



**UNIVERSIDAD DE GUANAJUATO**

---

CAMPUS IRAPUATO-SALAMANCA  
DIVISIÓN DE INGENIERÍAS

*“Extensión 3D de un telémetro 2D”*

**TESIS PROFESIONAL**

QUE PARA OBTENER EL TÍTULO DE:

***INGENIERO EN MECATRÓNICA***

PRESENTA:

*Jesús Abrahan Estrada Salazar*

DIRECTOR:

*Dr. Víctor Ayala Ramírez*

SALAMANCA, GTO.

Diciembre 2017

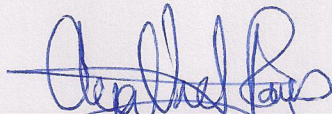
Salamanca, Gto., a 27 de Noviembre del 2017.

**DR. DONATO HERNÁNDEZ FUSILIER**  
**COORDINADOR DE ASUNTOS ESCOLARES**  
**PRESENTE.-**

Por medio de la presente, se otorga autorización para proceder a los trámites de impresión, empastado de tesis y titulación al alumno(a) Jesús Abraham Estrada Salazar del Programa de Licenciatura en Ingeniería en Mecatrónica y cuyo número de **NUA** es: 387451 del cual soy director. El título de la tesis es: Extensión 3D de un telémetro 2D

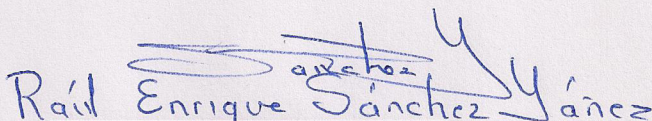
Hago constar que he revisado dicho trabajo y he tenido comunicación con los sinodales asignados para la revisión de la tesis, por lo que no hay impedimento alguno para fijar la fecha de examen de titulación.

**ATENTAMENTE**

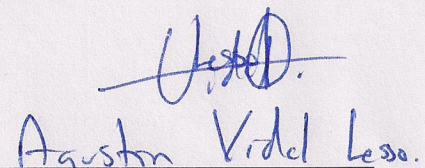


\_\_\_\_\_  
NOMBRE Y FIRMA  
**DIRECTOR DE TESIS**  
**SECRETARIO**

\_\_\_\_\_  
NOMBRE Y FIRMA  
**DIRECTOR DE TESIS**



\_\_\_\_\_  
NOMBRE Y FIRMA  
**PRESIDENTE**



\_\_\_\_\_  
NOMBRE Y FIRMA  
**VOCAL**

Llenar en computadora con ayuda del oficio de modalidad.

La modalidad de tesis es única para los posgrados

Nivel:	
Licenciatura	X
Maestría	
Doctorado	

Marcar con una X

<b>Modalidad:</b>	Trabajo de tesis
-------------------	------------------

<b>Año:</b>	2017
-------------	------

Poner el número de año p.e. 2015

**Información sobre Obtención de Grado Académico:**

<b>Nombre</b>	Jesús Abrahan Estrada Salazar
<b>NUA</b>	387451
<b>Programa</b>	Ingeniería en Mecatrónica

Para modalidades con Jurado completar la siguiente información:

**Lugar, hora y fecha de la presentación**

<b>Lugar</b>	Auditorio 102
<b>Hora</b>	2:00 pm
<b>Fecha</b>	Martes 5 de Diciembre de 2017



<b>Título del trabajo</b>	Extensión 3D de un telémetro 2D
---------------------------	---------------------------------

UG - CAMPUS SALAMANCA

**Jurado**

	Nombre con grado académico completo: p.e. Doctor en Informática Industrial Nombre Apellido Paterno Apellido Materno	Firma de autorización para realización de examen de grado o titulación.
<b>Presidente</b>	Doctor en Ciencias Raúl Enrique Sánchez Yañez	
<b>Secretario</b>	Doctor en Automática e Informática Industrial Víctor Ayala Ramírez	
<b>Vocal (1)</b>	Doctor en Ingeniería Mecánica Agustín Vidal Lesso	
<b>Vocal 2 (Doctorado)</b>		
<b>Vocal 3 (Doctorado)</b>		

**Asesoría**

<b>Director del trabajo</b>	Dr. Víctor Ayala Ramírez
<b>Codirector</b>	

(No llenar para uso exclusivo de la Coordinación.)

Valida (nombre y firma):
--------------------------

Una vez terminado de llenar imprimir en dos tantos (uno para entregar al iniciar el trámite de autorización del examen de grado o titulación y otro para firma de recibido).

---

*Dedicado a mi familia*

---

## AGRADECIMIENTOS

---

A Dios, por permitirme vivir y poder concluir con la realización de este proyecto.

A **mis Padres Jesús y Catalina**, por su apoyo y confianza que me han otorgado a lo largo de mi preparación académica, así como la formación y valores que recibí y sigo recibiendo de ellos.

A **mis Hermanos Alan y Héctor**, que me han apoyado brindándome su afecto, energía y consejos a lo largo de mi formación académica.

Al **Dr. Víctor Ayala Ramírez**, por su apoyo, orientación y paciencia así como brindarme un espacio de trabajo y recursos para culminar con la realización de este proyecto.

Al **Dr. Raúl Enrique Sánchez Yáñez**, por sus consejos y observaciones, las cuales fueron muy fructíferas tanto para mi vida personal como para el presente trabajo.

Al **Dr. Agustín Vidal Lesso y Dr. Uriel Haile Hernández Belmonte**, por su tiempo, consejos y ayuda los cuales fueron esenciales para la realización de este trabajo.

A **mis compañeros del LaViRIA**, Andrea, Gesley, Diana, Alberto, Eduardo, Juan Pablo, José Jesús, Leonardo, Lubin y Marco por su amistad, consejos y apoyo. Gracias.

A **mis amigos** Cristina, Enrique, Israel, Luis, Manuel, Pedro y Rodrigo por su amistad y apoyo a lo largo de la carrera. Además a muchos que no menciono pero debieran estar aquí.

---

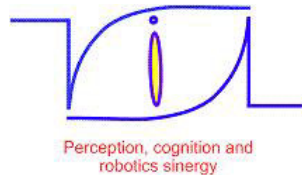
## AGRADECIMIENTOS INSTITUCIONALES

---

A la Universidad de Guanajuato, División de Ingenierías Campus Irapuato-Salamanca, por la formación académica obtenida y el apoyo recibido durante mi pertenencia en dicha institución como alumno en el programa de Licenciatura en Ingeniería Mecatrónica.



Al Laboratorio de Visión, Robótica e Inteligencia Artificial (LaViRIA), por la formación integral que recibí durante mi estancia en él, así como por el espacio de trabajo y los recursos proporcionados para la realización de este proyecto.



---

## ÍNDICE GENERAL

---

1	INTRODUCCIÓN	1
1.1	Justificación	1
1.2	Descripción del proyecto	2
1.3	Objetivos	2
1.4	Plan de la obra	2
2	MARCO TEÓRICO Y HERRAMIENTAS	4
2.1	Marco teórico	4
2.1.1	Reconstrucción 3D	4
2.1.2	Telémetro láser	5
2.1.3	Tipos de movimiento	7
2.1.4	Métodos de escaneo	8
2.1.5	Nube de puntos	9
2.2	Estado del Arte	10
2.3	Herramientas	11
2.3.1	ANSYS	12
2.3.2	Solidworks	12
2.3.3	Gazebo	12
2.3.4	ROS (Robot Operating System)	14
2.3.5	Matlab	14
2.4	Conclusiones del capítulo	15
3	DISEÑO MECÁNICO	16
3.1	Proceso de diseño	16
3.2	Reconocimiento de necesidades y definición de problema	17
3.3	Síntesis	18
3.4	Análisis	19
3.5	Conclusiones del capítulo	24
4	ENTORNOS DE SIMULACIÓN	25
4.1	Gazebo	25
4.2	ROS	26
4.2.1	Nivel archivos de sistema	27

4.2.2	Nivel gráfico de cómputo . . . . .	28
4.2.3	Nivel comunidad . . . . .	28
4.3	Descripción del Proyecto . . . . .	28
4.4	Descripción de Escenarios . . . . .	29
4.5	Conclusiones del capítulo . . . . .	31
5	RESULTADOS	32
5.1	Características de Simulación . . . . .	32
5.2	Prueba Conceptual: Primer escenario . . . . .	34
5.3	Prueba de Desempeño : Segundo escenario . . . . .	38
5.4	Conclusiones del capítulo . . . . .	53
6	CONCLUSIONES Y PERSPECTIVAS	54



---

## INTRODUCCIÓN

---

La extensión 3D de un telémetro láser 2D es un problema que ha sido estudiado en el campo de la robótica, y que busca obtener una representación de un entorno o escenario desconocido mediante el uso de estos sensores. También existen los telémetros láser 3D los cuales son capaces por sí mismos de obtener dicha representación sin necesidad de apoyo de otros sensores o dispositivos. La mayoría de los trabajos que buscan mapear o realizar la reconstrucción de un escenario utilizan telémetros láser 2D ya que estos tienen un costo más accesible que los telémetros láser 3D.

Algunos de los trabajos relacionados que hacen uso de telémetros láser son: Sheng *et al.* [1] describe el sistema de un robot móvil para SLAM. Morales *et al.* [2] propone un sensor 3D llamado UNOlaser integrado por un telémetro láser 2D provisto de movimiento. Wang *et al.* [3] presenta un robot móvil que mapea en 3D con dos telémetros láser 2D en tiempo real. Maurelli *et al.* [4] describe un sistema capaz de generar una nube de puntos 3D a partir de uno o dos telémetros láser 2D mientras un vehículo esta en movimiento.

### 1.1 Justificación

Reconstruir un escenario se puede lograr con sistemas de motores ya instrumentados que permiten el movimiento de un dispositivo para darle una función adicional. Este proyecto surge de la necesidad de darle ese extra mediante una plataforma diseñada para sustituir el sistema de motores y lograr una mayor precisión por medio de otros actuadores.

La población que se beneficia con la realización del proyecto va desde el público en general que quiera saber sobre el tema hasta estudiantes y profesores de ingeniería interesados en los temas de robótica móvil y visión por computadora.

## 1.2 Descripción del proyecto

Esta tesis tendrá como resultado esperado un sistema capaz de realizar la reconstrucción 3D de entornos o escenarios representados como una nube de puntos.

El desarrollo del sistema consistirá en el diseño, simulación, validación e implementación de los principales módulos que conformarán el sistema mencionado. Los requerimientos típicos que necesita un sistema de reconstrucción 3D de entornos o escenarios incluyen:

- La búsqueda de información en la literatura que de bases para las alternativas de los diferentes elementos que conformaran el sistema.
- La ideación de un modelo para el sistema teniendo como base el uso del telémetro láser disponible en el LaViRIA.
- La elección de un diseño que permita otorgar ese grado de libertad extra.
- La realización de software que permita la simulación del sistema.
- La validación de dicha simulación que mediante un mecanismo de evaluación que permita calificar el desempeño del sistema.

## 1.3 Objetivos

El objetivo general de esta tesis es el desarrollo de un sistema que permita obtener la representación 3D de un escenario como una nube de puntos. Esto incluye el diseño mecánico de la plataforma que sostendrá el telémetro láser, construir la plataforma mecánica diseñada, el desarrollo del software y la simulación de generación de nubes de puntos 3D adquiridos por el telémetro láser.

## 1.4 Plan de la obra

Este documento está estructurado de la siguiente forma:

- En el Capítulo 1 se muestra una introducción al problema de la representación 3D de escenarios representándolos como una nube de puntos.
- En el Capítulo 2 se presentan las bases teóricas y las herramientas seleccionadas para abordar el problema de la representación 3D de escenarios desconocidos.
- En el Capítulo 3 se muestra el proceso de diseño propuesto para solucionar el problema de la representación 3D de escenarios desconocidos. Se describe cada uno de los módulos esenciales considerados en este diseño.

- En el Capítulo 4, se describe el proceso de la simulación del sistema propuesto.
- En el Capítulo 5 se presentan los resultados de la simulación y su validación.
- En el Capítulo 6 se muestran las conclusiones y perspectivas generadas a partir del desarrollo de este trabajo.

---

## MARCO TEÓRICO Y HERRAMIENTAS

---

El problema de la reconstrucción 3D de entornos como una nube de puntos, requiere del conocimiento de conceptos básicos y uso de herramientas que faciliten la solución de dicho problema. En este capítulo se realizará una revisión de los conceptos relevantes así como de las herramientas que se usaron para el desarrollo del sistema. En este capítulo se definen los conceptos clave, algunos trabajos relacionados y las diferentes herramientas utilizadas.

### **2.1 Marco teórico**

Dentro de los conceptos relevantes se encuentran los siguientes: reconstrucción 3D, telémetro láser, nube de puntos, métodos de escaneo, tipos de movimiento de los cuales se describe una breve definición y factores de importancia para su comprensión.

