

HERRAMIENTAS VIRTUALES PARA LA SIMULACION Y ESTUDIO DE MOVIMIENTO DEL ROBOT KUKA YUBOT

VIRTUAL TOOLS FOR THE SIMULATION AND MOVEMENT STUDY OF THE ROBOT KUKA YUBOT

Erika Berenice Lázaro Flores

TecNM / Institución Tecnológico de Ciudad Guzmán, México
erika.lazflo@gmail.com

Jaime Jalomo Cuevas

TecNM / Instituto Tecnológico de Ciudad Guzmán, México
jaimejc@itcg.edu.mx

Recepción: 14/septiembre/2019

Aceptación: 2/octubre/2019

Resumen

Se presenta las herramientas actuales de simulación de movimiento de sistemas robóticos, en este caso un robot móvil manipulador, en específico, como objeto de estudio es el Kuka youBot, para su análisis y comparar la eficiencia del estudio realizado. En general para realizar la simulación de movimiento ROS, Gazebo, en Linux, se presentan como únicos software para poder realizarlo, pero se tiene poca accesibilidad al sistema operativo ya que es muy costoso, se requiere tener desde una computadora de un tipo específico y conocimientos de programación de código más avanzado, dejando a los usuarios sin opciones o con mucha limitación; pero no es así, en Windows se tienen opciones, en este caso se presenta el uso del software Blender, de código abierto, también VRML-Matlab y Simulink-SimMechanics, donde se puede trabajar desde el modelado, pasando por el proceso de simulación y puedes agregar controles al modelo, no necesitas una computadora especial ni tampoco ser un experto para poder entrar en el mundo de la simulación.

Palabra(s) Clave: Simulación, movimiento, Blender, VRML, Sim, SimMechanics.

Abstract

In this paper is presented the current simulation tools of robotic systems movement, in this case a mobile robot manipulator, specifically, as the object of study is the YouBot, for analysis and compare the study.

In general to perform the simulation of ROS movement, Gazebo, in Linux, is presented as the only software to be able to do it, but it has little access to the operating system since it is very expensive, it is required to have from a computer of a specific type and knowledge more advanced code programming, leaving users without options or with much limitation; but this is not the case, in Windows there are options, in this case the use of Blender software, open source, also VRML-Matlab and Simulink-SimMechanics, where you can work from modeling, through the simulation process and you can add controls to the model, you don't need a special computer or be an expert to enter the world of simulation.

Keyword(s): *Simulation, movement, Blender, VRML, Sim, SimMechanics.*

1. Introducción

Tanto en los robots manipuladores como en la robótica móvil existen puntos de interés común: el modelado cinemático, el modelado dinámico, el control (arquitecturas, algoritmos...), la planificación, el reconocimiento del entorno, etc.

La mayor parte de los robots manipuladores son brazos articulados tradicionalmente se modelan, desde el punto de vista cinemático, con matrices de transformación homogénea entre sistemas de coordenadas. Para el modelado dinámico se manejan distintas formulaciones y métodos: Lagrange-Euler, Newton-Euler, etc.

Para mejorar sus prestaciones, se investiga en técnicas para identificar los modelos dinámicos eficientemente y en métodos de control de articulaciones que compensan no-linealidades y acoplamientos, así como en optimización dinámica y control adaptativo para distintas condiciones de trabajo.

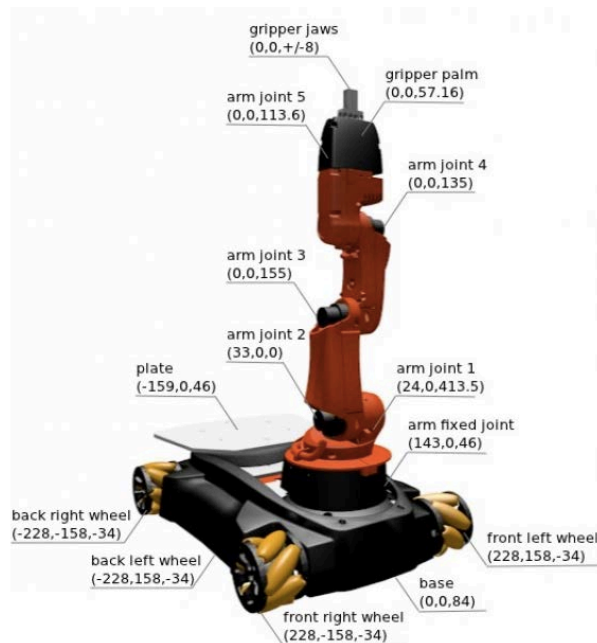
El objeto de estudio es el Kuka youBot es un robot manipulador móvil omnidireccional desarrollado para fines académicos y de investigación. Es una plataforma móvil omnidireccional en el que está instalado un brazo de robot de cinco ejes con una pinza de dos dedos, Kuka youBot cuenta con una computadora abordo que usa el sistema operativo Ubuntu, cuenta con una API (Application Programming Interface) desarrollada en C++, además se puede usar ROS (Robot Operating System). El robot tiene como característica de manejarse sobre superficies planas con desniveles de hasta 10% de su longitud horizontal [Spyros, 2009].

