestran la importancia que tienen tanto ROS en el fomento del desarrollo de la robótica como los laboratorios virtuales para mejorar la formación de los estudiantes relacionados con dicho campo. Es importante que en Nicaragua utilicemos herramientas como ROS para el estudio de la robótica y es por eso que la propuesta de trabajo monográfico aquí presentada, con pocos trabajos relacionados similares en la región, está enfocada en dar a conocer las principales características de ROS, sus alcances y tendencias así como mostrar su

utilidad en el mundo académico mediante el desarrollo de un laboratorio virtual que contará con modelos tridimensionales de robots cuyo comportamiento podrá ser manipulado desde una interface gráfica (GUI) diseñada para tal fin o mediante programas escritos por los estudiantes. El middleware ROS servirá como elemento centralizador de los diferentes nodos del sistema. A diferencia de otros laboratorios virtuales que están destinados a práctica específicas, en el laboratorio virtual que se propone el docente encontrará los elementos que generalmente se encuentran en un laboratorio físico y podrá decidir que experimentos realizar, elaborando las guías de laboratorio necesarias para garantizar los objetivos establecidos.

Justificación

El campo de la robótica ha tenido, principalmente en las últimas décadas, un desarrollo vertiginoso y está impactando de forma positiva, principalmente en países desarrollados, en áreas como la medicina, la industria, y la investigación, entre otras. En Nicaragua, país en vías de desarrollo, el uso de la robótica es relativamente bajo, sin embargo, dicha tecnología tendrá mayor presencia en el futuro cercano y por consiguiente es necesario contar con profesionales con las competencias requeridas para realizar efectiva y eficientemente las diferentes tareas asociadas con la aplicación, el desarrollo y la investigación de la robótica.

La formación de profesionales en el campo de la robótica requiere de recursos humanos e infraestructura apropiada para tal fin (Jara, 2011a). En Nicaragua pocas instituciones de educación superior cuentan con recursos básicos mínimos para la formación de los estudiantes en el campo de la robótica (Universidad Tecnológica La Salle, 2017), siendo uno de los principales obstáculos para tener mejores condiciones la imposibilidad de garantizar la alta inversión requerida para la adquisición de los robots, software, accesorios asociados, instalación y mantenimiento de acuerdo a un estudio de Castellanos, & Martínez (2010). Compañías como FESTO ofrecen Kits denominados modulares, para prácticas de un proceso automatizado utilizando un robot Mitsubishi, que cuesta alrededor de 41,500 euros.

En el campo industrial, muchos fabricantes ofrecen manipuladores industriales los cuales tienen precios altos y, además, cada uno ofrece su propio software para la programación de los robots (Owen, 2016). Lo anterior significa que los estudiantes o especialistas en robótica industrial tendrían que aprender a utilizar el software correspondiente al manipulador en turno lo cual es imposible lograr si no se cuenta con la infraestructura adecuada. Se han realizado intentos para estandarizar lo relacionado a la programación de los robots, hacerla independiente del fabricante, y uno de los resultados más destacado en la última década es el middleware “Sistema Operativo para Robot” (ROS, por sus siglas en inglés). En la actualidad es considerado el estándar de facto para aplicación, desarrollo e investigación de la robótica y en el campo industrial se ha creado ROS-Industrial.

El desarrollo de un laboratorio virtual en robótica contribuiría en la mejora de las condiciones requeridas para el estudio de los fundamentos básicos de esta ya que brindaría a los estudiantes, y docentes, una herramienta que podría ser utilizada en cualquier lugar y en cualquier momento.

El laboratorio virtual sería desarrollado tomando como elemento principal el middleware ROS lo cual, entre otras cosas, permitiría a los estudiantes experimentar con modelos de robots de diferentes fabricantes tales como ABB, KUKA, FANUC, entre otros.

La inversión en el desarrollo e implementación del laboratorio sería muy inferior respecto a la requerida para tener un laboratorio físico ya que la mayoría de las herramientas utilizadas, tales como C++, ROS, GAZEBO, Qt creator, no requieren de licencia.

Los principales beneficiarios de los resultados de este proyecto serán los docentes y los estudiantes. Los primeros dispondrán de una herramienta a partir de las cuales podrán diseñar experimentos, con su guía apropiada, relacionados con los fundamentos básicos de los robots industriales. Los estudiantes, por su parte, contarán con una herramienta flexible, podrán utilizar en cualquier momento y en cualquier lugar, que les posibilitará verificar los fundamentos teóricos relacionados con los robots industriales contenidos en programas de asignatura relacionados o programar los robots para que realicen tareas básicas.

La sociedad nicaragüense sería también beneficiada ya que, producto del trabajo desarrollado por los estudiantes en el laboratorio virtual, los mismos contarán con una formación más sólida en dicho campo y podrán contribuir efectivamente en la utilización, desarrollo e investigación de la robótica en el país.

1. **Objetivos**

1.1 **Objetivo general**

Desarrollar un laboratorio virtual para el estudio de los fundamentos de la robótica industrial utilizando, para su implementación, el sistema operativo para robots (ROS).

1.2 **Objetivos específicos**

- Identificar las principales características/recursos/estructura de ROS enfatizando aquellas aplicables directamente a la robótica industrial.
- Elaborar un documento donde se muestre el ABC de ROS.
- Identificar las herramientas de software requeridas, además de ROS, para el desarrollo del laboratorio virtual para el estudio de los fundamentos de la robótica industrial.
- Desarrollar la interfaz gráfica de usuario (GUI) del laboratorio virtual, según los requerimientos establecidos.
- Verificar la efectividad del laboratorio virtual mediante la realización de dos prácticas de laboratorio.
- Elaborar un manual de usuario del laboratorio virtual.

2. Marco Teórico

Las instituciones académicas, principalmente las de educación superior (IES), buscan constantemente mejorar la calidad de la educación y una forma de hacerlo es integrando nuevas técnicas y herramientas orientadas a mejorar los resultados del proceso enseñanza-aprendizaje. En la actualidad, en muchas IES, a la tradicional clase presencial se suman otras formas de enseñanza-aprendizaje las cuales se fundamentan en las TIC, permitiendo de esta manera nuevas formas de enseñar, aprender, generar y compartir conocimiento.

Un recurso de mucho impacto en la formación de los futuros profesionales y cuya presencia crece día a día en los procesos de enseñanza aprendizaje es el laboratorio virtual (LV), el cual puede presentar altos niveles de flexibilidad y la inversión es muy inferior a la requerida para instalar un laboratorio real.

Uno de los campos en el cual el desarrollo y utilización de un laboratorio virtual reviste mucha importancia, dada la inversión alta requerida para tener un laboratorio físico, es el de la robótica en general y en particular, la robótica industrial.

En la introducción del presente documento se estableció que el trabajo de monografía propuesto pretende dar a conocer el middleware ROS, su importancia en el desarrollo de la robótica industrial, y dar muestra de sus posibilidades mediante el diseño y construcción de un laboratorio virtual que sirva de base para el estudio de los fundamentos de los manipuladores industriales tipo serie.

En los siguientes apartados se presentan los elementos teóricos-tecnológicos requeridos para el desarrollo del laboratorio virtual.

2.1 Tipos de laboratorios

La experimentación de un fenómeno físico, o mediante la simulación, requiere de entornos de trabajos que reúnan los equipos o herramientas necesarias para el desarrollo efectivo de una práctica. En los ambientes académicos podemos encontrar, en la actualidad, diferentes tipos de laboratorios.

2.1.1 Laboratorio Físico

Es ampliamente utilizado en universidades con modelos clásicos de enseñanza convirtiendo estos sitios de trabajo como el único sitio concebido para elaborar un experimento el cual involucra la presencia física tanto del tutor como del alumno. Cabe destacar que la interacción directa con los equipos apropiados de un laboratorio físico aporta una experiencia difícil de igualar debido a que los alumnos perciben los resultados de carácter palpable entrando en juego los cinco sentidos (Vista, tacto, audición, olfato e incluso, a veces el gusto) Calvo, (2009a).

2.1.2 Laboratorio Virtual

Es una alternativa al laboratorio físico y utiliza recursos computacionales haciendo uso de modelos matemáticos y recursos de visualización como modelos CAD y animaciones gráficas. Se pretende, mediante de este tipo de laboratorio, que el estudiante pueda estudiar el comportamiento de la realidad mediante el uso de modelos, sencillos o complejos, de la misma. Los recursos de un laboratorio virtual pueden ser ampliados en el tiempo permitiendo la realización de prácticas asociadas a temas de mayor complejidad.