#### **2.1.1 Reconstrucción 3D**

La reconstrucción 3D es un proceso en el que se realiza una reproducción de sus características físicas (volumen, forma y dimensiones). Existen diferentes técnicas que se aplican a la reconstrucción 3D que junto a métodos de mallado 3D permiten obtener un algoritmo que realiza la interconexión de los puntos presentes en la nube de puntos formando elementos de superficie como cuadrados y triángulos por mencionar algunos.

En la literatura existen diferentes procesos para realizar la reconstrucción 3D los cuales se pueden clasificar en 5 grupos:

- Técnicas multivistas: mediante este proceso se puede extraer la información tridimensional por medio de la correspondencia de la información bidimensional de dos o más captadores de imágenes [5].
- Cámara móvil: de forma similar este proceso extrae la información 3D, ahora mediante el flujo de imagen obtenido por un sensor, a este se le conoce como flujo de velocidades de la cámara. Los parámetros de las cámaras continuamente cambian por lo que es no es posible realizar una calibración [6].
- Técnicas de luz estructurada: pertenece a las técnicas de visión activa, ya que modifica las condiciones del entorno. Permite la extracción de información 3D mediante la distorsión producida por la proyección de rayos o planos generados mediante luz coherente o luz láser [7].
- Telemetría láser: en base al tiempo transcurrido entre la emisión y detección de un pulso láser permite determinar un mapa de profundidad del escenario [8].
- Control de parámetros ópticos: a partir del nivel de enfoque en cada píxel de una escena, permite determinar el mapa de profundidad de dicha escena [9].

### 2.1.2 Telémetro láser

Un telémetro láser o *Laser Rangefinder* (LRF por sus siglas en inglés) es un sensor que como se revisó anteriormente permite determinar la distancia hasta un objeto mediante la proyección de un rayo láser y el tiempo de vuelo. Existen 3 tipos de telémetros láser, los cuales solo se diferencian en cuanto a la cantidad de dimensiones que pueden medir.

- Telémetros láser: son aquellos que miden solamente la distancia ortogonal.
- Telémetros láser 2D: mediante la rotación de un espejo el rayo láser puede describir un sector de círculo y medir las distancias contra objetos presentes en ese sector.
- Telémetros láser 3D o combinados: los cuales pueden proporcionar información tridimensional de los objetos presentes dentro de un rango establecido.

Dichos sensores también se pueden encontrar en otra clasificación donde existen 3 tipos:

- Ópticos.

- Ultrasónicos.
- Láser.

Óptico : se conforma de dos objetivos separados por una distancia fija ya conocida, los cuales apuntan a un objeto hasta que la imagen procedente de los objetivos se superpone formando una sola imagen. La distancia al objeto se calcula partiendo de la longitud entre los objetivos y los ángulos entre el eje de los objetivos y la distancia fija conocida. A mayor distancia entre objetivos se obtiene mayor precisión del telémetro.

Ultrasónico : el telémetro emite un ultrasonido, el cual se ve reflejado en el objetivo rebotando al emisor en base al tiempo transcurrido y la fase del eco, se calcula la distancia al objetivo.

Láser : usa un rayo láser para determinar la distancia a un objeto, la forma más común en la que operan es en base al tiempo de vuelo, enviando un pulso láser hacia el objeto, de esta forma midiendo el tiempo que tarda en rebotar el pulso hacia el emisor.

Para el presente trabajo de tesis se empleara el telémetro de la marca SICK modelo LMS200 el cual está disponible en LaViRIA, las ventajas que presenta son:

- Puede ser montado en cualquier posición.
- No requiere de iluminación externa.
- Los datos medidos están disponibles en tiempo real para ser usados posteriormente en cálculos o tareas de control.
- Puede ser empleado en varias aplicaciones por ejemplo: medición de objetos, medición de posición, navegación y monitoreo.

Entre sus características se encuentran las siguientes:

Tabla 2.1: Características de telémetro SICK LSM200

Campo de aplicación	Interior
Versión	Rango corto
Fuente de luz	Infrarrojo (905 nm)
Ángulo de abertura	180°
Resolución angular	0.25°, 0.5°, 1°
Rango de operación	0 m ... 80 m
Máximo rango con 10 % de reflectancia	10 m
Peso	4.5 Kg

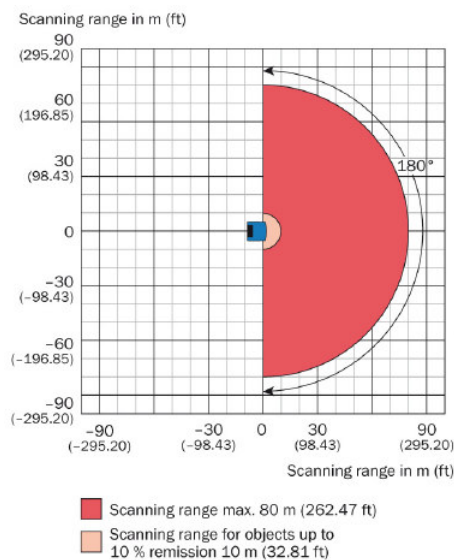


Figura 2.1: Campo de visión del telémetro SICK LMS200.

### 2.1.3 Tipos de movimiento

Los dos tipos de movimiento son el movimiento horizontal y el de inclinación (*Pan and Tilt*), los cuales están relacionados principalmente con las artes visuales, en específico con la cinematografía. El movimiento horizontal se puede describir como el movimiento en un plano horizontal de derecha a izquierda o viceversa. De igual forma la inclinación se puede definir como el movimiento en un plano vertical de arriba a abajo o viceversa.

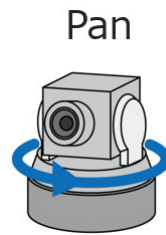


Figura 2.2: Movimiento horizontal



Figura 2.3: Inclinación

#### 2.1.4 Métodos de escaneo

Existen distintos métodos para obtener el escaneo de un entorno, a continuación se presentan cuatro formas que han sido utilizadas como métodos de escaneo. Oliver Wulf y Bernardo Wagner [10] llaman a estos métodos: *pitching scan*, *rolling scan*, *yawing scan* and *yawing scan top*; los cuales se presentan a continuación.

*Pitching scan*: realiza el escaneo mediante un plano horizontal y mediante una inclinación del telémetro de arriba a abajo o viceversa ( tilt ) se obtiene el escaneo. Esta forma fue utilizada por Surmann *et al.* [11] , Hähnel and Burgard [12] y Chou *et al.* [13].

*Rolling scan*: consiste en rotar el telémetro alrededor del centro del scanner con la ventaja de un solo punto de enfoque enfrente del telémetro.

*Yawing scan* y *Yawing scan top*: mediante un plano vertical y realizando la rotación alrededor del eje  $z$ , la diferencia entre estos métodos radica en el espacio requerido para realizar el escaneo. Este método fue utilizado por Riegl [14].



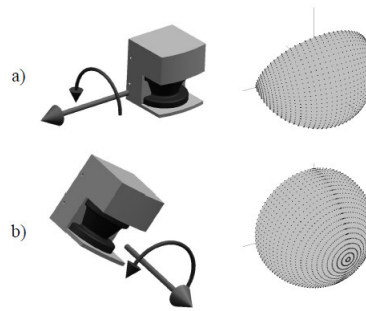


Figura 2.4: (a) Pitching scan, (b) Rolling scan.

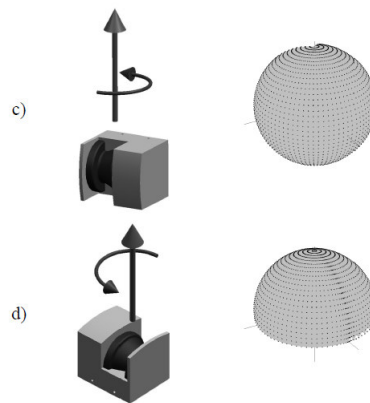


Figura 2.5: (c) Yawing scan, (d) Yawing scan top.

### 2.1.5 Nube de puntos

Muchas aplicaciones en manufactura, medicina, geografía, diseño, etc. requieren de escaneo de objetos tridimensionales complejos. Para (re-) incorporarlos a un procesamiento asistido o basado en computadora. Así, las técnicas de medición evolucionaron para producir de forma más sencilla una gran cantidad de puntos de las superficies de los objetos. Tal conjunto de puntos que representa el límite de un objeto tridimensional es llamado nube de puntos.

En otras palabras cada punto de la nube de puntos posee la información para ubicarlo en un espacio tridimensional  $(x,y,z)$  con un marco de referencia ya establecido. La cantidad de puntos presentes en la nube de puntos determina la densidad de esta y al mismo tiempo la resolución de la reconstrucción 3D.

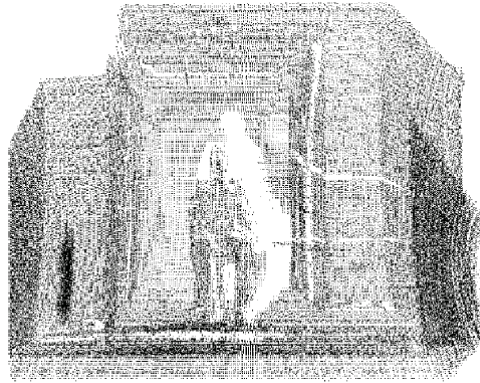


Figura 2.6: Nube de puntos [11]

## 2.2 Estado del Arte

Una nube de puntos tiene diferentes y diversas aplicaciones, y estas se pueden obtener de diferentes formas como se observo en el capítulo anterior donde se presentaban los tipos de escaneo que han sido utilizados. Los métodos de SLAM (*simultaneous localization and mapping*) son las tareas que hace mayor uso de las nubes de puntos, pues uno de los objetivos del SLAM es el mapeo de entornos desconocidos. Establecido esto a continuación se presentan algunos trabajos relacionados al presente trabajo de tesis.

- *An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments.*

En este trabajo se presenta un robot móvil denominado Ariadne, el cual consta de 3 módulos los cuales realizan diferentes tareas; el primero registra el escaneo 3D en un sistema coordinado común y re-localiza al robot. el segundo modulo consiste en un planificador de vista el cual computa la siguiente posición en base a los datos 3D adquiridos mientras esquiva obstáculos. Y el tercer módulo, consta de un controlador para un motor donde el robot móvil navega y en base a la odometría evade colisiones con obstáculos dinámicos [15].

- *Using 3D Laser Range Data for SLAM in Outdoor Environments.*

Se presenta un enfoque de SLAM el cual se basa en exploraciones de alcance nivelado. Dicho método es resultado de una combinación de percepción 3D con localización y mapeo 2D. El sistema procesa la adquisición de los

datos, realiza la segmentación de obstáculos y genera las exploraciones de alcance nivelado y SLAM con dichos escaneos [16].

- *3D Laser Scanning System and 3D Segmentation of Urban Scenes*

En este trabajo se describen los detalles del diseño del sistema de escaneo basado en un telémetro láser 2D, así como la técnica de segmentación para entornos urbanos exteriores para la detección de planos del modelo [17].

- *A robotic indoor 3D Mapping System using a 2D laser range finder mounted on a rotating four-bar linkage of a mobile platform*

Se describe el trabajo que propone un sistema robótico que mapa en 3D compuesto por una plataforma móvil experimental equipado con un telémetro láser [13].

- *The 3D Laser Range Finder Design for the Navigation and Mapping for the Coal Mine Robot*

Se describe la construcción de un telémetro 3D para un robot en una mina de carbón. El prototipo está conformado por un telémetro láser 2D, un servomotor, un sensor de posición, y una tarjeta de adquisición de datos [18].

- *Development of 3D laser range finder system for object recognition*

Se presenta un sistema de un telémetro láser 3D compuesto por un telémetro láser 2D y un módulo de rotación [19].

De los trabajos enlistados anteriormente se puede conocer cuáles son los sensores que se han utilizado para ayudar al telémetro láser a resolver uno o varios problemas. Así como las plataformas o sistemas de movimiento que le dan ese grado de libertad extra al telémetro láser para realizar el escaneo 3D, en dichos trabajos la nube de puntos es el medio para un fin.

## 2.3 Herramientas

Las herramientas utilizadas en este proyecto son paquetes o bibliotecas. Algunos de ellos trabajan de forma conjunta y otros son el medio para la obtención de datos de importancia para el mismo.

### 2.3.1 ANSYS

Software que permite la simulación de problemas de ingeniería, que por medio de análisis de elemento finito permite realizar simulaciones de diversas áreas, por ejemplo estructuras, fluidos, electromagnetismo, semiconductores, sistemas, diseño 3D entre otros. Para el presente proyecto se hizo uso de Workbench; software para análisis estructural.