En concreto como no se cuenta con el objeto de estudio en físico, el enfoque a la simulación de la cinemática y dinámica es importante, así podemos comprobar la veracidad del modelo matemático y del comportamiento que tendrá el sistema físico. Sin embargo, hemos investigado que se pueden utilizar otras interfaces, como las de Windows, mencionadas en el resumen, las cuales nos permiten poder estudiar el modelo.

2. Métodos

Modelo cinemático del Kuka youBot

En la primera parte se realizó el estudio el modelo cinemático del robot Kuka youBot, figura 1 [Spyros, 2009].



Fuente: Elaboración propia

Figura 1 Robot Kuka youBot.

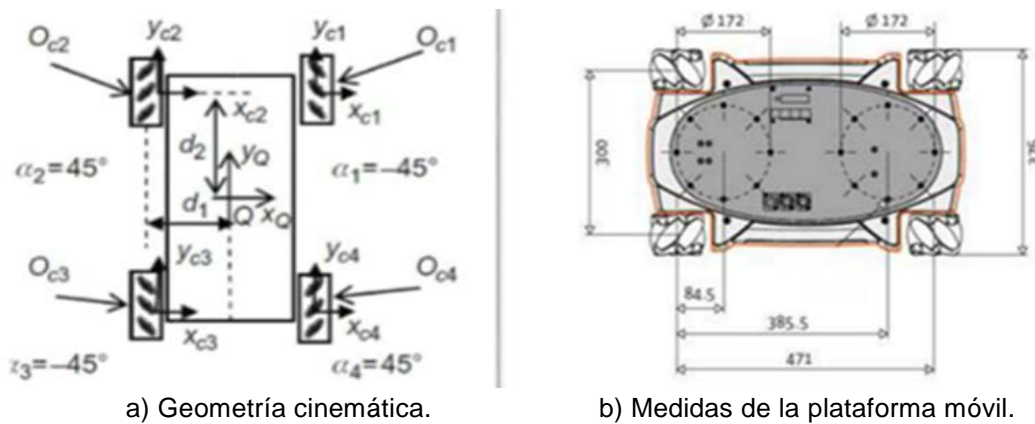
El modelo cinemático se dividió en:

- Modelo cinemático de la plataforma omnidireccional, en este incluye el modelo del tipo de rueda que se utiliza el Kuka youBot, tipo Mecanum.
- Modelo del brazo manipulador, realizado por el algoritmo Denavit-Hartenberg.

En la segunda parte se encuentra el Modelado y simulación de la cinemática y movimiento del YouBot en Blender, V-Real Builder, Matlab-Simulink-SimMechanics, se incluyen las pruebas realizar en los diferentes softwares

Modelo cinemático de la plataforma omnidireccional

Kuka YouBot cuenta con una plataforma omnidireccional de cuatro ruedas tipo Mecanum con libre desplazamiento en el plano cartesiano, pero es necesario considerar la disposición de los rodillos de este tipo de ruedas para obtener un modelo cinemático fidedigno (Figura 2) [Spyros, 2009] y [Nagatani,2000].



Fuente: Spyros 2009

Figura 2 Plataforma móvil.

La configuración cinemática de un robot móvil omnidireccional está dada por las ecuaciones 1, 2 y 3.

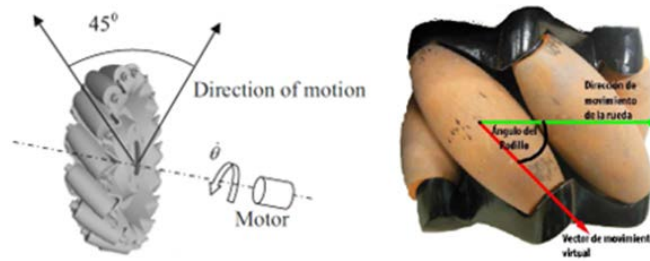
$$\dot{x} = v_l = \frac{1}{4}(d_1 + d_2 + d_3 + d_4) \quad (1)$$

$$\dot{y} = v_t = \frac{1}{4}(-d_1 + d_2 + d_3 - d_4) \tan(\alpha_b) \quad (2)$$

$$\dot{\theta}_b = v_a = w = \frac{1}{4}(d_1 + d_2 + d_3 - d_4)\beta \quad (3)$$

\dot{x} y \dot{y} son la velocidad de los ejes X y Y , o bien las velocidades longitudinal y transversal (v_l y v_t) mientras que $\dot{\theta}_b$ es la velocidad angular (v_a) de la plataforma móvil omnidireccional. d_i es la velocidad lineal de cada rueda de la plataforma móvil.

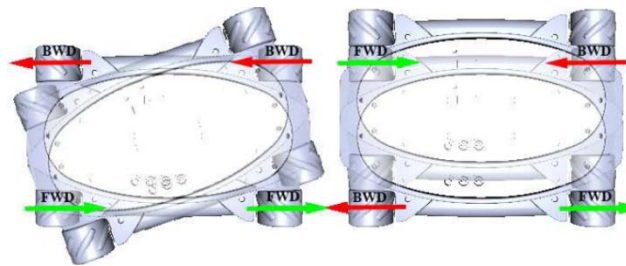
Los parámetros α_b y β son experimentalmente calculados debido a que dependen del ángulo en que estén dispuestos los rodillos dentro de las ruedas Mecanum (Figura 3) [Nagatani,2000].



Fuente: Nagatani, 2000.

Figura 3 Rueda tipo mecanum.