2.1.3 Laboratorio Remoto online

Nacen bajo la necesidad de dar acceso a los estudiantes a un espacio de trabajo, de forma online, que combina los recursos de un laboratorio físico con recursos de software. El estudiante accede al laboratorio por medio de una página web pudiendo de esta forma controlar los recursos disponibles y llevar a cabo el experimento de interés. Su aprovechamiento está basado en la accesibilidad por parte de un usuario en horarios flexibles.

2.1.4 Realidad virtual 3D

Es la integración de recursos de hardware y software para acercar aún más al usuario a la experimentación de una teoría estimulando los sentidos tanto de la vista al exponerlos a entornos virtuales 3D y audición bajo altavoces con sonido envolvente.

En la actualidad se puede encontrar diferentes recursos para la enseñanza de la robótica Industrial tales como kits de robótica industrial, laboratorios virtuales de robótica de categoría libre o de pago para ejecutar cierta práctica. El desarrollo de un laboratorio remoto requiere contar con acceso a un laboratorio físico y enlazarlo a un servidor para que el estudiante pueda acceder vía remota garantizándole una plataforma amigable y multiplataforma como pueden ser los navegadores WEB. Candelas, F. et al. (2004) analizan las ventajas del uso de un laboratorio remoto ROBOLAB, proyecto desarrollado por ellos mismos bajo el nombre de grupo de investigación AUROVA. El proyecto utiliza herramientas gráficas para modelado y visualización de objetos 3D de un brazo robótico que coincide con el mismo modelo de un robot físico, permitiendo al estudiante realizar sus practica al observar y configurar el modelo virtual y una vez alcanzado un nivel de aprendizaje aceptable, proceder a interactuar con el robot físico utilizando la misma plataforma web. Ellos concluyen que al utilizar el laboratorio Remoto ROBOLAB *“La mayoría de alumnos prefieren disponer de un laboratorio en la universidad dónde trabajar con la ayuda*

de los compañeros y el apoyo didáctico del profesor, pero también hay muchos alumnos que reciben con agrado la opción de un laboratorio virtual que les ofrezca unos horarios flexibles en los que realizar los experimentos". (Candelas, et al, 2004b).

En resumen, un laboratorio virtual es un elemento importante en el proceso enseñanza-aprendizaje ya que destacan, entre otras, las siguientes ventajas según Calvo, et al (2009b):

1. El estudiante se familiariza con el experimento evitando acudir al aula sin conocimiento previo.
2. Comparación del comportamiento de modelos matemáticos ante una simulación permitiendo extraer sus propias conclusiones de cierta práctica.
3. Manejo de herramientas informáticas contemporáneas para la formación integral de un estudiante.
4. Repetitividad de los experimentos realizados por el estudiante que permitirá reproducir cuantas veces desee hasta consolidar el conocimiento.
5. Disminución de riesgos y accidentes que pueden ocasionar una mala práctica o configuración de un equipo físico.
6. Multiplicidad de experimentos simultáneos realizados ya que cada estudiante podrá ejecutar la práctica indicada en su computador asignado, además de esta forma se favorecen los procesos colaborativos como el de "Lluvia de ideas" al opinar cada alumno sobre su percepción adquirida al ejecutar la práctica.

Considerando lo anterior, se decidió, para dar a conocer la importancia y posibilidades de ROS, desarrollar un laboratorio virtual el cual tendrá a ROS como elemento integrador. Es importante destacar la inversión relativamente baja requerida para implementar el laboratorio virtual, la posibilidad de ampliación del mismo y la posibilidad que el mismo ofrecerá a los docentes para diseñar nuevas prácticas a partir de los recursos disponibles.

2.2 Sistema operativo para robot (ROS)

El sistema operativo para robot (ROS), elemento fundamental del presente trabajo, es un middleware ampliamente utilizado en el mundo creciente de la robótica. Originalmente fue desarrollado en el 2007 en los laboratorios de Inteligencia Artificial de Stanford y en el 2008, cedieron el derecho de desarrollo al instituto de investigación de robótica Willow Garage (2010), donde la filosofía es proporcionar bibliotecas y herramientas libres para ayudar a los desarrolladores de software a crear aplicaciones para robots

En esta propuesta de trabajo monográfico, se promueve la importancia de ROS en el campo de la robótica y se hace mediante el diseño y construcción, utilizando los recursos proporcionados por este, de un Laboratorio Virtual de Robótica, en el cual será posible realizar experimentos sobre los fundamentos de la robótica industrial específicamente de los manipuladores industriales tipo serie.

Una característica importante de ROS es su código abierto (Open Source) lo que ha llevado al desarrollo, por muchos colaboradores a nivel mundial, de una amplia colección de herramientas, librerías y convenciones. Los recursos que ofrece ROS permiten el desarrollo, a gran escala, de sistemas robóticos complejos, tanto físicos como simulados.

Los desarrolladores de ROS han establecido un conjunto de convenciones y buenas prácticas en el desarrollo de software de forma tal que se fomente la reutilización de código para robots, trabajando sobre una arquitectura robótica totalmente sólida y funcional. De esta forma se garantiza, entre otras cosas, que el proyecto no sea fallido y que sus resultados satisfagan la calidad requerida en este tipo de trabajos.

2.2.1 Conceptos básicos de ROS

El sistema operativo para robot (ROS) fue diseñado bajo una estructura distribuida y modular con el propósito de que los usuarios puedan usar los recursos requeridos según la aplicación a desarrollar. Es decir, el usuario puede seleccionar los recursos de software necesarios para implementar la solución.

En su operación, ROS presenta una red de **procesos** que se ejecutan simultáneamente en una estructura **peer-to-peer** donde cada nodo o elemento del sistema puede actuar al mismo tiempo como cliente y como servidor. Para una aplicación distribuida de ROS se requiere que los nodos peer to peer estén sincronizados bajo un middleware el cual genera un orden temporal para los eventos generados en el sistema, además de utilizar la comunicación XML/RPC que define la información de registro de cada nodo al ROSMaster, una vez registrado los nodos se procede a ejecutar la comunicación peer to peer entre los nodos utilizando el protocolo TCP/IP y, entre los muchos conceptos básicos relacionados en la misma, destacan los siguientes:

1. **Nodos** (nodes): Son procesos que realizan cálculos. Por ejemplo, en un sistema robótico, un nodo realiza la planificación de la trayectoria y otro puede suministrar una vista grafica del sistema.
2. **Mensajes** (messages): Los nodos se comunican entre ellos enviándose mensajes. Un mensaje es simplemente una estructura de datos, que contiene campos de un solo tipo de datos. Tipos primitivos de datos (enteros, booleanos, punto flotante, etc.) son soportados, así como lo son arreglos de tipos primitivos.
3. **Tópico** (topic): El tópico es un nombre usado para identificar el contenido de un mensaje. Un nodo envía un mensaje publicándolo en un tópico dado. Los mensajes son enrutados vía un sistema de transporte con una semántica de publicación/suscripción. es un nombre utilizado para identificar el contenido de un mensaje. Un nodo que está interesado en cierta clase de datos debe suscribirse al tópico apropiado.
4. **Maestro** (master): El master proporciona los elementos necesarios para que los nodos se encuentren los unos a los otros, intercambien mensajes o invoquen servicios.
5. **Servicios** (services): Es un modelo de comunicación cliente-servidor, es una manera en el que los nodos se pueden comunicar. Estos permiten que se envíen solicitudes y se reciban respuestas
6. **Bolsas** (bags): Formato para guardar y reproducir datos de un mensaje proveniente de un nodo de ROS. Son un importante mecanismo para almacenar datos, tales como datos de sensores, que son difíciles de obtener pero que son necesarios para el desarrollo y prueba de algoritmos.

La arquitectura básica de comunicación de ROS, usando 3 nodos como ejemplo simplificado es mostrada en la figura 2.1. Para establecer comunicación con diferentes nodos es necesario la utilización de registros, cuyos nombres son suministrados por el maestro.

Como se muestra en la Figura 2.1 el Nodo 1, el Nodo 2 y el Nodo 3 se registran al Máster usando comunicación XML/RPC el cual, luego de conocer cada nodo registrado, permite a los diferentes nodos intercambiar la información entre ellos utilizando para esto el protocolo TCP/IP.