Figura 2.7: Workbench

### 2.3.2 Solidworks

Software de diseño asistido por computadora (CAD) desarrollado por Dassault Systemes, que provee de una interfaz muy intuitiva para los usuarios para el proceso de diseño de elementos que conformen un sistema. Este software trabaja en base a bocetos que posteriormente se deben transformar para visualizarlos en 3D. También es capaz de incluir diferentes piezas o elementos y unirlos mediante un ensamble, el cual en la mayoría de los casos provee movimiento.



Figura 2.8: SolidWorks

### 2.3.3 Gazebo

El desarrollo de Gazebo comenzó en el otoño de 2002 en la Universidad del Sur de California. Los creadores originales fueron el Dr. Andrew Howard y su estudiante Nate Koenig. El concepto de simulador de alta fidelidad se debe a la necesidad de simular robots en ambientes al aire libre bajo diversas condiciones.

A lo largo de los años, Nate continuó desarrollando Gazebo mientras terminaba su doctorado.

En 2009, John Hsu, Ingeniero Senior de Investigación de Willow, integró ROS y el PR2 en Gazebo, que desde entonces se ha convertido en una de las principales herramientas utilizadas en la comunidad ROS [20]. Dicho simulador es nativo del sistema operativo Linux, entre sus características se encuentran las siguientes:

**Simulación dinámica:** permite el acceso a múltiples motores físicos de alto desempeño entre los cuales se encuentran *ODE*, *Bullet*, *Simbody* y *DART*.

**Gráficos 3D avanzados:** Utilizando *OGRE*, Gazebo proporciona representaciones realistas de los entornos incluyendo iluminación, sombras y texturas.

**Sensores y Ruido:** genera datos de sensores, contando de manera opcional con el ruido, estos sensores van desde telémetros, cámaras 2D y 3D, sensores estilo *Kinect*, sensores de contacto, fuerza-torque y más.

**Plugins:** Desarrollar plugins personalizados para control del ambiente, de sensores y robots.

**Modelos de Robots:** se proporcionan muchos robots incluyendo *PR2*, *Pioneer2 DX*, *iRobot Create* y *Turtle bot*. Se puede incluso construir modelos propios de robots.

**Herramientas para línea de comando:** las extensas herramientas de línea de comandos facilitan la introspección y el control de la simulación.

La versión de Gazebo utilizada para el presente trabajo es Gazebo 7.0, el software se puede encontrar en la siguiente página web <http://gazebo.org/>, así como diversos tutoriales para empezar a trabajar con Gazebo.



Figura 2.9: Gazebo

### 2.3.4 ROS (Robot Operating System)

El sistema operativo robótico ROS es una colección de herramientas, bibliotecas y convenciones que tienen como objetivo simplificar la tarea de crear un comportamiento robótico complejo y robusto. Desde la perspectiva del robot, los problemas que parecen triviales para los seres humanos a menudo varían enormemente entre instancias de tareas y entornos.

Como resultado, ROS fue construido desde cero para fomentar el desarrollo de software de robótica colaborativa. ROS fue diseñado específicamente para grupos de investigación para colaborar y construir sobre el trabajo de cada uno [21].

ROS es un software nativo de Linux, al igual que Gazebo por lo que para el desarrollo de este trabajo de tesis es necesario un equipo de computo que use Linux como sistema operativo o en su defecto una partición del disco duro si no se desea abandonar el sistema operativo habitual.

Cabe mencionar que dependiendo de la versión de Linux que se instale deberá ser compatible con alguna de las versiones de ROS que se ofrecen en su pagina web <http://www.ros.org/> . La versión de Linux utilizada es Ubuntu 16.04 de 64 bits y la versión de ROS es *Kinetic Kame*.



Figura 2.10: ROS Kinetic Kame

### 2.3.5 Matlab

Software matemático desarrollado por *Mathworks*, cuyo nombre proviene de *Matrix Laboratory* (Laboratorio de Matrices) que utiliza un lenguaje propio de alto nivel. Es una de las herramientas muy utilizadas para el procesamiento de datos y la obtención de representaciones gráficas, etc. Se vale de herramientas que solucionan problemas de diversas áreas listando algunas a continuación por ejemplo:

- Visión por computadora

- Procesamiento de señales
- Robótica
- Sistemas de control
- Análisis de datos



Figura 2.11: Matlab

## 2.4 Conclusiones del capítulo

En este capítulo se realizó una revisión de los conceptos, trabajos en la literatura y herramientas. Los cuales fueron utilizados para el desarrollo del presente proyecto. Se habló de los tipos de reconstrucción 3D, destacando el de telemetría láser ya que el sensor a utilizar es un telémetro láser 2D. Se dio una introducción a las herramientas que se utilizaron para lograr la reconstrucción 3D de escenarios como una nube de puntos. También se revisaron los métodos de escaneo que se encuentran implementados en diversos trabajos en la literatura del campo de trabajo. En el siguiente capítulo se plantea el proceso de diseño y la metodología seguida para resolver el diseño mecánico del sistema.

Los telémetros láser han sido utilizados para resolver diversas problemáticas y se ha recurrido en mayor medida al uso de telémetros láser 2D, ya que el costo de estos es menor a un telémetro 3D. Debido a ello en el estado del arte se han implementado robots móviles o sistemas que se valen de un telémetro láser en conjunto con algún sistema motriz, cámara, etc. para poder realizar un escaneo en tres dimensiones. De esta forma se obtiene la representación de un escenario, el cual puede ser utilizado para resolver tareas específicas. Tales como mapear un entorno desconocido, detección de formas, detección de planos, etc. En la siguiente sección se mencionarán algunos trabajos relacionados.

---

## DISEÑO MECÁNICO

---

En este capítulo se presentará la metodología para realizar el proceso de diseño, así como el desarrollo de la misma, la cual fue empleada para obtener los requerimientos del presente sistema en cuanto a la parte mecánica para su futura implementación. La función de realizar un diseño de cualquier índole surge de formular un plan para satisfacer una necesidad específica o resolver un problema. Si el plan resulta en la creación de algo físicamente real, entonces este producto debe ser funcional, seguro, útil, confiable y que pueda fabricarse.

### **3.1 Proceso de diseño**

La función de realizar un diseño de cualquier índole surge de formular un plan para satisfacer una necesidad específica o resolver un problema. Si el plan resulta en la creación de algo físicamente real, entonces este producto debe ser funcional, seguro, útil, confiable y que pueda fabricarse [22]. A continuación se presenta un bosquejo del proceso necesario para realizar un diseño.



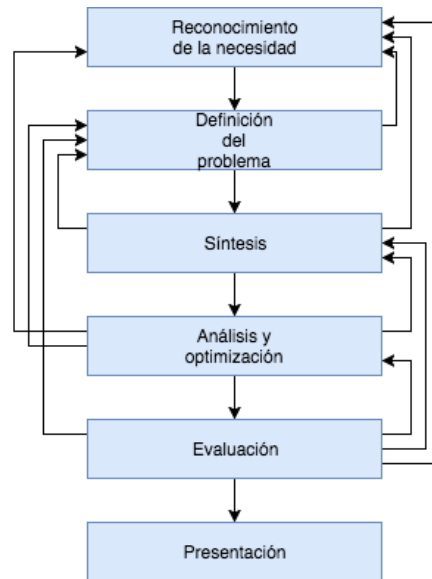


Figura 3.1: Proceso de diseño [22].

- Reconocimiento de la necesidad: Se establecen las necesidades o problemas a solucionar, se necesita de una abstracción del sistema para identificar los elementos que van a interactuar para el desempeño de la tarea a resolver.
- Definición del problema: Incluye las especificaciones del objeto a diseñarse. Estas pueden ser cantidades de entrada y salida, dimensiones, limitaciones, material.
- Síntesis: Se proponen esquemas y se cuantifican en medidas establecidas, donde ya se incluyen las especificaciones para cada esquema propuesto.
- Análisis y optimización: Se someten los esquemas para evaluar su desempeño, si este resulta cuando menos satisfactorio se prosigue con el proceso, aquí es donde algunos esquemas no sobreviven y se desechan.
- Evaluación: Se realiza una comparación cuando mas de algún esquema propuesto pasa por el análisis y es aquí donde algunas especificaciones como el costo, confiabilidad, entre otras definen que esquema es el indicado.

## 3.2 Reconocimiento de necesidades y definición de problema

La necesidad que se busca resolver para el presente diseño es: mover al telémetro de tal forma que se pueda obtener la información suficiente para reconstruir el escenario.

Las especificaciones establecidas son:

- Entradas: movimiento provisto por actuadores, lectura 2D provista por el telémetro.
- Salidas: información 3D para construir nube de puntos.
- Materiales: aleaciones de aluminio.
- Limitaciones: movimiento controlado.

### 3.3 Síntesis

Actualmente el diseño asistido por computadora (CAD *Computer-Aided Design*) permite obtener una visualización 3D de los esquemas ideados. Para tal tarea existen diversos software de CAD como AutoCAD, CadKey, ProEngineer y SolidWorks, sólo por mencionar algunos.

En particular para el proyecto propuesto se eligió usar SolidWorks debido a que es una herramienta que posee una cierta facilidad de uso así como cierto grado de intuición para crear partes o ensamblajes. Cabe mencionar que en lo personal se tiene más experiencia en dicho software que fue obtenida a lo largo de la licenciatura. Los ensamblajes y piezas se deberán guardar con extensión .STEP y .STL para poder importarlos a Workbench y Gazebo respectivamente.

Para obtener una pieza en SolidWorks, primero se debe tener una idea clara de lo que se planea dibujar, una vez realizado esto se debe establecer un plano donde se iniciara la elaboración de *Sketch* (boceto) alternando con operaciones de extracción o corte según se necesite para elaborar la pieza deseada.

Ya que se tienen las piezas necesarias ahora se necesita crear un ensamblaje, el cual nos permitirá unir y restringir el movimiento de las piezas que lo conforman. Para los esquemas que se desarrollaron se hizo uso de modelos 3D provistos por los fabricantes, y se realizaron los modelos y ensamblajes de las piezas faltantes para completar dichos esquemas.

A continuación se describen y presentan los esquemas que se idearon para satisfacer la necesidad planteada .

- Manivela-biela-corredera: Mediante un actuador lineal, en conjunto con elementos de transmisión de movimiento, puede dar el movimiento de rotación al telémetro.
- Rotación pura: Un soporte fijo que sostiene al telémetro y mediante un actuador (servomotor, motor a pasos, moto reductor) y que logra así la rotación del telémetro.

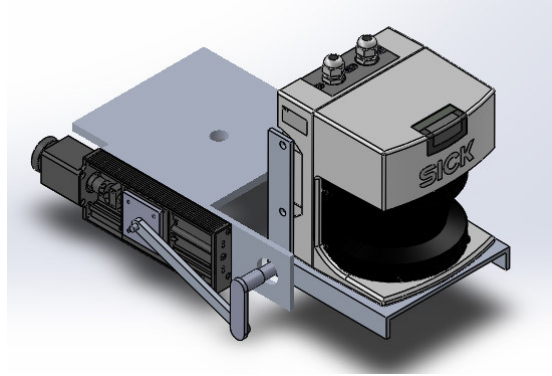


Figura 3.2: Manivela-biela-corredera.

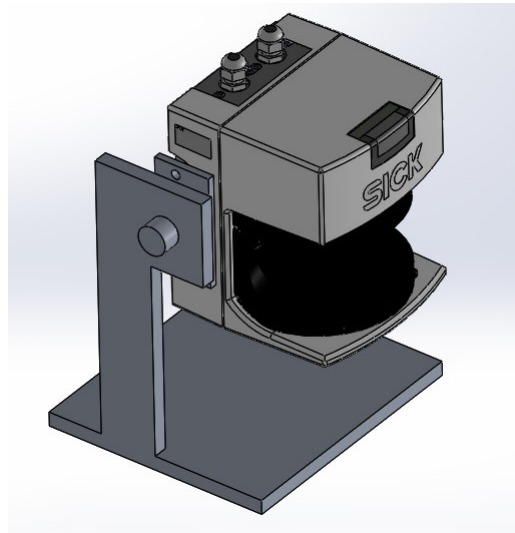


Figura 3.3: Rotación pura.

## 3.4 Análisis

Así como CAD también existe la Ingeniería Asistida por Computadora o CAE (*Computer-Aided Engineering*) que permite realizar análisis específicos o tareas de simulación que son de gran ayuda al diseñador, algunos de los software mencionados tienen ambas funciones (CAD y CAE). EL CAE basado en ingeniería utiliza el análisis del elemento finito para realizar la simulación, el cual es un método numérico para la aproximación de soluciones mediante ecuaciones diferenciales parciales.

Para realizar el análisis de los esquemas propuestos se utilizó ANSYS; que permite al igual que SolidWorks, crear partes y realizar el análisis. En mi apre-

ciación personal, es más entendible el entorno de ANSYS en comparación al de SolidWorks. Dentro de las plataformas de ANSYS se encuentra Workbench la cual fue la herramienta que permitió obtener los resultados del análisis de ambos esquemas.