El movimiento del robot móvil en el plano depende de las velocidades lineales de cada una de las ruedas de la plataforma en los movimientos de traslación transversal y rotación. En la figura 4 donde se ve difuminada la plataforma indica hacia donde se moverá el robot móvil según las velocidades lineales de las 4 ruedas [Nagatani,2000].



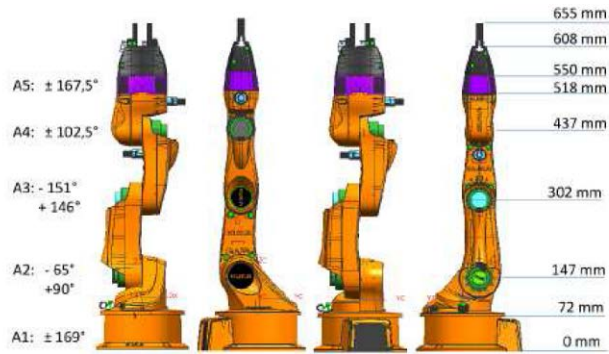
Fuente: Nagatani, 2000.

Figura 4 Movimiento de la plataforma móvil

Modelo cinemático del manipulador

El brazo manipulador del Kuka YouBot es un robot de 5 grados de libertad (todos con ángulo de rotación), las medidas de cada eslabón, así como sus límites articulares se pueden observar en la siguiente figura. El efector final es una pinza de dos dedos que permite sujetar y manipular dedos pequeños.

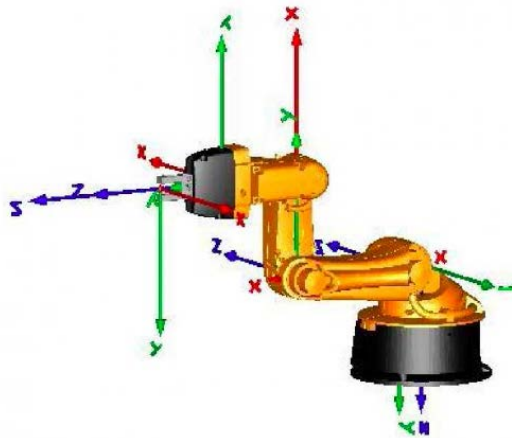
El modelo cinemático del robot manipulador se obtiene mediante el algoritmo Denavit-Hartenberg (Figura 6) [Lipkin, 2005].



Fuente: Elaboración propia

Figura 6 Manipulador del Kuka .YouBot

Los marcos de referencia de cada articulación se pueden apreciar en la siguiente figura (Figura 7) y los parámetros de obtenidos, las tablas 1 y 2 muestran distancias y ángulos entre marcos de referencia para el brazo manipulador [Lipkin, 2005].



Fuente: Elaboración propia

Figura 7 Posicionamiento de los marcos de referencia para cada articulación.

Tabla 1 Parámetros D-H para el manipulador del YouBot

Eslabón	θ	d	a	α
1	q_1	0.147	0.0330	$\frac{\pi}{2}$
2	q_2	0	0.1550	0
3	q_3	0	0.1350	0
4	$q_4 + \frac{\pi}{2}$	0	0	$\frac{\pi}{2}$
5	q_5	0.2175	0	0

Fuente: Elaboración propia

Tabla 2 Cadena cinemática para el manipulador del YouBot

	Marco Anterior	Traslación [cm]			Rotación [grados]		
		x	y	z	x	y	z
Articulación 1	Base	2.4	0	11.5	180°	0°	0°
Articulación 2	Articulación 1	3.3	0	0	90°	0°	-90°
Articulación 3	Articulación 2	15.5	0	0	0°	0°	-90°
Articulación 4	Articulación 3	0	13.5	0	0°	0°	0°
Articulación 5	Articulación 4	0	11.36	0	-90°	0°	0°
Pinza	Articulación 5	0	0	5.716	90°	0°	180°

Fuente: Elaboración propia

Con los parámetros anteriores es posible obtener las matrices de transformación homogéneas, ecuaciones 4 y 5.

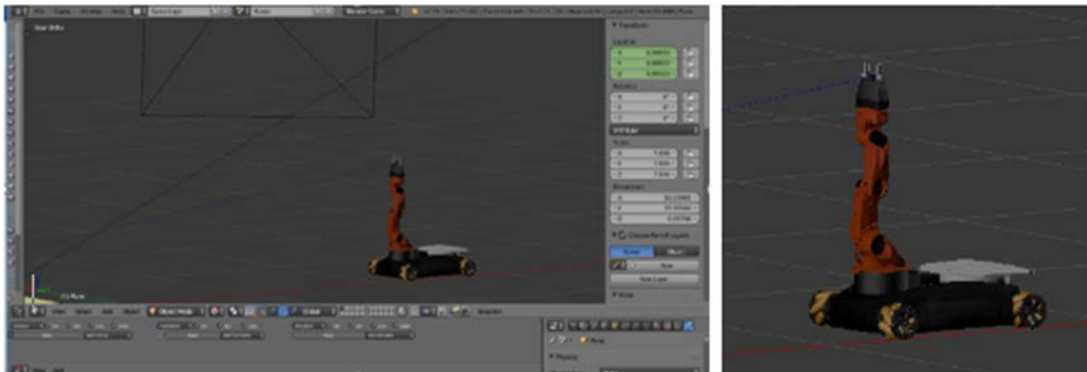
$$(\mathbf{A}_i^{i-1}) = \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & -S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$\mathbf{T}_n^0 = \mathbf{A}_1^0 \dots \mathbf{A}_n^{n-1} \quad (5)$$