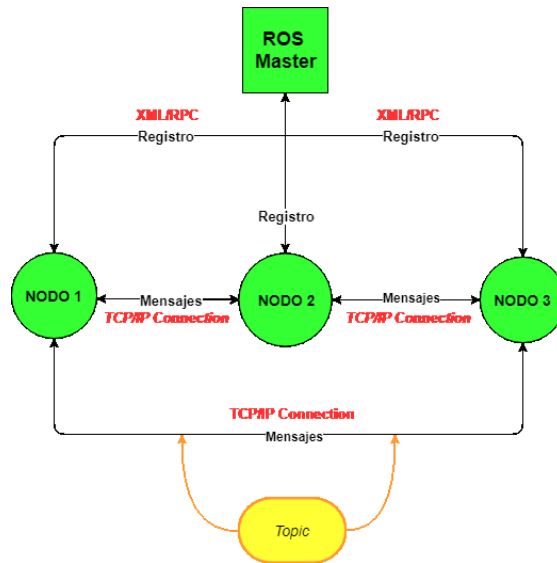


Fig.2.1.- Esquema de funcionamiento de la Arquitectura de ROS

Dado que cada nodo puede compartir múltiples mensajes, a estos se les clasifica como tópicos (topics) lo cual permite la centralización de información proveniente de varios procesos de un sistema robótico ya sea este físico o simulado. En la figura 2.2 podemos observar un comportamiento básico de un nodo y sus tópicos.

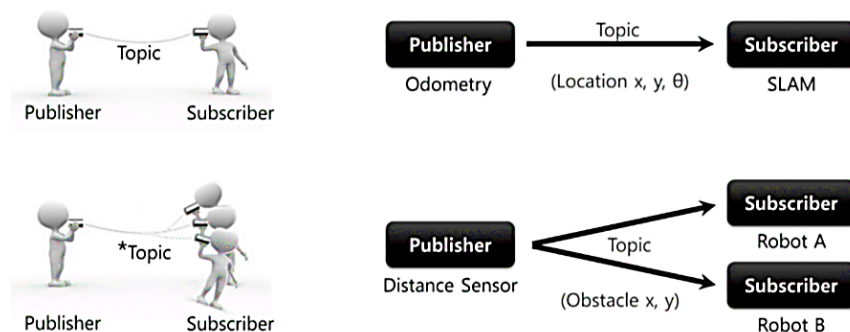


Figura 2.2. – Comunicación utilizando Tópicos. Copyright 2017 por ROBOTIS Co., Ltd.

Un ejemplo del funcionamiento de la arquitectura de ROS es mostrado en la figura 2.3 en la cual se identifica el **nodo** "robot_localization." El nodo mencionado recibe mensajes desde varios nodos bajo los **tópicos**: IMU, Camera, GPS, Detección 3D (Sensing 3D) y odometría (odometry), los datos suministrados en los **mensajes** son procesados por el nodo y como resultado comparte bajo un único tópico, y varios mensajes, al visualizador 3D RVIZ las coordenadas de localización del objeto físico para que este pueda ser representado de manera virtual.

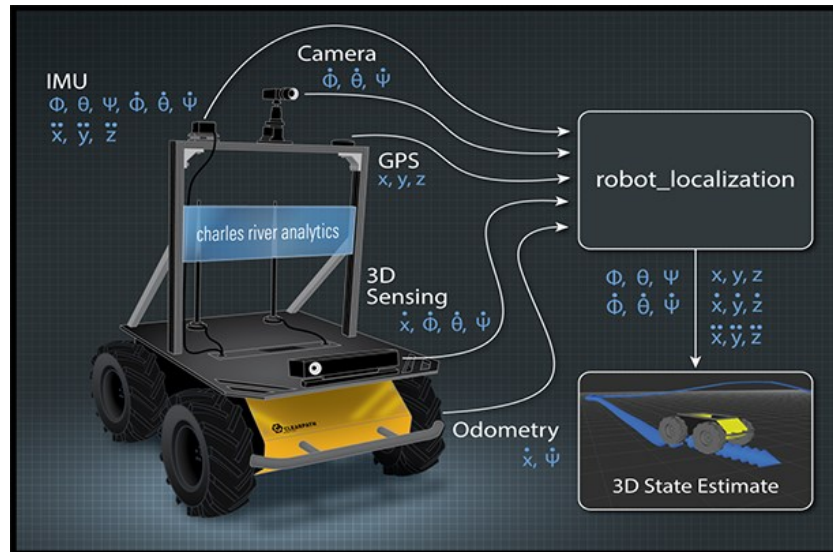


Fig.2.3 - Nodo robot_localization recibiendo tópicos de un robot físico y mostrando su localización en el ambiente virtual. Copyright 2017 por Charles River Analytics Inc.

Un nodo podemos considerarlo como un ejecutable dentro del paquete de ROS y utiliza la librería cliente, lista de nodos disponibles de ROS, para comunicarse con otros nodos, y estos a su vez se les configura para publicar o suscribirse a un tópico, o algunos procesos de ROS tales como servicios o almacenamiento de datos (Bags).

2.2.2 Librerías de ROS

El sistema operativo para robots (ROS) presenta una serie de características que lo hacen atractivo, para las personas dedicadas a la robótica, a la hora de diseñar e implementar sistemas basados en robots. Una de las características principales es que puede acceder a una amplia colección de librerías las cuales varían de acuerdo con la distribución de ROS (turtle, kinetic, etc.) con la que se desee trabajar. Las librerías de las distribuciones más recientes cuentan con librerías con códigos más estables e incorporan nuevas librerías probadas y recomendadas por la comunidad ROS.

Para implementar un nodo en ROS se pueden utilizar diferentes librerías tales como roscpp para C++ o rospy para Python.

ROS dispone de las librerías necesarias para la implementación de las actividades que deben ser desarrolladas en un sistema robótico tales como el intercambio de mensajes entre nodos o la estructuración de datos para descripción de trayectorias. En el desarrollo del trabajo que se propone será utilizada la distribución (versión) Kinetic dado que es una de las más estables hasta la fecha.

2.2.3 Herramientas de visualización de datos en ROS

Un elemento importante en cualquier laboratorio virtual de robótica es la posibilidad de visualizar el comportamiento del modelo de un robot en un escenario dado. También es fundamental que el mismo cuente con los recursos necesarios para facilitar el análisis de los datos generados por el sistema robótico bajo estudio. El sistema operativo para robots (ROS) dispone de una serie de herramientas para la visualización de los mensajes de los sistemas robóticos ya sean estos físicos o virtuales. Entre el sinnúmero de herramientas disponibles mencionamos RVIZ, RQT, y MoveIt, nodos en el sistema de suscripción de ROS con sus tópicos y servicios para permitir la entrada y salida de los datos a procesar. Las mismas herramientas mencionadas serán utilizadas en el desarrollo del laboratorio virtual propuesto.

- a) **RVIZ**: es una herramienta de visualización 3D que permite combinar en una misma pantalla modelos de robots, datos de sensores (cámara, láser, etc.) y otros datos provenientes de mensajes 3D. RVIZ será utilizada en el desarrollo del laboratorio virtual principalmente para visualizar la morfología y estudiar la cinemática de los robots. La figura 2.4 muestra una imagen generada utilizando RVIZ.

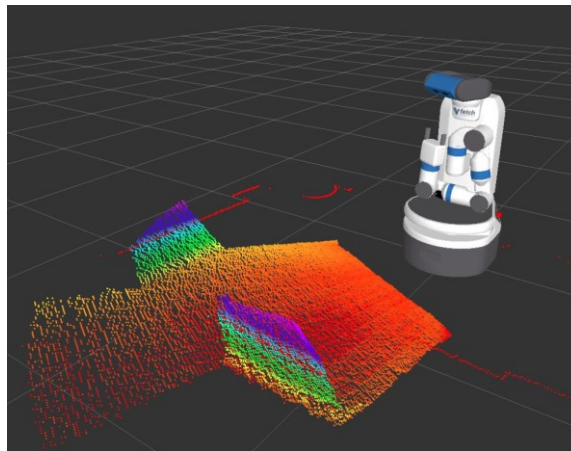


Fig.2.4. – RVIZ y visualización de modelo de robot y datos provenientes de mensajes 3D.

- b) **RQT**: La herramienta rqt ofrece varios plugins, entre los cuales destacan 'rqt_image_view', 'rqt_graph', 'rqt_plot' y 'rqt_bag', los cuales pueden ser utilizados para la introspección y visualización de datos provenientes de procesos de ROS. Por ejemplo, si es necesario visualizar los mensajes provenientes de una cámara en un sistema robótico, podríamos utilizar el plugin rqt-image_view. rqt expone los nodos, y las conexiones entre ellos, lo que permite entender la estructura el sistema, su funcionamiento, así como su fácil depuración. En la

figura 2.5 se muestra el resultado de utilizar rqt_plot para visualizar los datos provenientes de un nodo.