Para realizar un análisis en Workbench existe una metodología que se debe seguir para que nuestro análisis sea exitoso, en breve se explicara en que consiste dicho procedimiento.

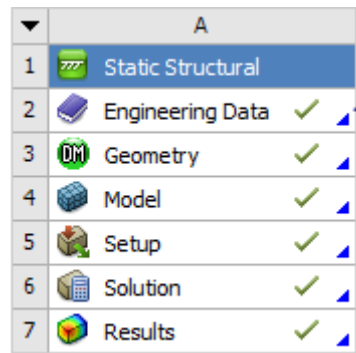


Figura 3.4: Esquema para un análisis.

- Definir el análisis a realizar: Workbench dentro de sus herramientas ofrece análisis de diversos campos( estructurales, fluidos, magnéticos, dinámicos, etc.). Para el diseño que se busca realizar se eligió análisis estructural estático.
- Selección y definición de material: Existen diversas categorías de materiales, éstas darán a nuestro modelo las propiedades necesarias para simular su comportamiento. Dichas propiedades poseen valores que se pueden modificar para simular un material específico. Para el presente diseño se selecciono aleación de aluminio 6061 T6. Debido a que es una aleación comercial y cumplió con las necesidades planteadas.
- Creación de geometría: ANSYS posee ambas funciones (CAD y CAE) así podemos crear nuestra pieza y proseguir con el análisis, ya que resulta complicado realizar una pieza en el entorno de Workbench se acudió a SolidWorks, se debe importar el ensamble y crear una operación para reducir el número de caras del modelo.
- Definición de Modelo: Teniendo ya nuestro modelo importado se deberán añadir y especificar las condiciones de simulación (Mallado, Conexiones entre cuerpos, fuerzas, soportes, magnitudes a medir).
- Visualización e interpretación de resultados: Al ejecutar la simulación, se desplegaran los resultados obtenidos de forma cuantitativa y gráfica.

A continuación se muestran los resultados obtenidos de ambos esquemas. Para el esquema de manivela-biela-corredera existen limitaciones proporcionadas por el actuador lineal. Como consecuencia las magnitudes medidas en nuestro análisis para esta propuesta deben estar por debajo de las especificadas por el fabricante. Se realizó el análisis variando la velocidad de desplazamiento del actuador (máx 20 mm/s).

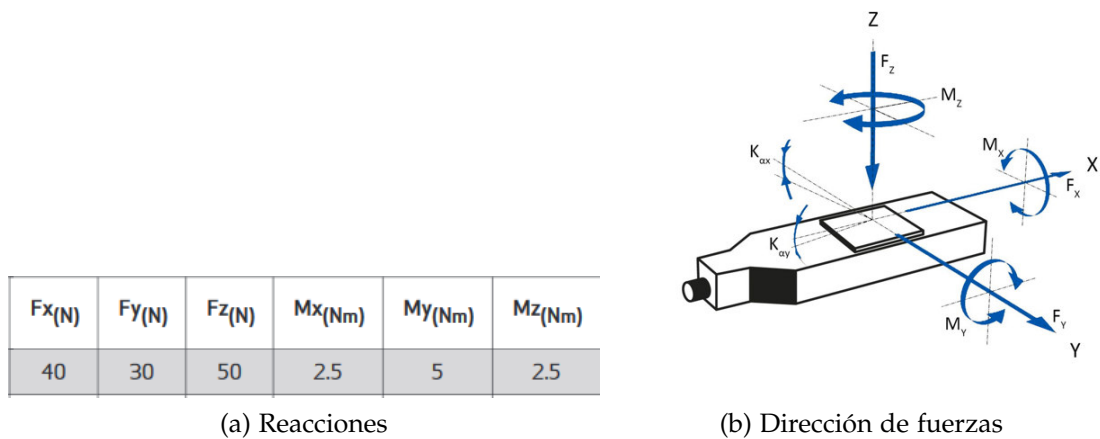


Figura 3.5: Especificaciones de actuador lineal.

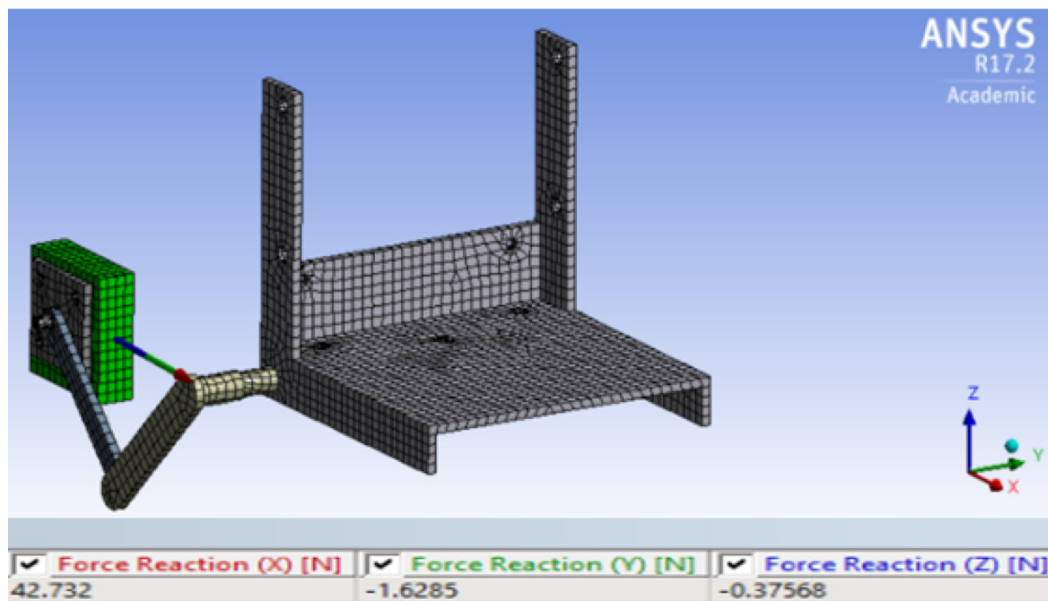


Figura 3.6: Actuador a 1 mm/s.

Ya que los resultados obtenidos en este primer análisis sobrepasan los límites de las reacciones en el actuador en la dirección X de la imagen, aunque la ve-

locidad de desplazamiento es muy baja lo cual repercute en el tiempo total de operación. Se procedió a aumentar la velocidad para observar su comportamiento. Para el segundo se estableció una velocidad de 20 mm/s.

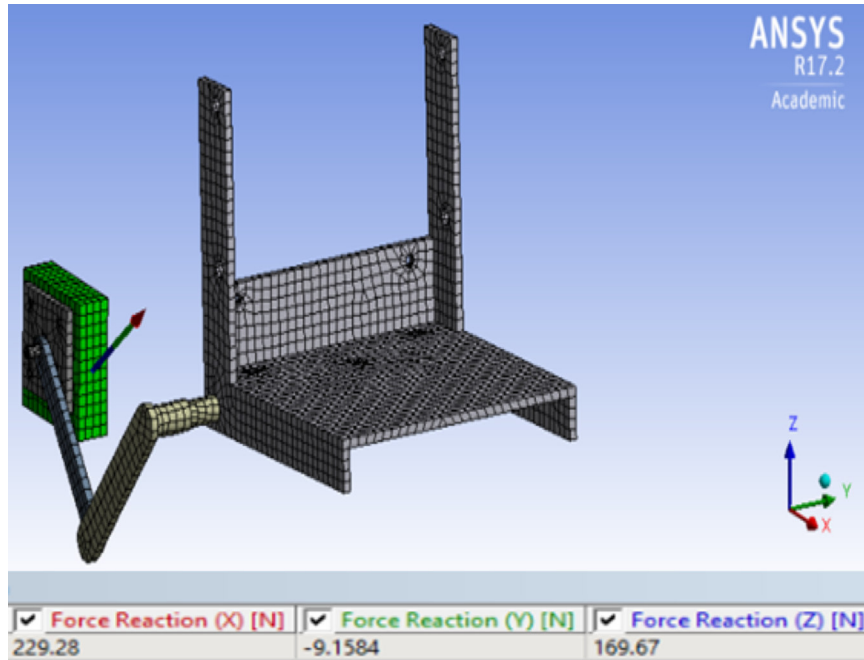


Figura 3.7: Actuador a 20 mm/s.

Los resultados obtenidos en las reacciones ahora sobrepasan los límites estipulados en 2 direcciones y se observó que estas reacciones son directamente proporcionales a la velocidad de desplazamiento.

Para el esquema de rotación pura se realizó el análisis en busca de conocer el torque o momento necesario para rotar el telémetro mediante un servomotor o motor a pasos. A continuación se muestran los resultados obtenidos del torque así como de las deformaciones en el soporte en dos posiciones; la primera cuando el telémetro se encuentra a  $0^\circ$  con respecto al plano ZY en la imagen y la segunda cuando el telémetro se encuentra a  $-60^\circ$  con respecto al mismo plano.

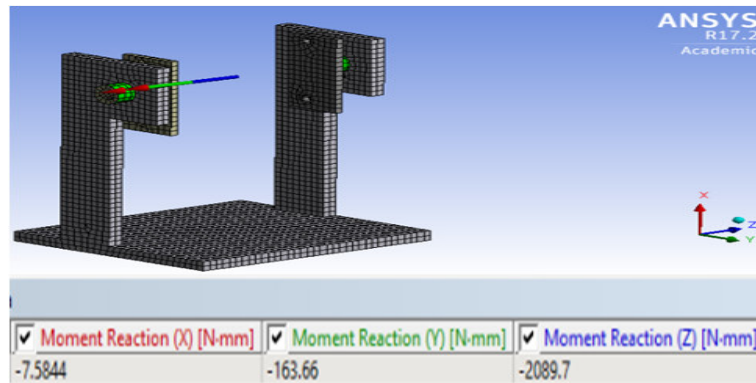


Figura 3.8: Torque a  $0^\circ$  respecto al plano ZY.

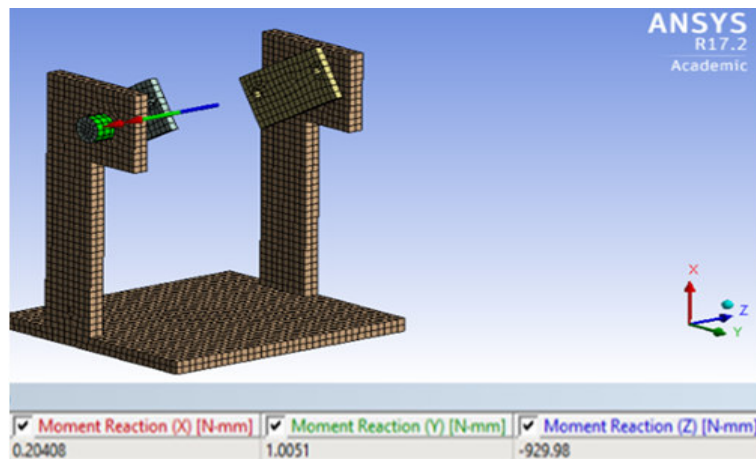


Figura 3.9: Torque a  $-60^\circ$  respecto al plano ZY.

El torque se calcula de siguiente forma:

$$T = W * d \quad (3.1)$$

Donde:

- $T$  es el torque o momento (Nm)
- $W$  es el peso del cuerpo (N).
- $d$  es la distancia perpendicular a la dirección del peso (m).

Para calcular el peso del telémetro se tiene la siguiente fórmula:

$$W = m * g \quad (3.2)$$

Donde:

- $W$  es el peso del cuerpo (N).
- $m$  es la masa del objeto (Kg)
- $g$  es la aceleración de la gravedad ( $m/s^2$ ).

El peso mantiene su dirección en el eje X de las figuras mostradas y a causa de que se requiere medir el torque en la dirección del eje Z la distancia que debemos medir es en la dirección del eje Y, la cual es aproximadamente 4.7 cm medida obtenida en Solid Works y la masa del telémetro es 4.5 Kg, ahora se sustituye en la ecuación del torque y se obtiene:

$$T = 4,5(9,81)(0,0472) = 2,074Nm \quad (3.3)$$

El resultado obtenido se aproxima al del análisis ya que falta considerar el efecto ejercido por las piezas que soportan al telémetro en el soporte.

### 3.5 Conclusiones del capítulo

En este capítulo se presentó el proceso de diseño que fue necesario para resolver el diseño mecánico que requiere el sistema para la lograr la reconstrucción 3D de escenarios. El sistema propuesto se elaboró de forma modular, en el que se destacan dos módulos esenciales. El módulo de construcción 3D de los componentes del sistema y el módulo de simulación estructural, ambos módulos se implementaron para los dos esquemas propuestos. El análisis se realizó para obtener en el caso del esquema manivela-biela-corredera las reacciones que el actuador lineal sufría al realizar su funcionamiento, ya que los resultados no resultaron satisfactorio se desechó ese diseño por causa de la baja velocidad a la cual se podía operar el actuador lineal sin dañarlo.