Los parámetros relacionados con la dinámica nos la indica la posición relativa del centro de masa y el momento de inercia del tensor [Lipkin, 2005], [Spyros, 2009]. Modelado y simulación de la cinemática y movimiento del YouBot en Blender, V-Real Builder, Matlab-Simulink-SimMechanics:

- **Descripción.** Cada cuerpo rígido tiene una posición relativa y una orientación con respecto a su marco base. La orientación se describe en los ángulos de Euler. La convención de rotación compuesta es Yaw Pitch Roll (ZYX). Tipo de junta describe el tipo de una restricción, donde el tipo puede ser uno de los siguientes:
 - ✓ Revolute. Una articulación de bisagra que gira a lo largo del eje y tiene un rango limitado especificado por los límites superior e inferior;
 - ✓ Continua. Una articulación de bisagra continua que gira alrededor del eje y no tiene límites superiores e inferiores;
 - ✓ Prismático. Una junta deslizante que se desliza a lo largo del eje, y tiene un alcance limitado especificado por los límites superior e inferior. El eje de la articulación es el eje de rotación de la articulación revoluta o el eje de la traslación para la articulación prismática.

- **Modelado en Blender.** Es un programa informático multi plataforma, dedicado especialmente al modelado, iluminación, renderizado, animación y creación de gráficos tridimensionales. También de composición digital utilizando la técnica procesal de nodos, edición de vídeo, escultura (incluye topología dinámica) y pintura digital. En Blender, además, se pueden desarrollar vídeo juegos ya que posee un motor de juegos interno. software libre. Es un software libre y actualmente es compatible con todas las versiones de Windows, Mac OS X, GNU/Linux (Incluyendo Android), Solaris, FreeBSD e IRIX. En la Figura 8 se presenta el modelado del Kuka youBot.



Fuente: Elaboración propia

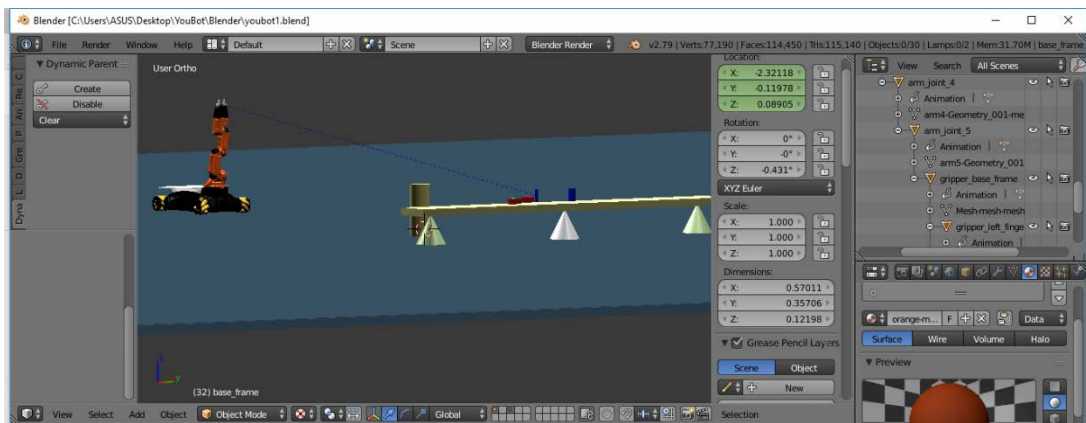
Figura 8 Modelado del Kuka youBot en Blender.

En el BGE (Blender Game Engine) o motor de juego te permite representar imágenes y sonidos especializados, simular movimientos físicos, colisiones, etc., de objetos y controlar la lógica mediante componentes lógicos propiedades y programación en Python, **todo en tiempo real**.

En figura 9 se presenta a Kuka youBot en el BGE de Blender. En figuras 10 a 13, se muestra la simulación de movimiento realizada, utilizando drivers y respetando la cinemática del modelo. En figuras 14 y 15 se muestra la simulación realizada con BGE el motor Logic Editor, donde se puede hacer todos los movimientos del sistema por medio de una lógica realizada, la cual incluye, sensores, controladores tipo lógicos y actuadores, que en este caso son los joints del robot móvil con brazo manipulador. Para manejar el robot en la simulación se programó para que se hiciera por medio del teclado.



Fuente: Elaboración propia
Figura 9 Kuka YouBot en BGE.



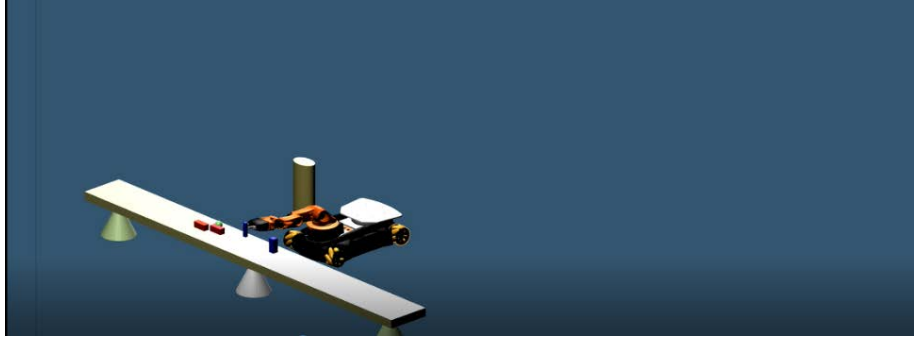
Fuente: Elaboración propia

Figura 10 Entorno de la simulación.