En el laboratorio virtual de robótica RQT será utilizado para graficar la estructura de datos correspondiente a la posición del robot durante su movimiento.

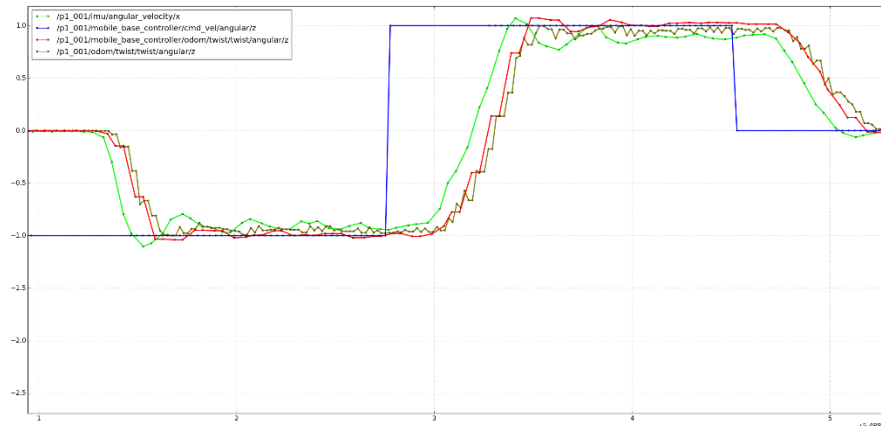


Fig.2.5. - Gráfico de línea XY de datos provenientes de nodos.

- c) **Movelt!**: Es una herramienta de software, escrita en C++, que permite la planeación de trayectorias para un robot industrial, así como la visualización de los datos y del modelo 3D del mismo, valiéndose de un plugin que es incorporado a la aplicación de visualización de datos RVIZ. Actualmente, Movelt (s. f.) es utilizado en diversas industrias para hacer la planeación de las trayectorias para diferentes tipos de manipuladores industriales, adoptando la visión de ROS de facilitar la programación de diferentes tipos de robots con los mismos recursos de Software.

Al sistema en desarrollo se le puede suministrar mayor funcionalidad al utilizar ROS ya que el mismo permite la integración y utiliza la información generada por aplicaciones externas de código abierto muy populares tales como GAZEBO y OpenCV. Por ejemplo, es posible utilizar OpenCV para hacer reconocimiento de imágenes y tomar decisiones, a partir de los datos suministrados. Más adelante en el documento se presentarán las características de GAZEBO dado que es el simulador que será incorporado en el LVR en la practicas de programación del robot.

2.2.4 Lenguajes de programación soportados por ROS

En el desarrollo del software para una aplicación de robótica, ROS permite el uso de distintos lenguajes de programación. De manera oficial soporta Python, C++ y LISP. Java podría ser soportado en el futuro, pero en la actualidad se encuentran en una fase experimental, apoyada por Google. De igual forma C# se encuentra en una fase experimental utilizando UNITY 3D, el cual es un motor de video juegos multiplataforma que permite simular robots importando el URDF como un GameObject en UNITY3D y obtener de este una simulación más realista con el renderizado de UNITY3D.

Con ROS es posible utilizar nodos creados en diferentes lenguajes de programación tales como Python y C++ y garantizar el intercambio de información entre estos.

En el desarrollo de la GUI y el intercambio de información entre los nodos ROS se utilizará el lenguaje C++ y el compilador CMake el cual fue diseñado para construir, probar y empaquetar software. CMake es el compilador oficial soportado por la comunidad ROS y el mismo desplazó al anterior compilador llamado rosbuilt.

El ambiente integrado de desarrollo Qt-Creator posee un plugin para ROS (ROS Industrial, 2016) que cuenta con los recursos necesarios (por ejemplo, templates) para crear nodos, servicios, y archivos URDF, entre otros. Qt-Creator permite al desarrollador construir los recursos de software necesarios, desde el IDE, para un sistema robótico.

2.3 Laboratorio virtual para robótica industrial

En una institución de educación superior, que dispone de un laboratorio físico para la enseñanza de la robótica, el robot es parte de un escenario que simula un ambiente industrial y las prácticas generalmente están relacionadas con la programación del robot para que realice una tarea determinada. El sistema no presenta la flexibilidad necesaria para que los estudiantes mejoren su comprensión acerca de los fundamentos de la robótica industrial. De hecho, aprenden a programar el robot utilizando el lenguaje específicamente diseñado para este.

El laboratorio virtual para robótica que se propone desarrollar contempla la incorporación de los recursos necesarios para que los estudiantes estudien o realicen experimentos relacionados con temas tales como la morfología del robot, la cinemática del robot, la programación de robots de diferentes fabricantes independientemente del lenguaje específico desarrollado para cada uno. Se incorporarán recursos que permitirán al estudiante entender las herramientas matemáticas utilizadas en el campo de la robótica industrial, específicamente manipuladores industriales tipo serie.

En el desarrollo de software es común el uso de los términos Backend y Frontend los cuales aportan una perspectiva de la clasificación de tipo de Software a utilizar en la construcción del propio laboratorio virtual, se contempla el uso de ROS para manejo de datos en Backend y el uso de librerías de software libre para desarrollar GUI del lado de Frontend.

En el LVR el estudiante, para la realización de los experimentos, podrá utilizar modelos de robots de fabricantes tales como ABB o KUKA. De igual forma, contará con las herramientas y la guía para desarrollar sus modelos propios.

Los principales elementos del laboratorio virtual son:

- Interfaz gráfica de usuario (GUI) para (interacción, programación, visualización, etc.)
- Modelos de robots industriales de ABB, KUKA, y DELTA
- Escenarios para la realización de tarea específicas
- Documentación sobre aspectos fundamentales sobre robótica industrial
- Otros

La interacción entre los diferentes elementos del laboratorio virtual es garantizada mediante el middleware ROS.

A continuación, son presentados algunos de los conocimientos o recursos tecnológicos requeridos para el desarrollo del LVR, específicamente para la implementación de la interfaz gráfica de usuario (GUI) y la construcción de algunos recursos para el estudio de los fundamentos de los manipuladores industriales.

2.3.1 Interfaz gráfica de usuario (GUI)

Una interfaz gráfica de usuario (GUI, por sus siglas en inglés) facilita la explotación de una aplicación mediante la incorporación de elementos destinados para tal fin. La GUI debe comportarse de una manera comprensible y predecible, de modo que un usuario sepa qué esperar cuando realiza una acción. La interfaz gráfica de usuario del laboratorio virtual contará con los elementos necesarios (botones, sliders, editor de texto, pantalla de visualización, etc.) para realizar las prácticas sobre los fundamentos de la robótica industrial, así como para acceder a documentos de interés tales como libros, guías de laboratorio, entre otros. La GUI será diseñada de tal forma que el estudiante podrá usar los recursos de manera transparente sin necesidad de tener que configurar aspectos básicos de ROS tales como su inicialización y el lanzamiento de nodos.

En la actualidad existen muchos lenguajes que pueden ser utilizados para desarrollar la interfaz de usuario y entre ellos destacan C++, C#, Java, Python, Visual Basic. Un criterio para utilizar un lenguaje en particular es que el mismo sea multi-plataforma ya que la interfaz podría ser utilizada bajo diferentes sistemas operativos.

Hoy día están disponibles un sinnúmero de ambientes integrados de desarrollo (IDE, por sus siglas en inglés) que facilitan la programación ya que aglomeran todas las herramientas necesarias tales como conjuntos de librerías, preprocesador, depurador, y compiladores. Entre los IDEs más conocidos destacan, entre otros, Visual Studio, Eclipse, NetBeans, CLion, y Qt Creator. Para la construcción de la GUI del laboratorio virtual de robótica industrial será utilizado el IDE Qt Creator, debido, principalmente, a que este posee un plugin de soporte para ROS.

Qt Creator será utilizado para crear las aplicaciones o nodos de ROS y serán exploradas las capacidades que posee Qt Creator para la creación de aplicaciones de escritorio. Qt Creator cuenta con QML, un estándar propio del IDE, que permite crear aplicaciones dinámicas y más elaboradas capaces de incorporar recursos WEB. Esa línea de creación será utilizada para incorporar un lector de PDF con el objetivo que los estudiantes dispongan en el LVR de documentación de interés tales como libros o guías de laboratorio.