Del análisis estructural para el primer esquema se obtuvo que las reacciones en el actuador lineal están por debajo de las condiciones de funcionamiento previstas por el fabricante. Para el segundo esquema se obtuvo el momento o par necesario para rotar el telémetro en sentido antihorario, con el fin de vencer la fuerza de gravedad para poder lograr el movimiento tilt del telémetro.



# 4

---

## ENTORNOS DE SIMULACIÓN

---

En este capítulo se presentan el entorno de simulación realizado en Gazebo en conjunto con ROS. Este capítulo está dividido en cuatro secciones. En la primera sección se habla acerca de Gazebo. En la segunda sección se habla de ROS y de su forma de operar. En la tercera sección se describe la estructura y los elementos que contiene el proyecto. En la cuarta sección se encuentra la descripción de los escenarios simulados.

Un simulador bien diseñado hace posible probar rápidamente algoritmos, diseñar robots, realizar pruebas de regresión y entrenar el sistema IA utilizando escenarios realistas. Gazebo ofrece la capacidad de simular de forma precisa y eficiente poblaciones de robots en complejos ambientes interiores y exteriores. A su alcance está un robusto motor de física, gráficos de alta calidad y cómodas interfaces gráficas y programables. Lo mejor de todo, Gazebo es gratis con una comunidad muy activa. [20]

### 4.1 Gazebo

En primera instancia debemos tener ya el modelo 3D del robot previamente obtenido de algún software de CAD, pero con una pequeña modificación; el formato que necesitamos para poder introducirlo en el ambiente de trabajo de Gazebo debe ser el formato collada o por su extensión .dae . Si el software de donde se obtuvo el modelo 3D no puede exportar a collada lo que se debe de hacer es exportar a *Standard Triangle Language* o por su extensión .stl y este a su vez importarlo para obtener el formato .dae. Blender es un software libre para la creación de modelos 3D el cual puede exportar al formato que se necesita.

Ahora la simulación en Gazebo se puede realizar de dos formas: la forma gráfica y la forma en código. La forma gráfica consiste en utilizar el entorno

de Gazebo para construir el robot o se puede importar como se explicó anteriormente. También se puede orientar, posicionar, establecer las uniones de las diversas partes que conforman al robot y especificar el ambiente bajo el cual se va a realizar la simulación.

Esta forma de simulación utiliza el formato SDF, el cual es el predeterminado por Gazebo para guardar nuestra simulación. Los sensores son modelos que también se pueden agregar y establecer a las partes del robot mediante uniones. Para utilizar esta forma es necesario tener un esquema de la posición y orientación de cada eslabón del robot.

La forma en código consiste en utilizar el formato URDF para describir los elementos que conforman al robot, las uniones que presentan los elementos, los sensores ha utilizar, etc. Todo esto escrito desde un archivo xml y guardarlo con la extensión URDF. El formato XACRO es una simplificación del URDF lo cual nos permite una descripción más simple de nuestro robot.

La gran diferencia entre estas dos formas radica en que Gazebo no puede leer por si mismo el formato URDF y necesita de ROS para poder hacer la transformación a SDF y de esta forma poder visualizarlo. Sin embargo, el formato SDF genera un archivo xml similar al del formato URDF pero con modificaciones en la sintaxis y el uso de algunos comandos, la mayor desventaja del SDF radica en que ROS se basa en el formato URDF para la ejecución de algunas tareas.

## 4.2 ROS

La arquitectura de ROS ha sido diseñada y dividida en tres grupos o niveles:

- Nivel archivos de sistema
- Nivel gráfico de cómputo
- Nivel comunidad

### 4.2.1 Nivel archivos de sistema

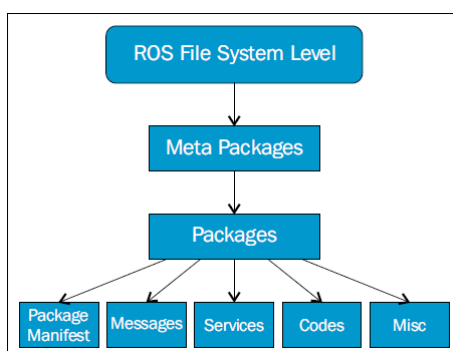


Figura 4.1: Archivos de sistema ROS. [23]

De forma similar a un sistema operativo, los archivos de ROS están organizados dentro del disco duro de cada maquina donde se encuentra instalado. En la siguiente imagen podemos ver como esta dada dicha organización:

- Meta paquetes: es la agrupación de paquetes para un propósito especial.
- Paquetes: es la unidad más básica de ROS, los cuales contienen nodos, librerías, archivos de configuración, etc.
- Servicios: es una especie de interacción solicitud/respuesta entre procesos.
- Mensajes: son un tipo de información que es enviada por un proceso a otro.
- Códigos: son los procesos presentes en el paquete.

### 4.2.2 Nivel gráfico de cómputo

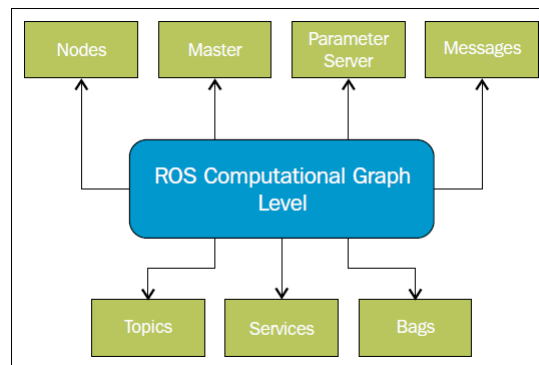


Figura 4.2: Gráfico de cómputo. [23]

- **Nodos:** son los procesos que realizan el cómputo.
- **Maestro:** proporciona registro de nombres y busca a los nodos ya que estos no pueden encontrarse mutuamente.
- **Tópicos:** son los buses mediante los cuales son transportados cada uno de los mensajes en ROS.
- **Mensajes:** son la estructura de datos mediante la cual los nodos se pueden comunicar.
- **Servicios:** son la forma de comunicación en ambos sentidos entre nodos. Los servicios están conformados por dos nodos el que solicita y el que responde.
- **Bolsas:** son un formato para guardar y reproducir datos de los mensajes.

### 4.2.3 Nivel comunidad

Son los recursos que permiten el intercambio de software y conocimiento dentro de la comunidad de ROS, entre los cuales se encuentran: *blogs*, *ROS Answers*, repositorios, etc.

## 4.3 Descripción del Proyecto

Para poder realizar la simulación y así obtener las nubes de puntos. Se requiere de la creación de un *Workspace* (Espacio de trabajo), el cual se crea desde

la terminal, mediante comandos que se encuentran en tutoriales de ROS [21]. Ya creado el espacio de trabajo se procede a la creación de un *Package* ( Paquete ), el cual contendrá los nodos, modelos, archivos URDF o XACRO, *launch* ( lanzadores ), etc.

Una vez concluida la creación del Espacio de trabajo y paquete(s), se debe comenzar por visualizar de forma adecuada nuestro robot o sistema. Esto se logra gracias a los archivos URDF o XACRO, en los cuales como se explicó en la sección anterior se especifican las características necesarias para poder visualizar los modelos. A continuación se muestra una imagen del contenido de un archivo XACRO.

Para poder utilizar los sensores dentro de Gazebo se necesita especificar en el archivo XACRO el *plugin* que nos permite hacer uso de ellos, en nuestro caso ocupamos el plugin para el telémetro láser. Se requiere de nodos o servicios que realicen los procesos o acciones que necesitamos para realizar la simulación y obtener información de ella. ROS nos permite escribir dichos nodos o servicios en lenguaje C++ o Python. Los nodos que se ocuparon para la presente simulación fueron dos; uno que nos permite realizar la función de un servomotor y el otro nos permite obtener la información que obtiene el telémetro láser del escenario.

También se utilizó un servicio el cual nos proporcionaba información de la posición y orientación del telémetro. Tanto la información del servicio como la del nodo del telémetro se escriben en un archivos de texto que posteriormente de manera conjunta nos permitirán obtener las nubes de puntos.

## 4.4 Descripción de Escenarios

Existen dos tipos de escenarios para los cuales se realizó la simulación, el objetivo del primer escenario será cumplir con la prueba conceptual del sistema, el segundo escenario será sometido a una prueba de desempeño de la cual se obtendrá la exactitud en base a la detección de un plano.

El primer escenario está conformado por 6 objetos, 3 de ellos conforman el sistema que realiza el escaneo del escenario y los 3 restantes son objetos situados en diferentes posiciones frente al sistema. Los objetos que conforman el sistema son: una caja, la cual le proporciona un soporte al sistema, el soporte propio del sistema, el cual es el que sostiene al telémetro y al servomotor mientras se realiza el escaneo y por último el telémetro con sus respectivas extensiones que le permiten la rotación proporcionada por el servomotor.

Los objetos escaneados son: una caja, una esfera y una figura tipo lego; a la derecha, frente y a la izquierda del sistema respectivamente. Todas las ubicaciones de los objetos escaneados están referidas al sistema de referencia del simulador.

Las dimensiones de la caja son (2, 2, 2) m y su centro se encuentra ubicado en (4, -4, 2) m, el diámetro de la esfera es 2 m y el centro se encuentra ubicado a (6, 0, 1) m, la figura lego está ubicada a (7.5, 3, 0) m. A continuación se muestra una imágenes tanto del sistema como del escenario.

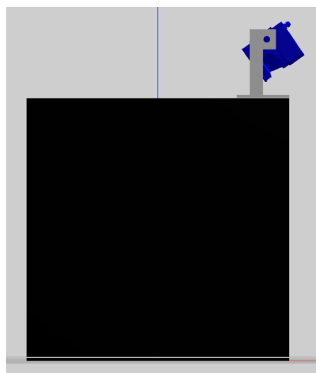


Figura 4.3: Sistema

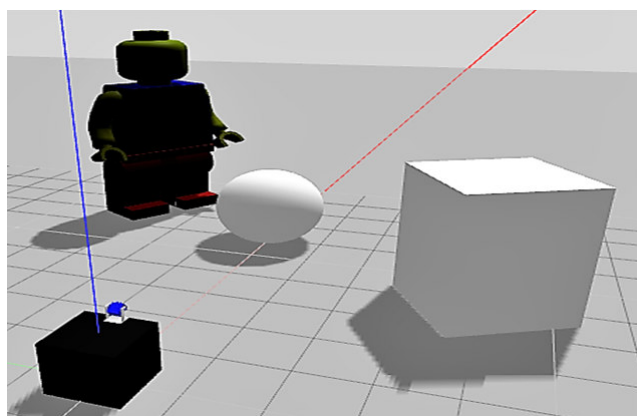


Figura 4.4: Primer escenario

El segundo escenario está formado por 4 objetos, al igual que el primer escenario los 3 que componen al sistema y el objeto de prueba el cual es una caja de 2 m x 2 m x 2 m y su centro se encuentra en  $(x, 0, 1.25)$  donde  $x$  varía desde 2 m a 11 m. A continuación se muestra la imagen del escenario para la prueba de desempeño.



Figura 4.5: Segundo escenario vista plano xz

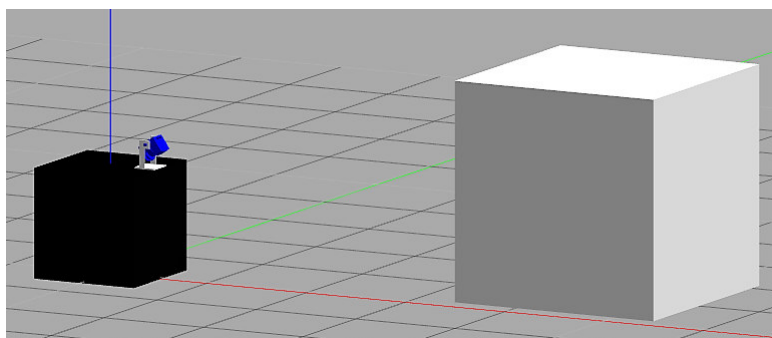


Figura 4.6: Segundo escenario perspectiva

## 4.5 Conclusiones del capítulo

En este capítulo se revisaron las herramientas para poder realizar la simulación de nuestro sistema, se clarifican las diferencias entre los formatos URDF y SDF, de tal forma que se puede decir que ROS prefiere el uso del formato URDF para ejecutar algunas instrucciones al momento de llevar a cabo la simulación. En cuanto a ROS se explicó la forma en que trabaja y se establecieron los conceptos de los diferentes elementos que interactúan entre sí para el cumplimiento de un proceso. También se describe el escenario de prueba y la estructura del proyecto. En el siguiente capítulo se presentaran los resultados obtenidos de dichas simulaciones.