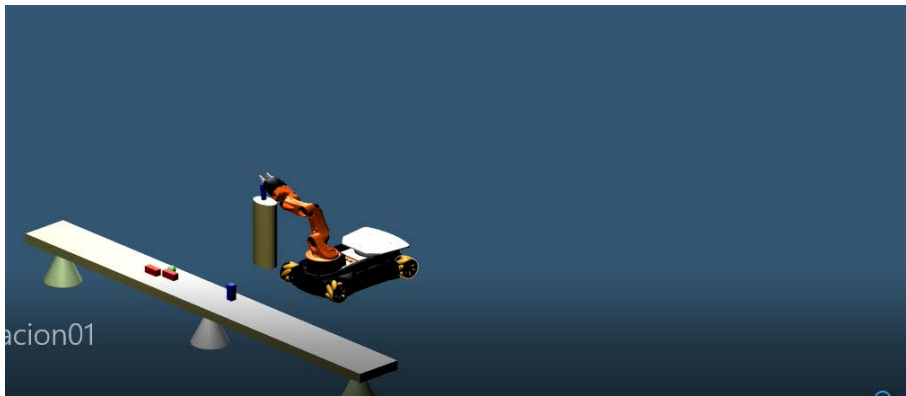
Fuente: Elaboración propia

Figura 11 Parte de la secuencia de la simulación.



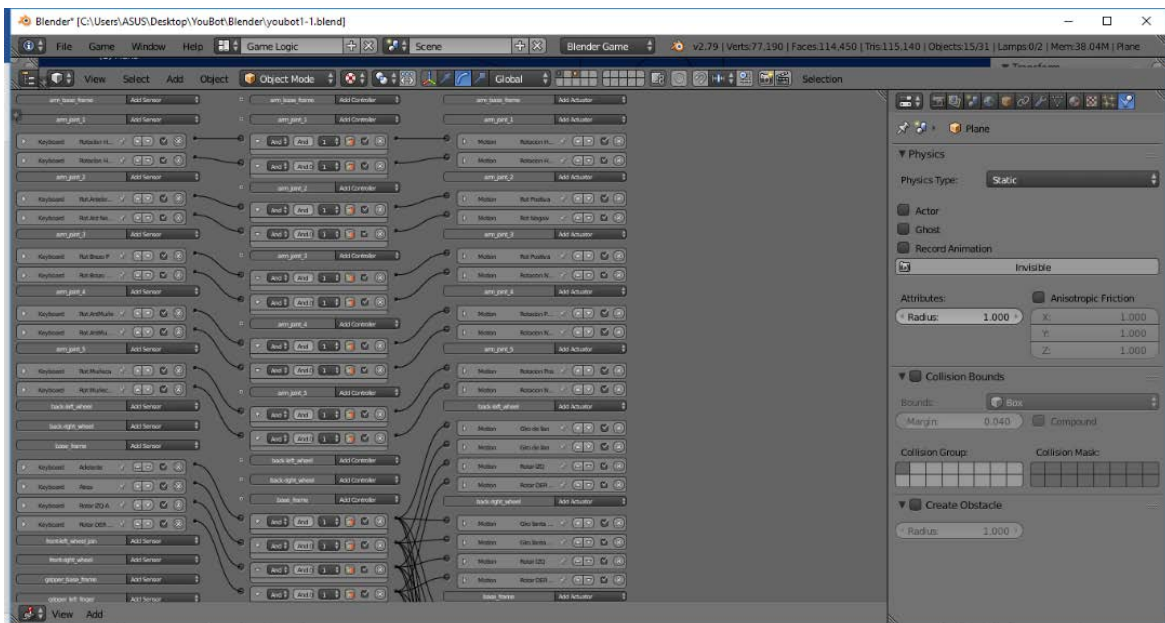
Fuente: Elaboración propia

Figura 12 Parte de la secuencia de simulación de movimiento en la pinza toma un objeto.



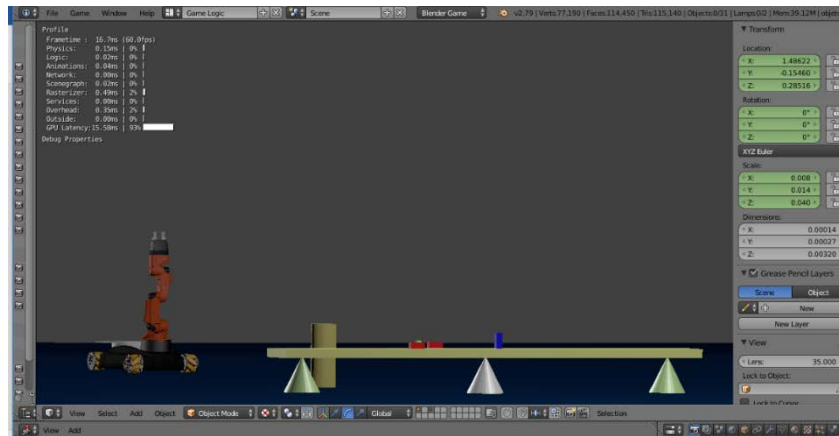
Fuente: Elaboración propia

Figura 13 Parte final de secuencia de simulación de movimiento, deja objeto en otro lugar.