En el desarrollo de la GUI del LVR serán utilizados los widgets de Qt Creator (slider, botones, etc.) los cuales están escritos en C++. Esta línea de creación es muy importante para lograr la visualización en el LVR ya que es muy fácil incorporar las librerías de ROS, escritas en C++, a una aplicación utilizando widgets. Un ejemplo sería el desarrollo de una aplicación de escritorio personalizada que se acople a los

requerimientos del LVR al incorporar las librerías de RVIZ para visualizar los modelos de los robots.

Es importante mencionar que en el laboratorio virtual se realizarán procesos, con el objetivo de garantizar la transparencia en el uso de los sus recursos por los usuarios, que darán inicio a la ejecución de las interacciones en la interfaz de usuario tales como la visualización de los robots y datos, programación y simulaciones de los robots. Dichos procesos estarán basados en nodos que serán lanzados o eliminados de acuerdo con la lógica implementada en el propio LVR.

2.3.2 Fundamentos de la robótica industrial.

El laboratorio de robótica virtual que se propone tiene un alcance limitado ya que de los temas considerados en un curso básico de robótica industrial el mismo solo contempla recursos para la experimentación en temas como herramientas matemáticas, morfología del robot, cinemática y programación de robots. El trabajo pretende mostrar la potencialidad del sistema operativo para robots (ROS) y se pretende hacerlo desarrollando el LVR. La información suministrada acerca de ROS y el laboratorio propuesto deben servir de base para ampliar los recursos de este.

2.3.2.1 Morfología del robot

El laboratorio virtual de robótica contará con modelos creados utilizando URDF y los mismos podrán ser visualizados con la herramienta RVIZ. El estudiante podrá visualizar el espacio de trabajo, para una morfología dada, con la ayuda de plugins desarrollados para la visualización. Interactuar con diferentes tipos de articulaciones ya sean prismáticas o de revolución y robots de diferentes grados de libertad. El estudiante tendrá la posibilidad de experimentar con modelos construidos por él mismo utilizando el formato URDF.

RVIZ es una abreviación para ROS visualización y es una herramienta poderosa para la visualización en 3D. Permite al usuario ver el modelo simulado del robot, obtener información de los sensores del robot, y reproducir la información obtenida de los sensores. Mediante la visualización de lo que ve y hace el robot el usuario puede eliminar los errores de la aplicación robótica desde entradas de sensores hasta acciones planificadas o no planificadas.

2.3.2.2 Herramientas matemáticas

El modelado cinemático de un robot busca las relaciones entre las variables articulares y la posición (expresada normalmente en forma de coordenadas

cartesianas) y orientación del extremo del robot (expresada como matrices de rotación, ángulos de Euler o algún otro de los métodos establecidos para tal fin.

Entender las herramientas matemáticas utilizadas para determinar la localización espacial de los diferentes elementos de un robot industrial, especialmente del efector final, presenta cierta dificultad a los estudiantes, principalmente cuando se utiliza solamente la pizarra para su explicación. Para contribuir al entendimiento y aplicación de dichas herramientas el laboratorio virtual de robótica incorporará los recursos necesarios para que los estudiantes puedan analizar la posición de un objeto utilizando coordenadas esféricas y cilíndricas, así como la orientación de este mediante la matriz de rotación, los ángulos de Euler, y los cuaternios. Las operaciones requeridas para el desempeño de los recursos estarán soportadas por la librería TF2, disponible para C++ y Phyton, propia de la distribución de ROS, que permite hacer conversiones de sistemas de coordenadas.

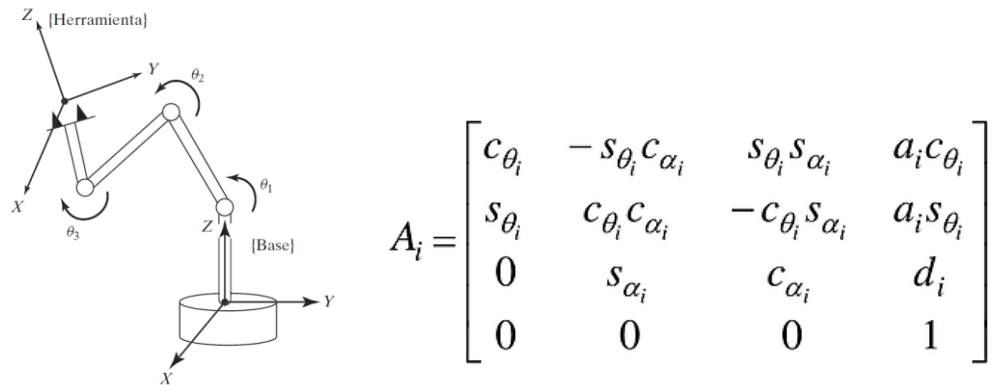


Fig.2.6. – Modelo de robot, sistemas de referencia.

2.3.2.3 Cinemática directa e inversa

La cinemática del robot estudia el movimiento de este con respecto a un sistema de referencia sin considerar las fuerzas que intervienen. Así, la cinemática se interesa por la descripción analítica del movimiento espacial del robot como una función del tiempo, y en particular por las relaciones entre la posición y la orientación del extremo final del robot con los valores que toman sus coordenadas articulares. Dentro de la cinemática se estudian la posición, velocidad, aceleración y todas las derivadas de mayor orden de las variables de posición (respecto al tiempo o a cualquier otra variable).

En muchas ocasiones es necesario conocer en cuál posición se encontrará el efector final del robot dadas ciertas posiciones de las articulaciones y dicho problema es abordado por la cinemática directa. En otras ocasiones es necesario conocer cuál debe ser la posición de las articulaciones para que el efecto sea ubicado en una

posición deseada. El último problema es abordado por la cinemática inversa. La figura 2.7 muestra la relación entre las variables para cada uno de los tipos de cinemática.

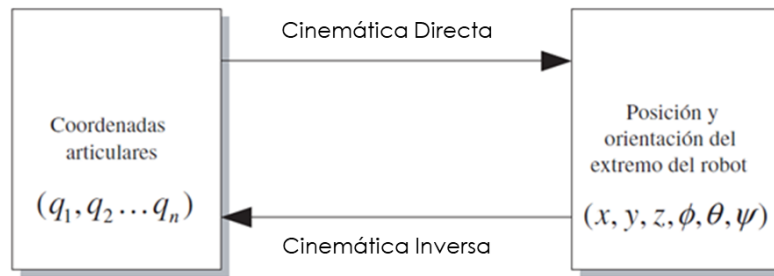


Fig.2.7. – Cinemática directa e inversa.

El laboratorio virtual de robótica que se propone contará con recursos que permitirán al estudiante realizar experimentos sencillos que le ayudarán a comprender las características de la cinemática directa e inversa.

A partir del modelo de un robot, construido utilizando el formato URDF, es posible obtener toda la información necesaria de la estructura de este utilizando la librería KDL. La información obtenida al utilizar el KDL en combinación con la información suministrada por el usuario (por ejemplo, los ángulos de los joints) mediante ciertos widgets de la GUI permiten manipular y/o determinar la posición del robot.

La librería KDL también puede ser utilizada para determinar la matriz de transformación homogénea la cual es utilizada para determinar la posición y rotación del efector final del robot. Los datos obtenidos serán visualizados mediante widgets del GUI desarrollados para la representación de datos.

2.3.2.4 Programación y simulación del robot

En la robótica, en general, uno de los más aspectos de mayor importancia es el relacionado con la programación de los robots. Durante la ejecución de un programa se interacciona con la memoria del sistema, leyendo y actualizando el contenido de las variables utilizadas en el programa; con el sistema de control cinemático y dinámico del robot, encargados de dar la señal de mando a los accionamientos del robot a partir de las especificaciones del movimiento que se les proporciona; y con las entradas-salidas del sistema, consiguiéndose así la sincronización del robot con el resto de las máquinas y elementos que componen su entorno.

El sistema de programación es, por tanto, la herramienta con que cuenta el usuario para acceder a las diversas prestaciones del robot, existiendo una relación directa entre las características y posibilidades del sistema de programación y las del robot en sí mismo.

La programación puede ser guiada o textual. Es importante destacar que en la actualidad es muy frecuente que los sistemas de programación de robots tiendan a combinar los dos modos básicos (guiado y textual), permitiéndose desarrollar el programa mediante la escritura de las instrucciones, y utilizando la posibilidad de guiado en línea en aquellos momentos en que sea necesario. Sistemas como RAPID de ABB, VAL II de Staübli y V+ de Adept Technology son ejemplos de esta ambivalencia.