---

## RESULTADOS

---

En este capítulo se presentan los resultados obtenidos de las simulaciones realizada en Gazebo en conjunto con ROS. Este capítulo está dividido en cuatro secciones. Para la primera sección se especifican las características de ambas simulaciones. En la segunda sección se describen los resultados del primer escenario. En la tercera sección se describen los resultados del segundo escenario. En la cuarta sección se muestran las conclusiones del capítulo.

### 5.1 Características de Simulación

Una vez concluida la simulación y ya con la información obtenida de ella en los archivos de texto, se procede a realizar las operaciones necesarias para poder dibujar la nube de puntos referida a un sistema de referencia especificado en este caso el sistema de referencia quedo ubicado en  $(0, 0, 0)$ . Dicho sistema de referencia se puede modificar en base a las necesidades que se presenten.

Las especificaciones de la simulación son las siguientes:

- El láser realiza un barrido desde  $0^\circ$  a  $180^\circ$  en pasos de un grado.
- El movimiento *tilt* se logra debido al nodo que realiza la función del servomotor que mueve al telémetro desde  $-55^\circ$  a  $60^\circ$  cada grado en referencia al plano XZ del simulador.

Dichas especificaciones se pueden modificar en el nodo llamado *move* y en el archivo XACRO que contiene la descripción del robot. Como se especificó en el segundo capítulo, el telémetro láser tiene tres resoluciones para el barrido  $0.25^\circ$ ,



0.5°, 1°. haciendo un total de 724, 362 y 181 lecturas respectivamente.

La nube de puntos se obtuvo gracias a Matlab el cual es un software con un lenguaje de alto nivel el cual es un muy utilizado en el área de ingeniería para poder resolver problemas de dicha área de una forma más fácil de comprender para los usuarios. Mediante un código escrito en dicho software se leen los archivos de texto con la información de la simulación y se realizan las operaciones que dan origen a encontrar las coordenadas X,Y,Z y así poder visualizar la nube de puntos.

La información que se obtuvo de la simulación esta conformada por:

- Posición Z de la unión entre el soporte y el telémetro.
- Orientación Y de la unión entre el soporte y el telémetro.
- Lectura obtenida del telémetro.

Como se mencionó anteriormente la nube de puntos debe estar referida a un sistema de referencia para poder llevar la información a dicho sistema se necesitan realizar operaciones con la información para poder visualizar la nube de puntos de forma correcta. Las operaciones utilizadas se muestran a continuación.

$$x = (\lambda \sin(\phi) + d) \cos(\theta) + px \quad (5.1)$$

$$y = \lambda \cos(\phi) \quad (5.2)$$

$$z = pz - (\lambda \sin(\phi) + d) \sin(\theta) \quad (5.3)$$

Donde:

- $x$  es la coordenada en el eje X.
- $y$  es la coordenada en el eje Y.
- $z$  es la coordenada en el eje Z.
- $\lambda$  es la lectura del telémetro láser.
- $\phi$  es el ángulo en el cual se realiza la lectura sobre el barrido.
- $\theta$  es el ángulo que describe la orientación del telémetro.
- $d$  es igual a 0.1512 m y establece la distancia entre el centro de rotación y el centro del láser.

- $p_z$  es igual a 1.2246 m y establece la posición del centro de rotación del telémetro en el eje Z.
- $p_x$  es igual a 0.4142 m y establece la posición del centro de rotación del telémetro en el eje X.

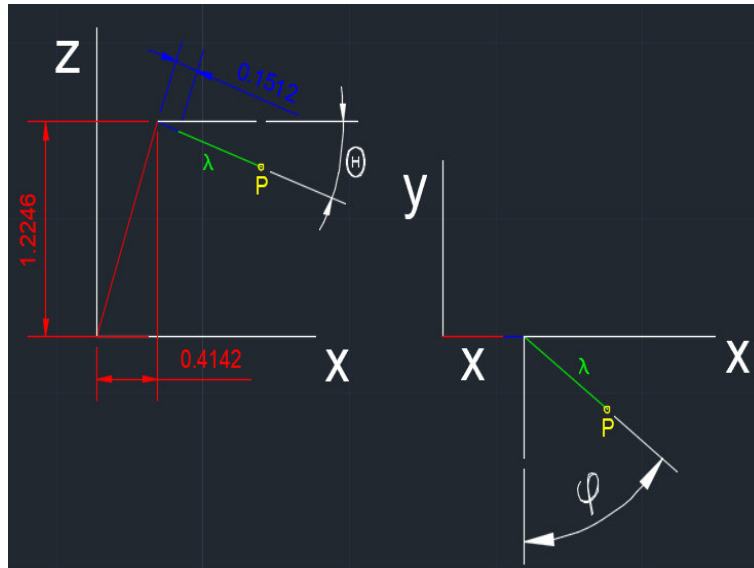


Figura 5.1: Diagrama de vectores para la obtención de las coordenadas de un punto en el espacio

## 5.2 Prueba Conceptual: Primer escenario

La simulación se realizó un total de 12 veces en cada uno de las simulaciones se distingue por la orientación de la figura tipo lego esto con el fin de verificar que la construcción de una nube de puntos funciona para cuerpos simples (caja y esfera) e irregulares como la figura tipo lego. Ya que el fin de esta prueba solo es probar el funcionamiento del sistema no se requiere de algún trabajo extra sobre los datos presentes en la nube de puntos. A continuación se muestran imágenes de las pruebas conceptuales.

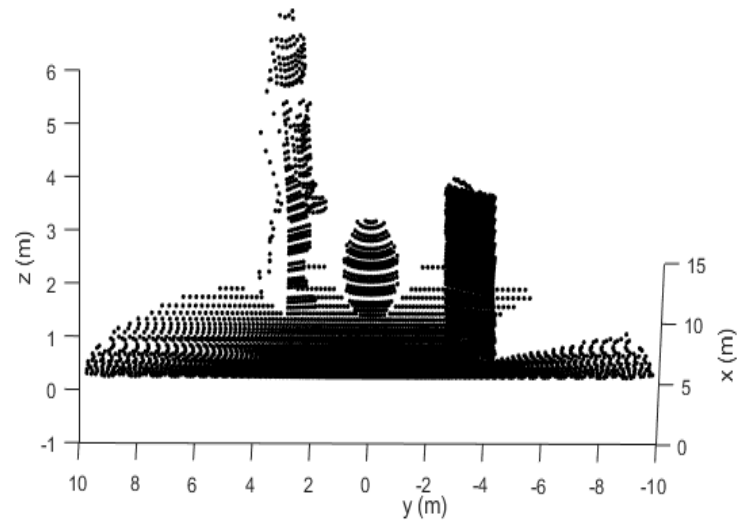


Figura 5.2: Prueba conceptual Lego a  $30^\circ$  respecto al eje  $y$  negativo.

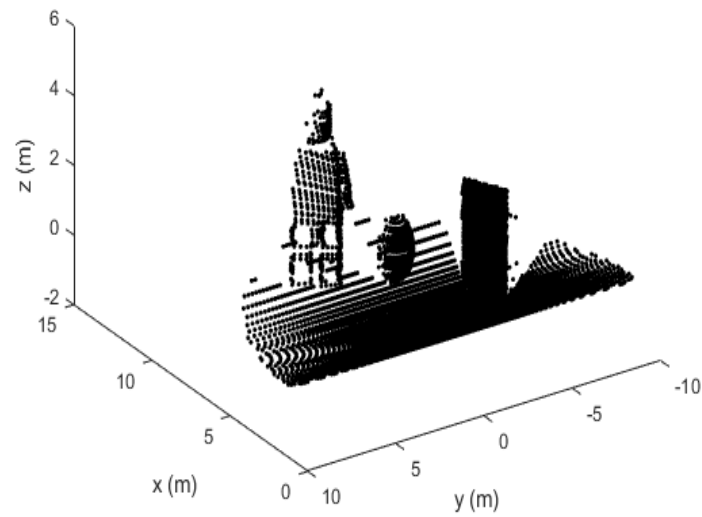


Figura 5.3: Prueba conceptual Lego a  $60^\circ$  respecto al eje  $y$  negativo.

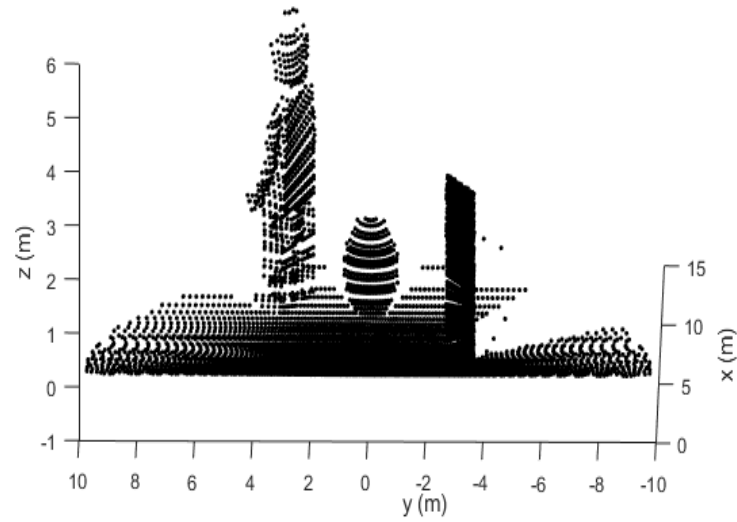


Figura 5.4: Prueba conceptual Lego a  $150^{\circ}$  respecto al eje y negativo.

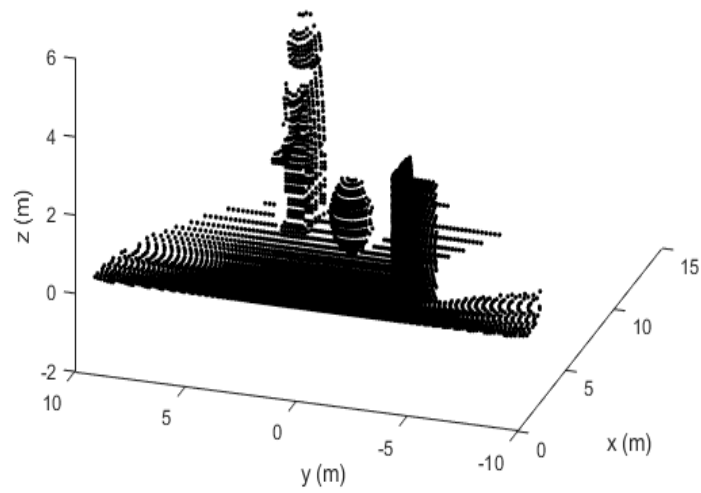


Figura 5.5: Prueba conceptual Lego a  $180^{\circ}$  respecto al eje y negativo.

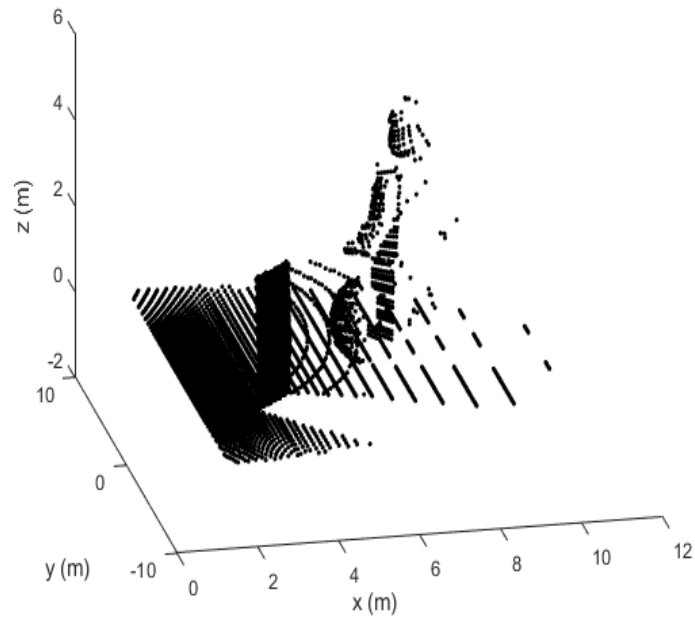


Figura 5.6: Prueba conceptual Lego a  $210^{\circ}$  respecto al eje y negativo.