Fuente: Elaboración propia

Figura 14 Lógica realizada en el BGE de Blender.



Fuente: Elaboración propia

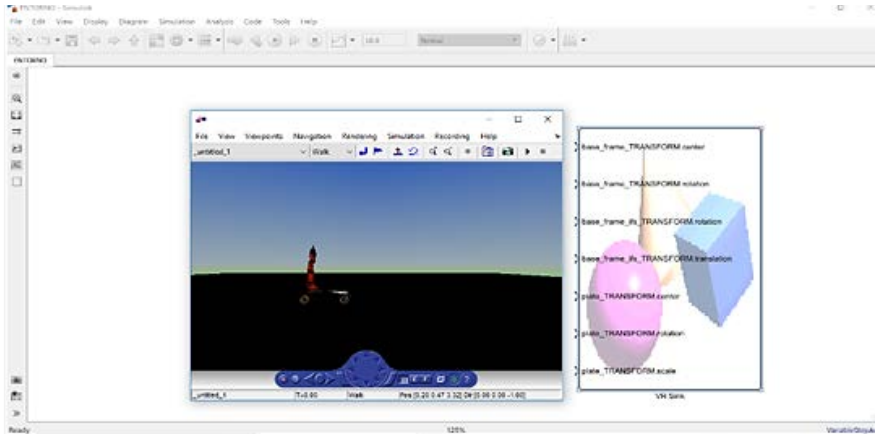
Figura 15 Lógica de simulación, control del Kuka YouBot por teclado.

- **V- Real Builder (VRML) – Simulink.** V-Realm Builder es un poderoso paquete de creación tridimensional para la creación de objetos tridimensionales y "mundos" en plataforma Matlab para ser vistos con el navegador V-Realm o cualquier otro navegador compatible con VRML 2.0.

Los campos de cada nodo son editables directamente con retroalimentación inmediata en la escena para las modificaciones hechas a cualquier variable o nodo. Muchas de las funciones de edición se completan con un solo clic del mouse, el uso del teclado está limitado solo a las funciones que requieren entrada.

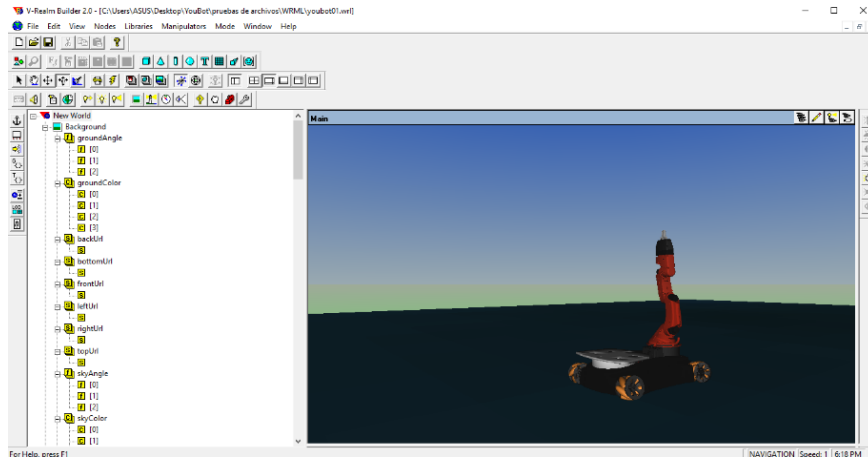
Simulink es un software que funciona bajo la plataforma de Matlab y es una herramienta muy útil para modelar, simular y analizar sistemas, tanto lineales como no lineales. Permite al usuario realizar sus estudios tanto en el dominio del tiempo como el de Laplace, expresar las funciones de transferencia en las diferentes formas incluyendo la del espacio de los estados y otras opciones. Para este caso el objeto de estudio se realizó con un entorno sencillo, el cual se compone del YouBot y un suelo por donde se desplazará. En Simulink se accede por medio de un bloque ese "mundo" que ha sido realizado. Una vez realizado en VRML-Builder se asigna a un bloque específico en Simulink de Matlab, para programarlo de modo que pueda simular el entorno realizado, entonces, desde Simulink puedes agregar los controles necesarios. En la figura 16 se muestra como los dos programas se

enlazan para formar la interface y en figura 17 se muestra como se ve el sistema físico, refiriéndose al Kuka youBot dentro del mundo de VRML-Builder.



Fuente: Elaboración propia

Figura 16 Simulink- Entrando al sistema “mundo” VRML-Builder.



Fuente: Elaboración propia

Figura 17 “Mundo” VRML. Sistema físico YouBot.