Uno de los problemas relacionados con la programación de los robots es que no existe un estándar y el programador debe aprender el lenguaje específico para los robots de diferentes fabricantes. ROS brinda la posibilidad de programar robot de diferentes fabricantes desde el mismo entorno.

Supongamos que es necesario que dos robots de diferentes fabricantes, pero la misma morfología, deben seguir la misma trayectoria. En el ambiente de ROS, es posible escribir mensajes de ROS (que contiene la posición, velocidad, aceleración, etc., de cada articulación del robot), escritos en C++ o Python. El driver de cada robot, una vez recibido el mensaje, generará los comandos para que su robot siga la trayectoria indicada.

El laboratorio virtual de robótica permitirá al usuario escribir un programa (script, programación textual) que adoptará ciertas convenciones, utilizadas en el software de programación de robots de ABB Rapid, para que uno de los robots (modelos disponibles en el LVR) realice una tarea sencilla.

Una forma mediante la cual podemos verificar si el robot realiza la tarea indicada es la simulación. En la actualidad existen muchos simuladores, sencillos y complejos, para robots. También podemos clasificarlos en comerciales y de código libre.

Entre los comerciales destaca V-Rep el cual es un simulador dinámico con un entorno de desarrollo integrado que permite desde su interfaz lenguajes de programación tales como C, C++, Python, entre otros.

V-Rep es un simulador con entorno de desarrollo integrado donde cada objeto/modelo se puede controlar individualmente a través de un script de programación que puede ser escrito usando lenguajes de programación como C/C++, Python, Java, Lua, Matlab or Octave, V-REP se utiliza para el desarrollo rápido de algoritmos, simulaciones de automatización de fábricas, prototipado rápido y verificación y educación relacionada con la robótica.

De los simuladores open-source uno de los más utilizados es GAZEBO, el cual provee simulaciones dinámicas donde los robots pueden interactuar con el entorno (pueden coger, empujar, rodar, deslizarse por el suelo), capacidad de manipular las características del entorno tal como la gravedad del mundo virtual.

Gazebo es simulador dinámico 3D que tiene la habilidad para simular poblaciones de robots, de forma exacta y eficiente, en ambientes complejos internos y externos. Aunque similar a máquinas de juego, Gazebo ofrece una simulación física a un grado de fidelidad mayor, un conjunto de sensores, e interfaces tanto para programas como para los usuarios. Usos típicos de Gazebo incluyen:

- Prueba de algoritmos robóticos
- Diseño de robots
- Realización de pruebas de regresión con escenarios realísticos.

Algunas características claves de Gazebo:

- Máquinas físicas múltiples
- Una librería de modelos de robots y ambientes.
- Una amplia variedad de sensores

Aunque ambos simuladores tienen la posibilidad de comunicarse utilizando ROS se ha seleccionado GAZEBO como el simulador a incorporar en el LVR debido a que este es open-source. El estudiante podrá observar en el simulador GAZEBO el comportamiento del modelo del robot en respuesta al programa (script) escrito por el usuario. El script incorporará el conjunto de instrucciones las cuales serán procesadas por un nodo el cual enviará a GAZEBO, utilizando el plugging de conexión correspondiente, los mensajes apropiados para lograr el movimiento de las articulaciones del robot en el mundo.

SDF es un formato XML utilizado para describir objetos y ambientes para simuladores de robots, visualización, y control. SDF tiene capacidad para describir todos los aspectos de un robot, objetos estáticos y dinámicos, iluminación, terreno y también la física. En el laboratorio virtual el lenguaje SDF será utilizado para modelar los escenarios de trabajo en el ambiente de Gazebo.

Podrán ser utilizados los siguientes parámetros:

- ✓ **Escena:** Iluminación ambiental, propiedades del cielo, sombras.
- ✓ **Física:** Gravedad, paso del tiempo, motor de la física.
- ✓ **Modelos:** Colección de enlaces, objetos de colisión, articulaciones y sensores.
- ✓ **Luces:** Punto, espacio y fuentes de luz direccionales.
- ✓ **Plugins:** plugins soportado por gazebo, plugins del mundo, del modelo, del sensor y del sistema.

2.4 Uso del laboratorio virtual de robótica industrial

Los diferentes elementos involucrados en el uso, y posible ampliación del laboratorio virtual, son mostrados en la figura 2.8. El laboratorio virtual incluye un manual de usuario cuyo objetivo es presentar al usuario los recursos disponibles y un par de guías de laboratorio para mostrar como utilizarlos. Como se aprecia en la figura 2.7, el docente podrá, partiendo de los temas relacionados con los fundamentos de la robótica industrial que desee desarrollar y de los recursos disponibles en el laboratorio, elaborar las guías que utilizarán los estudiantes para realizar las actividades de estudio o las prácticas de laboratorio correspondientes.

Se pretende, como parte del trabajo monográfico, desarrollar un taller, para los docentes interesados, con el objetivo de brindarles los detalles de diseño y herramientas utilizadas en la implementación del LVR para que tengan una base para la ampliación de los recursos de este.

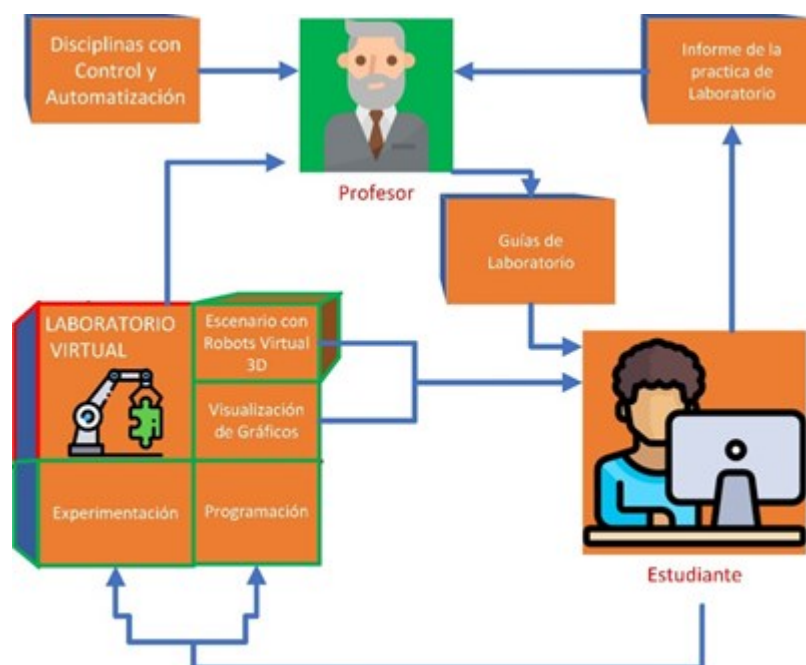


Fig.2.8. – Modelo de utilización del LVR.

3. Hipótesis y Variables

El laboratorio virtual de robótica industrial será el resultado de un proyecto de desarrollo ya que el producto será una aplicación de software que los estudiantes podrán utilizar para comprender y/o afianzar conocimientos básicos de la robótica industrial tales como estructura de un robot manipulador, cinemática directa e inversa, espacio de trabajo, y programación del robot, entre otros.

Variables: Amigabilidad del GUI, Efectividad del GUI, Efectividad de ROS, Efectividad y eficiencia del LVR.

4. Diseño Metodológico

El laboratorio virtual de robótica será desarrollado siguiendo las fases del método científico, análisis, hipótesis, síntesis, y validación, según lo establece Peter Bock en el libro *Getting it right: R&D Methods in Science and Engineering*. A continuación, una descripción breve de cada una de las fases.

- **Fase de Análisis:**

El laboratorio virtual de robótica estaría diseñado para ofrecer al estudiante un entorno que le permita comprender y verificar aspectos relacionados con la robótica industrial, por lo tanto, en la fase de análisis, con la ayuda del tutor, se determinarán los requerimientos que deberá satisfacer el LVR. Los requerimientos están asociados con los aspectos de la robótica industrial considerados importantes en la formación del estudiante en dicho campo, así como con los aspectos de la interfaz de usuario del LVR.

La fase de análisis contempla la revisión de literatura relacionada con los aspectos fundamentales de la robótica y el desarrollo de laboratorios virtuales incluido el diseño y construcción de las interfaces gráficas de usuario.

Entre otros resultados, de la revisión de literatura, se obtendrá información relacionada a las aplicaciones de software más utilizadas en el desarrollo de laboratorios virtuales en el campo de la robótica industrial. Dichos resultados serán evaluados para determinar los lenguajes de programación, simuladores, etc., apropiadas para el desarrollo efectivo y eficiente del LVR.