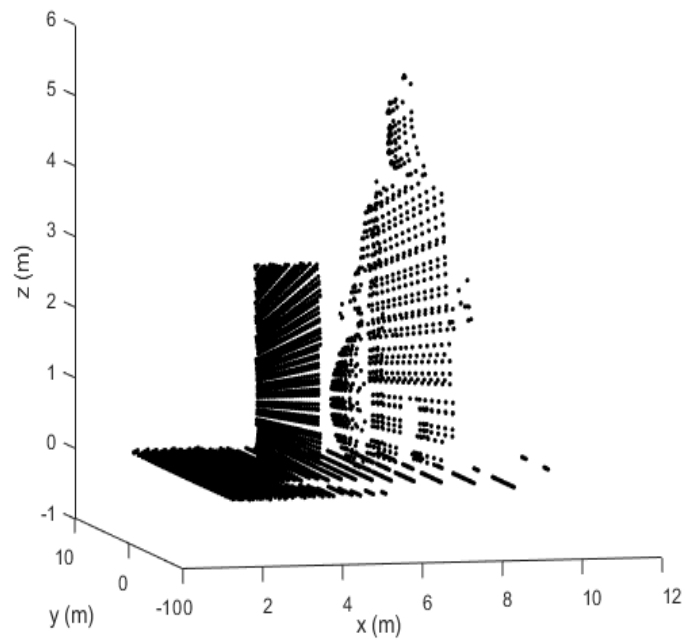


Figura 5.7: Prueba conceptual Lego a  $330^{\circ}$  respecto al eje y negativo.

## 5.3 Prueba de Desempeño : Segundo escenario

La prueba de desempeño debe estar orientada a la medición de algún parámetro que describa su funcionamiento. Esta prueba está dedicada a medir la exactitud de una nube de puntos, entendiendo por exactitud como la confiabilidad de un dato medido. Como el objetivo de esta prueba es medir esa exactitud se tiene que recurrir a la medición del error la cual nos dirá que tan cerca se esta de un dato real. Sin embargo se debe de tener la información necesaria para poder medir la exactitud, se requiere de los valores medidos y una verdad de referencia para poder realizar la comparación de los datos medidos contra los datos reales. La verdad de referencia sera la coordenada  $x$  donde se encuentra la cara que colisiona con las mediciones del telémetro.

$$dcx = cx - (dx/2) \quad (5.4)$$

Donde:

- $dcx$  es la verdad de referencia
- $cx$  es la coordenada  $x$  del centro de la caja.
- $dx$  es la dimensión  $x$  de la caja.

Recordando del capítulo anterior para el segundo escenario el objeto de prueba se encuentra "flotando", de esta forma los puntos que representan a la caja se encuentran separados del piso. Por medio de una regresión múltiple aplicada a dichos puntos se obtiene una ecuación que describe al plano que conforman los puntos que representan a la caja.

$$Ax + By + Cz + D = 0 \quad (5.5)$$

Donde:

- $x$  es la coordenada en el eje X.
- $y$  es la coordenada en el eje Y.
- $z$  es la coordenada en el eje Z.
- $A, B, C, D$  son coeficientes obtenidos de la regresión.

Después para obtener los valores medidos se evalúa la ecuación obtenida para encontrar  $x$  en base a los puntos que representan a la caja. Estos valores de  $x$  serán los que se utilizarán como valores medidos para la medición del error. Específicamente para esta prueba se utilizó el error cuadrático medio, el cual

es una herramienta de uso muy frecuente. El error cuadrático medio se obtiene mediante la siguiente fórmula:

$$e = 1/n * \sqrt{\sum_{i=1}^n (X_i - X)^2} \quad (5.6)$$

Donde:

- $e$  es el error.
- $X_i$  es el  $i$ -ésimo valor del vector de valores medidos.
- $X$  es el valor real.
- $n$  es el total de valores del vector .

Ya que para cada simulación los valores de los ángulos en el movimiento "tilt" del telémetro no son los mismos, esto se refleja en el error obtenido anteriormente. Por ello, es necesario saber como está distribuido dicho error. Una de las herramientas más utilizadas para resolver este problema es por medio de una distribución normal, para este caso una distribución normal del error cuadrático medio. Esta tiene como objetivo expresar la probabilidad de un valor de una variable aleatoria continua.

Para obtener la distribución normal del error es necesario evaluar la media y la desviación estándar de los datos que describen el error, una vez obtenidos se gráfica la función de distribución de probabilidad.

A continuación se muestran las ecuaciones que describen a la media y a la desviación estándar

$$\mu = 1/n * \sum_{i=1}^n x_i \quad (5.7)$$

$$\sigma = 1/n * \sqrt{\sum_{i=1}^n (x_i - \mu)^2} \quad (5.8)$$

Donde:

- $\mu$  es la media.
- $\sigma$  es la desviación estándar
- $x_i$  es el  $i$ -ésimo valor de un vector de datos.
- $n$  es el total de valores en el vector .

La simulación se realizó un total de 10 veces, en cada una de las simulaciones se distingue por la posición de la caja a lo largo del eje X esto con fin de observar el comportamiento del error en la exactitud conforme el objeto de prueba se va ubicando a diferente profundidad. A continuación se muestran de forma gráfica los resultados obtenidos de 5 escenarios, donde los puntos negros representan la nube de puntos; los azules son aquéllos que pertenecen a la caja; en negro se encuentra la verdad de referencia; en rojo el plano estimado mediante la regresión múltiple y en verde el plano promedio para cada simulación .

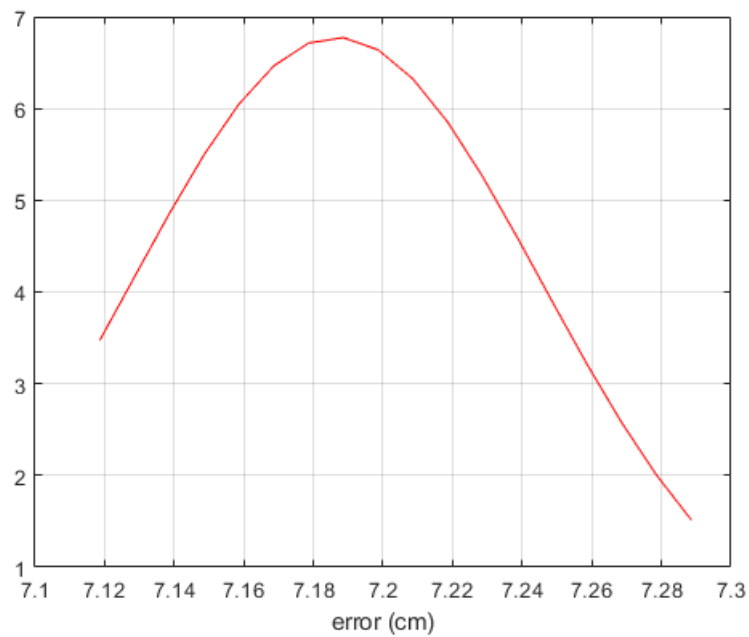


Figura 5.8: Dist. Normal  $\mu=7.18$  cm,  $\sigma=0.058$  cm a 1 m.



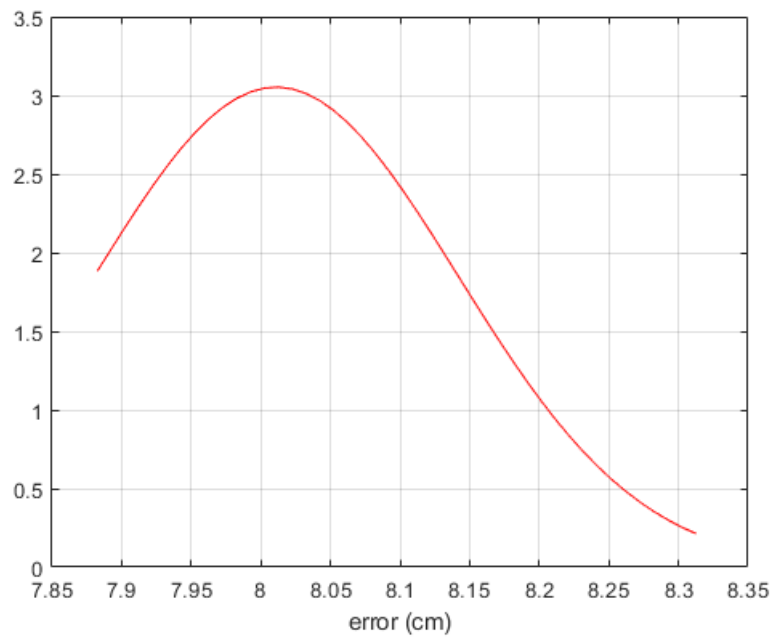


Figura 5.9: Dist. Normal  $\mu=7.77$  cm,  $\sigma=0.094$  cm a 3 m.

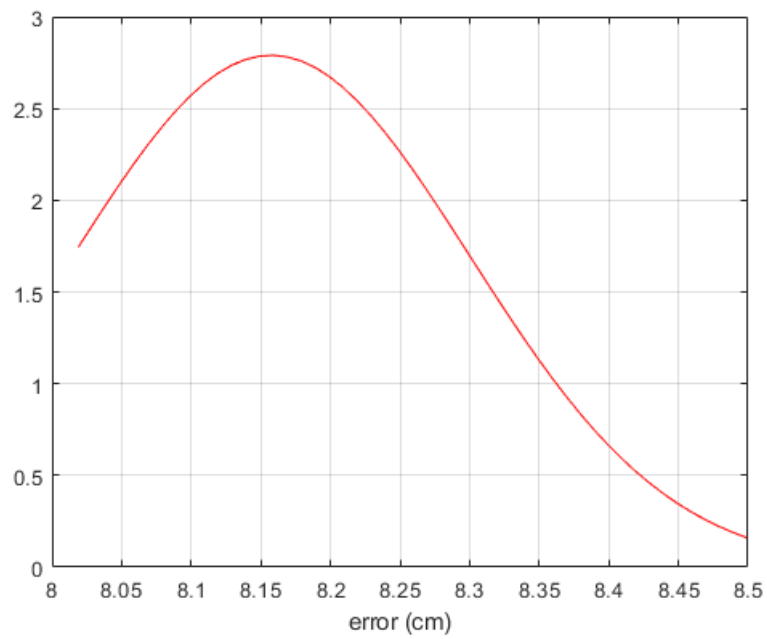


Figura 5.10: Dist. Normal  $\mu=8.15$  cm,  $\sigma=0.14$  cm a 5 m.

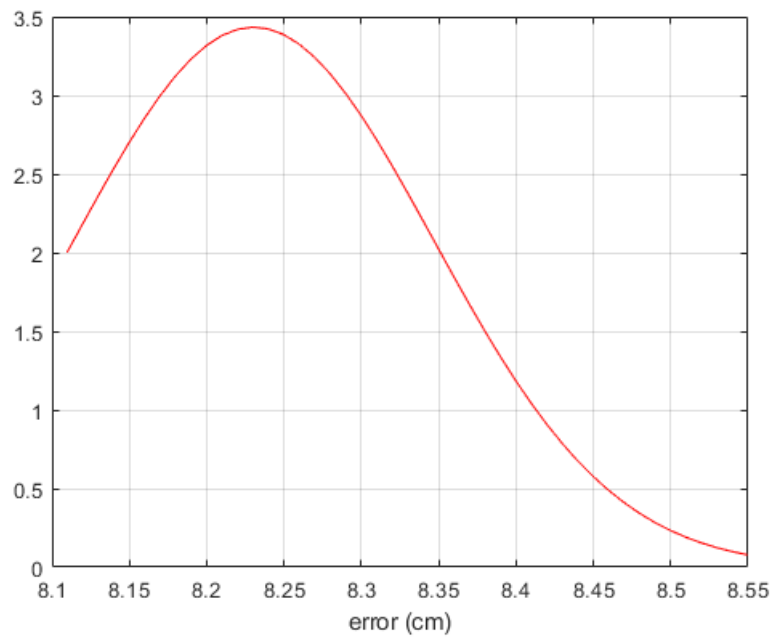


Figura 5.11: Dist. Normal  $\mu=8.23$  cm,  $\sigma=0.11$  cm a 7 m.

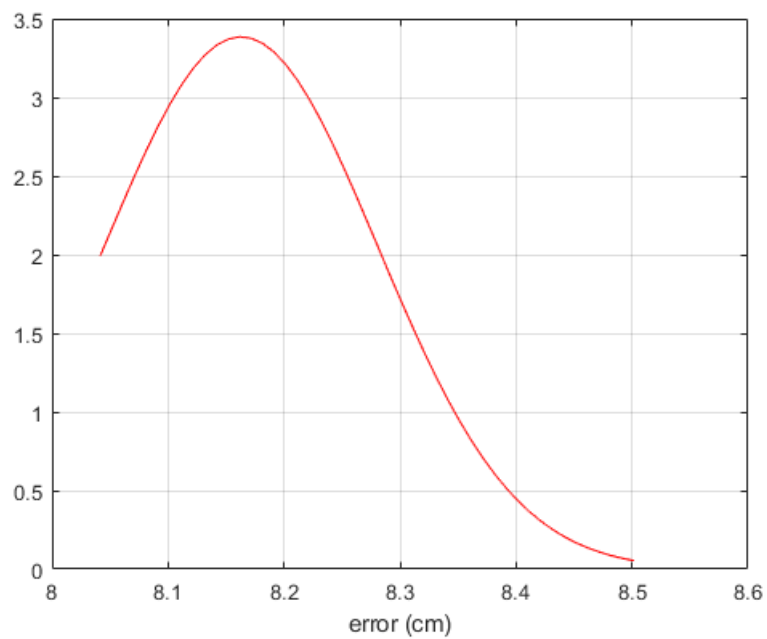


Figura 5.12: Dist. Normal  $\mu=8.16$  cm ,  $\sigma=0.11$  cm a 9 m.