D.SolidWorks- SimMechanics

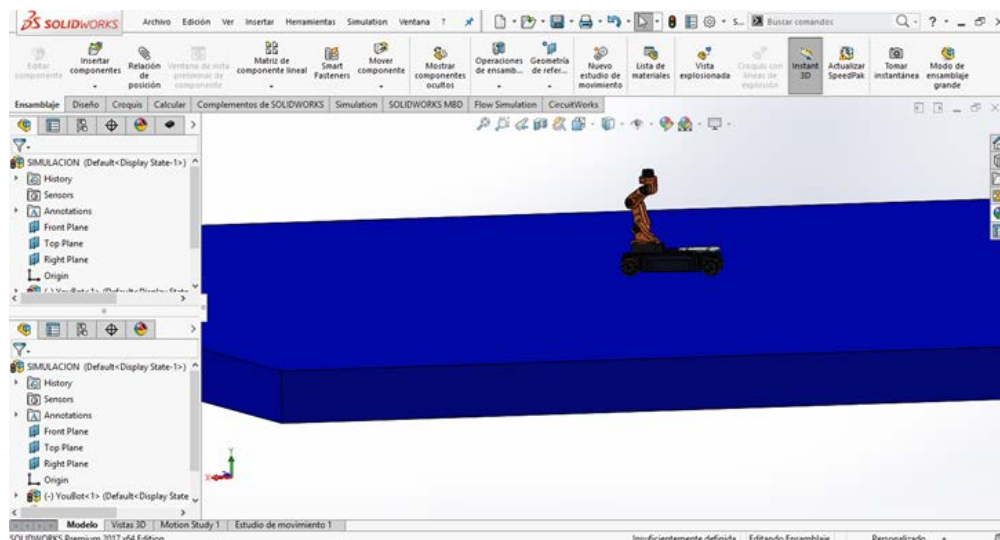
Simscape Multibody™ (antes SimMechanics™) proporciona un entorno de simulación multicuerpo para 3D sistemas mecánicos, como robots, suspensiones de vehículos, maquinaria y tren de aterrizaje de aviones. Puede modelar usando los bloques que representan órganos, articulaciones, limitaciones, elementos de fuerza y sensores de sistemas multicuerpo. Multicuerpo Simscape formula y resuelve las

ecuaciones de movimiento para el sistema mecánico completo. Puede importar conjuntos de CAD completadas, incluyendo todas las masas, inercias, juntas, restricciones y geometría 3D, en el modelo. Una animación 3D generada automáticamente le permite visualizar la dinámica del sistema.

Primero se realiza el modelo en SolidWorks, una vez realizado se exporta a SimMechanics, pero para realizar este movimiento se debe enlazar los dos programas, una vez hecho esto con una orden de tipo código en Matlab se manda llamar para que lo pueda ver el diagrama de bloques en SimMechanics y en él poder aplicar control. Se continúa con las pruebas de simulación en el entorno de Motion Study de SolidWorks y por lo pronto se están haciendo pruebas del enlace para abrir los diagramas de bloque la plataforma y el brazo móvil por separado, se muestran en figuras 18 a 20.

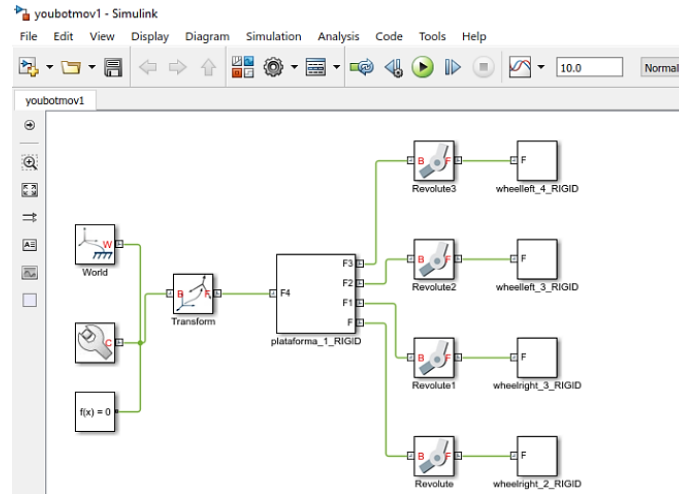
3. Resultados

En el primer caso de Blender está completa la simulación, aunque pueden realizarse mejoras. De forma interna Blender calcula matemáticamente la posición que se encontrará. Respeta en todos los sentidos la geometría y cinemática del objeto, ya que, si sales fuera de los parámetros o si estas aplicando un resultado de manera incorrecta se notará en el resultado de la simulación.



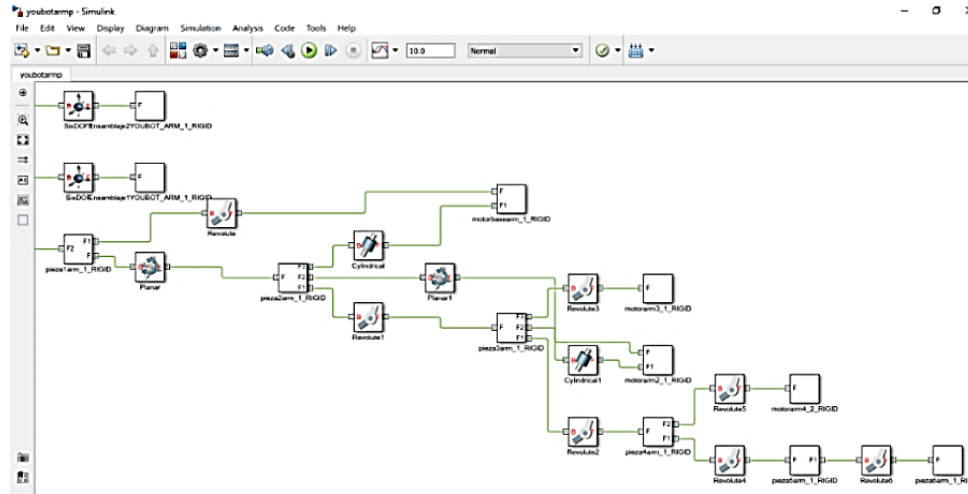
Fuente: Elaboración propia

Figura 18 Entorno y sistema físico YouBot en SolidWorks.