- **Fase de diseño:**

El resultado de la fase de diseño será una propuesta de la estructura del laboratorio virtual y la descripción de los recursos a utilizar en su implementación, ROS, lenguajes de programación, simuladores, etc.

- **Fase de desarrollo:**

En esta fase se implementa el laboratorio virtual de robótica y se diseñan y ejecutan los experimentos necesarios para verificar el funcionamiento del sistema. Los resultados de dichos experimentos serán utilizados para evaluar la respuesta del laboratorio virtual. Se implementarán los diferentes escenarios donde se realizarán los experimentos, la interfaz gráfica de usuario, el manual de usuario, y algunos ejemplos de guías de laboratorio.

- **Fase de validación**

Se determinará el desempeño del laboratorio virtual a partir de los resultados obtenidos en los experimentos y se formularán las conclusiones correspondientes. En esta etapa, se preparará el informe de resultados del proyecto.

5. Cronograma de ejecución

Actividad	Octubre				Noviembre				Diciembre				Enero			
Semana	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Fase Análisis																
Fase de Diseño																
Fase de Desarrollo																
Fase de validación																
Lanzamiento de la plataforma																
Elaboración del informe																
Defensa del Trabajo Monográfico																

6. Bibliografía

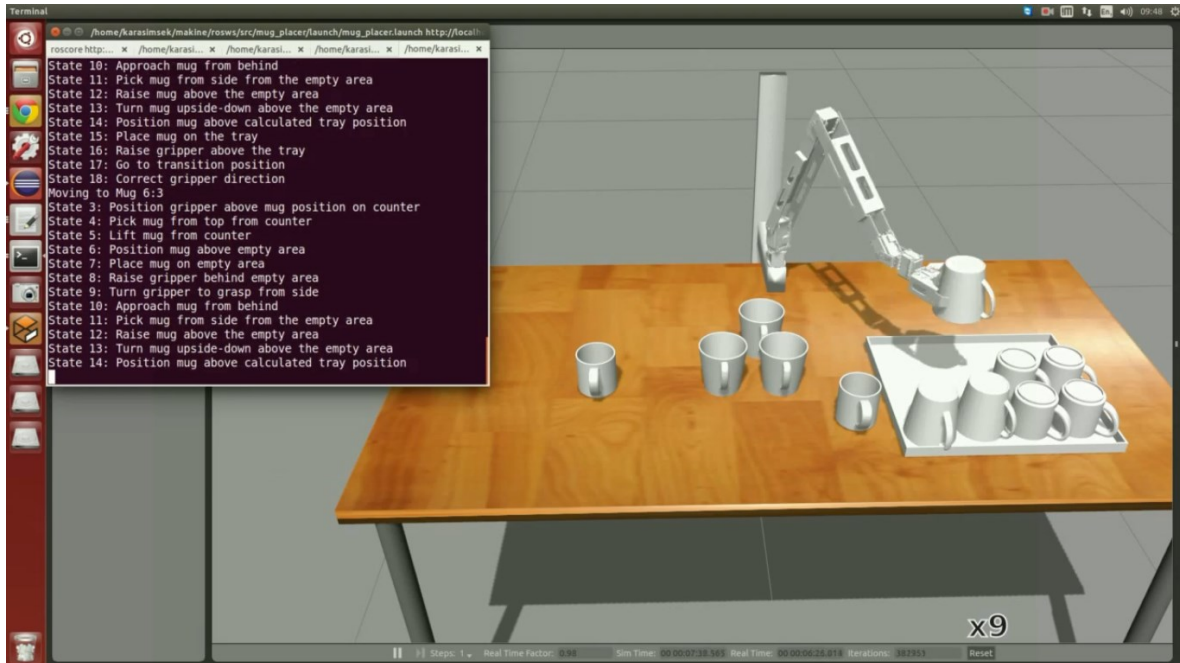
- [1] Jara, C., et al. (2011). Hands-on experiences of undergraduate students in Automatics and Robotics using a virtual and remote laboratory. *Computers & Education*, 57(4), 2451–2461. Obtenido de <https://doi.org/10.1016/j.compedu.2011.07.003>
- [2] Candelas, F., Torres, F., Gil, P., Ortiz, F., Puente, S., & Pomares, J. (2004). Laboratorio virtual remoto para robótica y evaluación de su impacto en la docencia. *RIAI: Revista Iberoamericana de Automática e Informática Industrial*, 1(2), 49–57. Obtenido de <http://rua.ua.es/dspace/handle/10045/4609>
- [3] Castellanos, F., & Martínez, O. (2010). Laboratorios virtuales (LV) como apoyo a las practicas a distancia y presenciales en Ingenieria. *INGE CUC*, 6(1), 267-280. Obtenido de <http://revistascientificas.cuc.edu.co/index.php/ingecuc/article/view/311>
- [4] Ortega, J., Sánchez, R., González, J., & Reyes, G. (2016). Virtual laboratories for training in industrial robotics. *IEEE Latin America Transactions*, 14(2), 665-672. doi 10.1109/TLA.2016.7437208
- [5] CNU (2018, abril) Repositorios Nicaragua. En Wellcome to the university repository of the CNU Obtenido de <http://repositorio.cnu.edu.ni/>
- [6] Ramírez, K. (2018, abril) Material del Curso CI-2657. En Lista de Presentaciones Obtenido de <http://www.kramirez.net/ci-2657/materialci2657/>
- [7] COSTARICAMAKERS. (2018, abril) Tag Archives: ROS. YA CONTAMOS PERSONAS... AHORA HAGAMOS ALGO CON PYTHON Obtenido de <http://costaricamakers.com/?tag=ros>
- [8] Valverde, S. (2015). “Robótica inteligente: Implementación de sensores 3D para desenvolvimiento de robots móviles y vehículos autónomos” Obtenido de https://repositoriotec.tec.ac.cr/bitstream/handle/2238/6932/robotica_inteligente_implimentaci%C3%B3n_sensores_desenvolvimiento.pdf?sequence=1&isAllowed=y.
- [9] Rosas, L., Fuentes, M., Samaniego, C., Alvarado, R., & Valencia, J. (2013). MODELADO Y CONTROL DEL ROBOT MÓVIL ROBOTNIK SUMMIT XL. Obtenido de <http://cerescontrols.com/wp-content/uploads/2013/10/SUMMIT-XL-Informe-Final.pdf>
- [10] Justina (2017, mayo) Justina. En user_manual Obtenido de https://github.com/RobotJustina/JUSTINA/blob/master/user_manual/user_manual.pdf

- [11] Eagle X (2018, marzo) Un robot construido por alumnos del TEC. En Vinculación y prestigio Obtenido de <https://tec.mx/es/noticias/queretaro/vinculacion-y-prestigio/un-robot-construido-por-alumnos-del-tec>
- [12] Juárez, G. (2008) IMPLEMENTACIÓN DE UN LABORATORIO VIRTUAL CON LA AYUDA DE LABVIEW, AL CURSO DE CIRCUITOS ELÉCTRICOS 1. Obtenido de http://biblioteca.usac.edu.gt/tesis/08/08_0148_ME.pdf
- [13] Prieto, R., Zaldivar, U., & Bernal, R. (2010) Creación de un Laboratorio Virtual para Optimizar el uso de un Laboratorio de Robótica Real. Obtenido de https://www.academia.edu/374898/Creaci%C3%B3n_de_un_Laboratorio_Virtual_para_Optimizar_el_uso_de_un_Laboratorio_de_Rob%C3%B3tica_Real
- [14] Ayala, J., Pupo, L., & Salazar, L. (2016) LABORATORIO VIRTUAL DE INSTRUMENTACIÓN Y CONTROL Obtenido de <http://www.informaticahabana.cu/sites/default/files/ponencias/EDU095.pdf>
- [15] Owen, A. (14 de marzo de 2016). What is the Best Programming Language for Robotics? [Entrada en un blog] Robotiq. Recuperado de <https://blog.robotiq.com/what-is-the-best-programming-language-for-robotics>
- [16] Tellez, R. (2017). A thousand robots for each student: Using cloud robot simulations to teach robotics. En *Robotics in Education*, 143-155. Springer, Cham.
- [17] MoveIt! (s. f.) Concepts. En Documentacion Obtenido de <http://moveit.ros.org/documentation/concepts/>
- [18] Corke, P. I. (1996). A robotics toolbox for MATLAB. *IEEE Robotics & Automation Magazine*, 3(1), 24-32. doi 10.1109/100.486658
- [19] ROS (s. f.) Core components. En Communications Infrastructure Obtenido de <http://www.ros.org/core-components/>
- [20] Charles River Analytics Inc (2018). [fotografía]. robot_localization Recuperado de <https://www.cra.com/work/case-studies/robotlocalization>
- [21] Pitzer, B., Osentoski, S., Jay, G., Crick, C., & Jenkins, O. C. (2012, May). Pr2 remote lab: An environment for remote development and experimentation. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on* 3200-3205. IEEE.
- [22] Casañ, G. A., Cervera, E., Moughlbay, A. A., Alemany, J., & Martinet, P. (2015). *ROS-based online robot programming for remote education and training. In Robotics and Automation (ICRA), 2015 IEEE International Conference on* 6101-6106.