Fuente: Elaboración propia

Figura 19 Sistema físico en SimMechanics parte móvil.

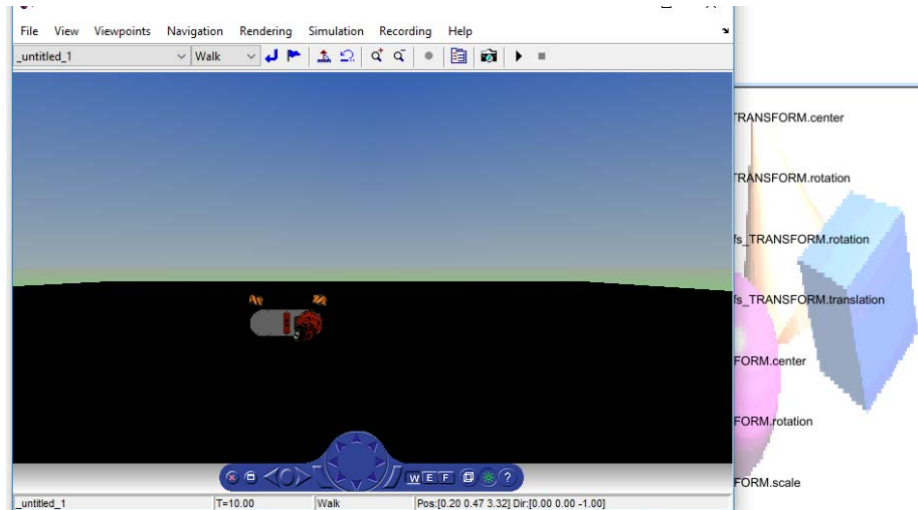


Fuente: Elaboración propia

Figura 20 Sistema físico en SimMechanics parte móvil y el brazo manipulador YouBot.

VRML parece ser fácil demasiado amigable pero muy “antiguo”, aunque para su uso en Simulink es esencial, se continuará trabajando en el desde Simulink, ahí para trabajar en la simulación utilizando controles iniciando por controles PID. Como se puede ver en la siguiente figura (Figura 21) sin controles la simulación de movimiento no es completa.

En el caso en SolidWorks se tiene en un 85% por ciento el modelado del objeto ya que faltan de realizar el gripper (efecto final), el enlace esta correcto entre éste y Simulink.



Fuente: Elaboración propia

Figura 21 Kuka youBot sin controles desde Simulink de Matlab.

4. Discusión

Como podemos observar existen opciones en Windows que son más accesibles, e incluso más eficientes en comparación con Linux-Gazebo, claro no despreciando la eficacia de este dentro de su sistema, a continuación, se describe lo encontrado en cada uno de ellos:

Blender es un software muy amable con controles muy amigables son fáciles de entender, puedes simular en tiempo real y respeta los parámetros del modelado cinemático. Si existen errores te los marca y puedes ver cómo le afectan al modelo. VRLM-Builder es un poco ambiguo no te deja manejar al modelo con libertad y el modelado es algo tedioso de realizar, pero en contraste junto con Matlab-Simulink lo hacen una herramienta muy poderosa.

SolidWorks es muy eficiente, sobre todo en el enlace con SimMechanics, para el estudio de controles para los sistemas robóticos, pero si requiere de ciertas bases de conocimientos de diseño y programación, se continuará con el presente objeto de estudio para completar la simulación.

Se espera continuar las pruebas en Motion Study de SolidWorks; se debe completar el modelo en SimMechanics, la parte móvil y la parte del brazo manipulador del Kuka youBot una vez exportado de SolidWorks, para agregar los controles y poder simular y completar la interface desde Matlab.

5. Bibliografía y Referencias

- [1] Khalil, Hassan. *Nonlinear Systems*. Pearson, Estados Unidos. 1996.
- [2] Isidori, Alberto. *Non Linear Control Systems*, Springer Verlag, Alemania. 1989.
- [3] Knobloch, Hans. *Topics in Control Theory*, Birkhauser Verlag, Suiza. 1994.
- [4] Spyros G. Tzafestas *Web-Based Controls and Robotic Educations*, Athens, Greece, 2009. Volumen 38.
- [5] Dwiputra, R., Zakharov, A., Chakirov, R., and Prassler, E. Modelica model for the youbot manipulator. In *Proceedings of the 10th International Modelica Conference*, March 10-12, 2014, Lund, Sweden, 1205–1212. doi:10.3384/ECP140961205.
- [6] Nagatani, K., Tachibana, S., Sofne, M., and Tanaka, Y. Improvement of odometry for omnidirectional vehicle using optical flow information. In *Intelligent Robots and Systems, 2000. (IROS 2000). Proceedings. 2000 IEEE/RSJ International Conference on*, volumen 1, 468–473 vol.1. doi:10.1109/IROS.2000.894648.
- [7] Lipkin, H. (2005). A note on Denavit-Hartenberg notation in robotics. In *29th Mechanisms and Robotics Conference*, volume 7, 921–926. doi:10.1115/DETC2005-85460.