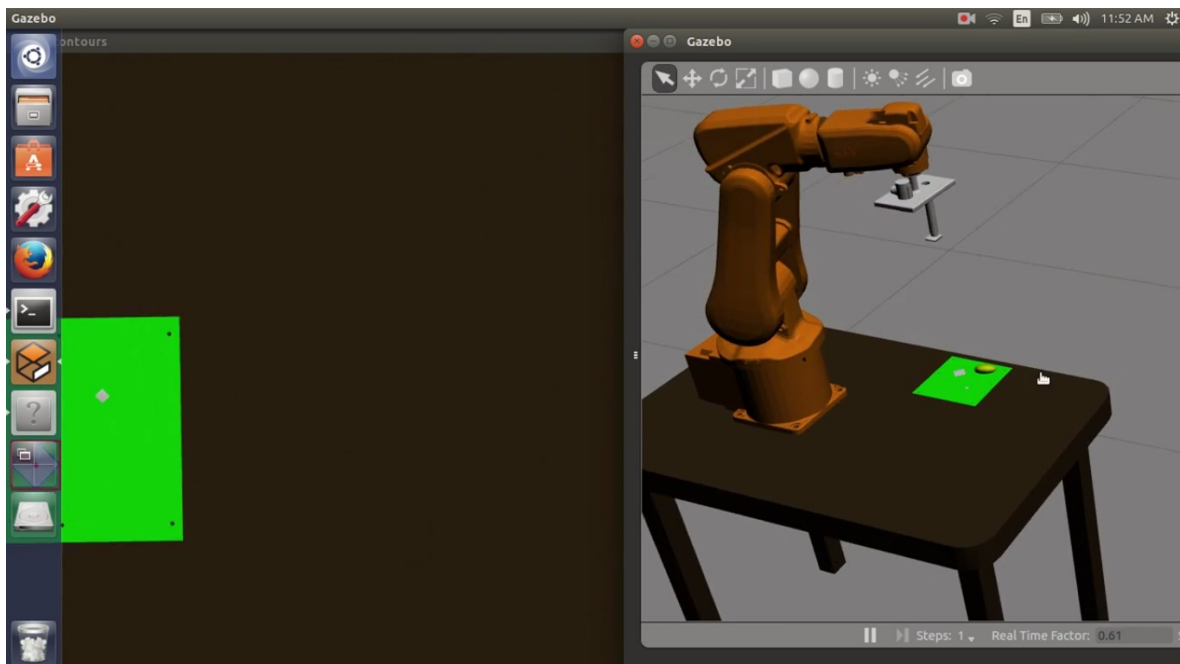
- [23] Universidad Tecnológica La Salle (2017) Ingeniería en Mecatrónica y Sistemas de Control. En Plan de estudio Obtenido de <http://www.ulsal.edu.ni/index.php/ingenieria-en-mecatronica-y-sistemas-de-control>
- [24] Calvo, I., Zulueta, E., Gangoiti, U., López, J. M., Cartwright, H., & Valentine, K. (2009). *Laboratorios remotos y virtuales en enseñanzas técnicas y científicas* (Vol. 3, No. 3, pp. 1-21). Ikastorratza.
- [25] Pérez, X., Cerda, A., & Incer, W. (2016) Desarrollar un laboratorio virtual para la realización de prácticas de laboratorio en la disciplina de control automático y automatización industrial
- [26] ROS Industrial (2016, noviembre) ROS QTC Plugin. En Repositorio Obtenido de https://github.com/ros-industrial/ros_qtc_plugin
- [27] ROS-Industrial (2017, Marzo) En ROS-Industrial Obtenido de <http://wiki.ros.org/Industrial/Tutorials>
- [28] Willow Garage. (2010, Abril) En Comic: Reinventing the Wheel [Entrada en un blog] Obtenido de <http://www.willowgarage.com/blog/2010/04/27/reinventing-wheel>
- [29] Merino, E. (2015) DISEÑO DE UN SIMULADOR DE COMPILADOR PARA PLATAFORMA MOODLE E IMPLEMENTACIÓN DE UN LABORATORIO VIRTUAL PARA LA ENSEÑANZA DE PROGRAMACIÓN.
- [30] Guerrero, L., Gómez, D., Sandoval, E., Thomson, P., Marulanda, J. (2014). SISMILAB, UN LABORATORIO VIRTUAL DE INGENIERÍA SÍSMICA, Y SU IMPACTO EN LA EDUCACIÓN.
- [31] ROBOTIS Co., Ltd. (2017). [fotografía]. *ROS Robot Programming*. Obtenido de <https://community.robotsource.org/t/download-the-ros-robot-programming-book-for-free/51>

Anexos

Trabajos relacionados con ROS y Gazebo.

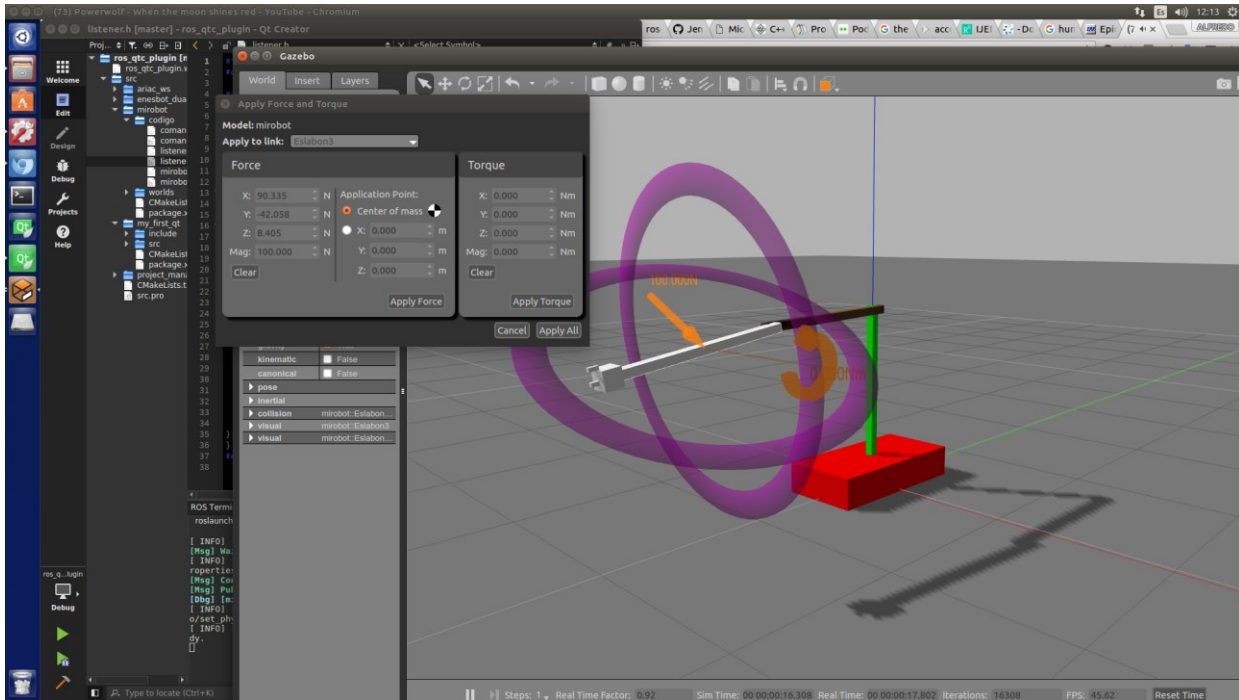


A.1: Captura de pantalla trabajos relacionados en simulación usando ROS y Gazebo para practica de Pick and Place.



A.2: Captura de pantalla trabajos relacionados en simulación usando ROS y Gazebo para practica de Visión simulada.

Pruebas realizadas con ROS y Gazebo.



A.3: Captura de pantalla Pruebas realizadas en simulación usando ROS y Gazebo para practica con un robot simple de 4 grados de libertad.