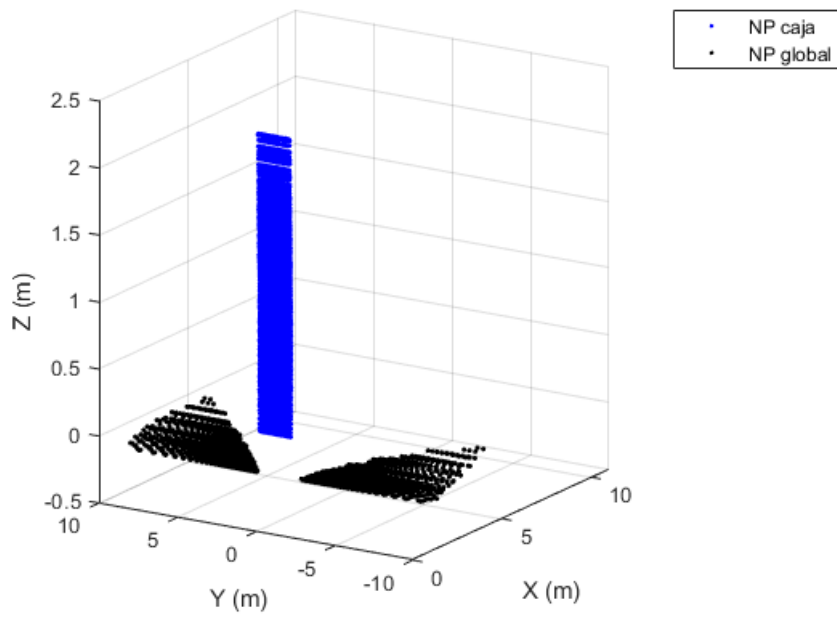


Figura 5.13: Nube de Puntos a 1 m ejemplo 1

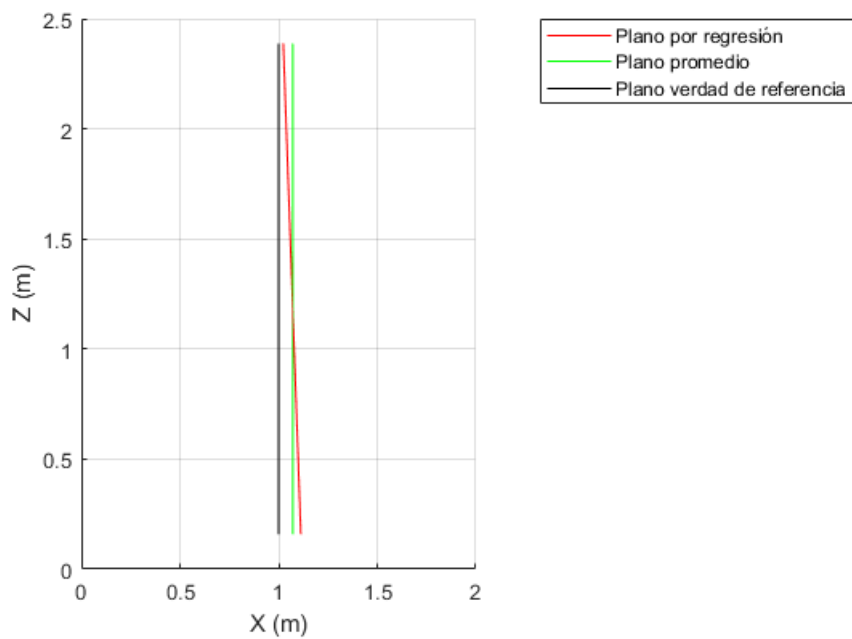


Figura 5.14: Comparación de planos a 1 m ejemplo 1

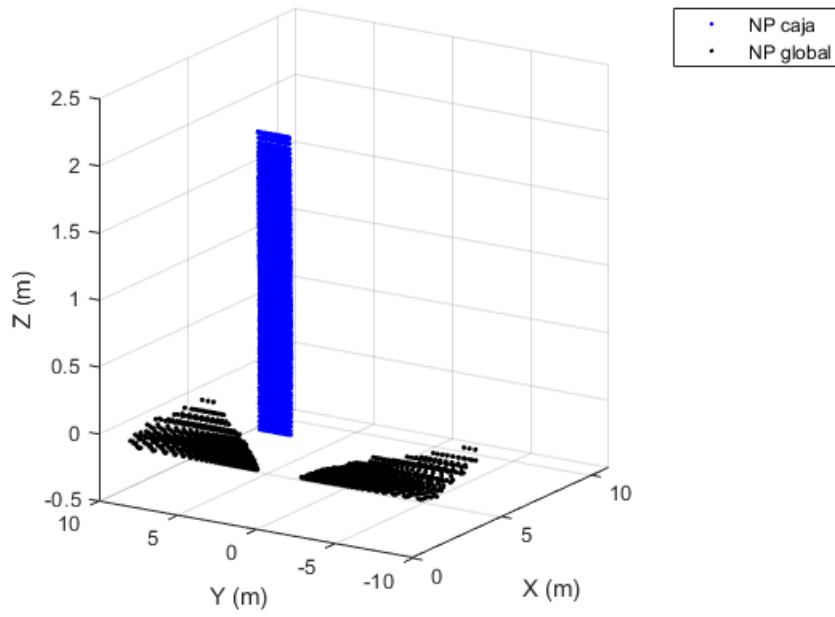


Figura 5.15: Nube de Puntos a 1 m ejemplo 2

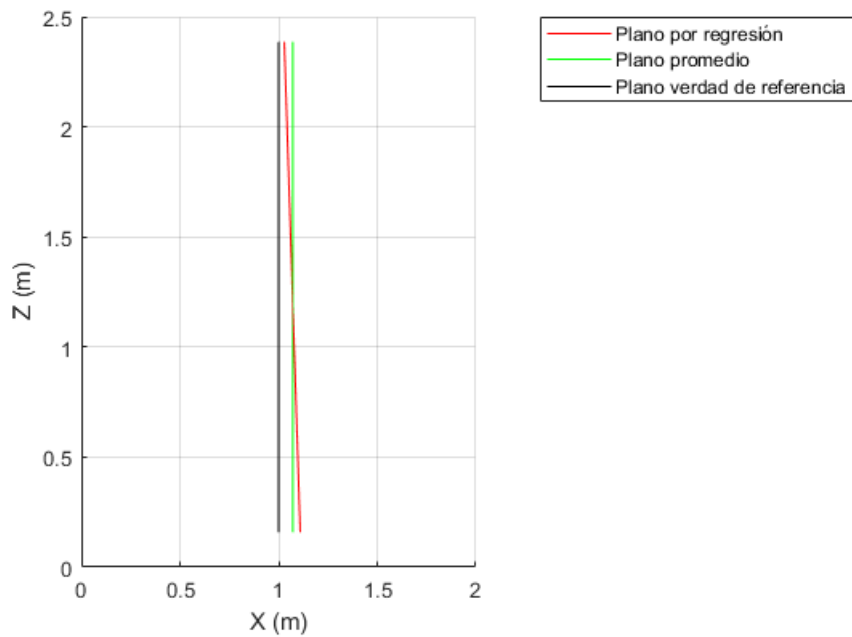


Figura 5.16: Comparación de planos a 1 m ejemplo 2

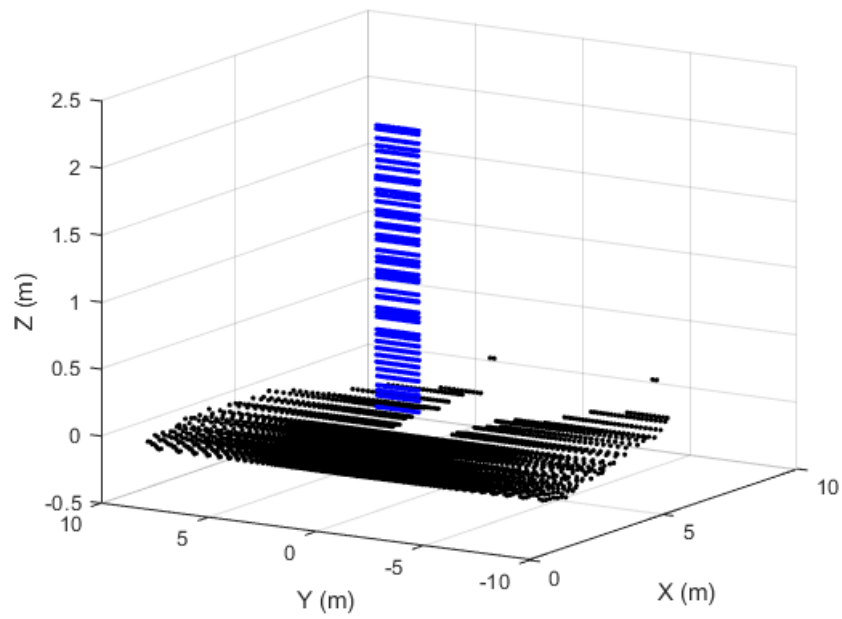


Figura 5.17: Nube de Puntos a 3 m ejemplo 1

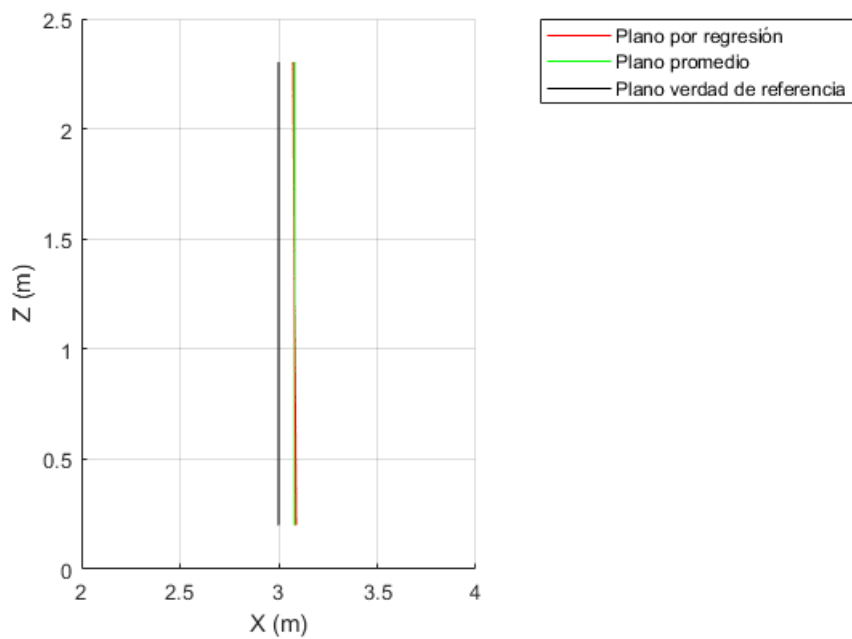


Figura 5.18: Comparación de planos a 3 m ejemplo 1

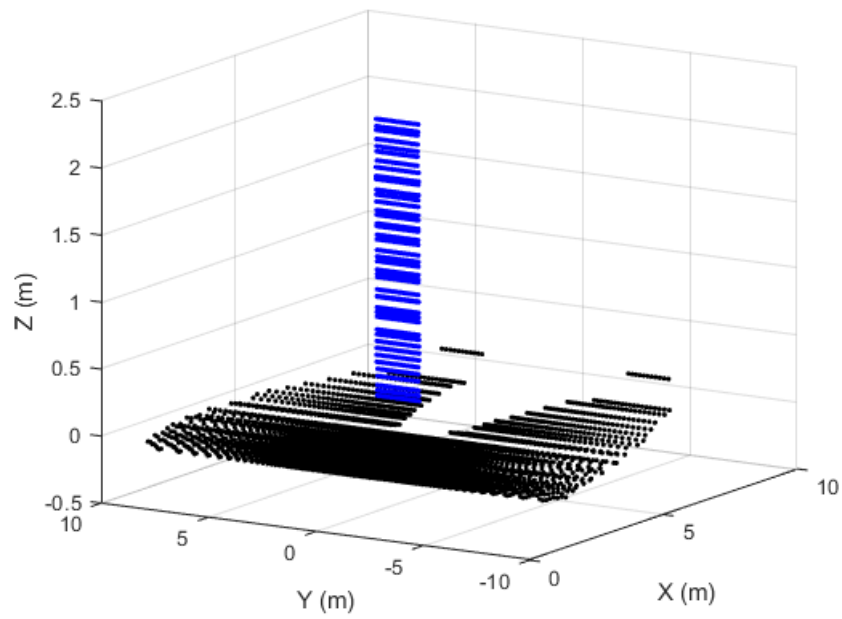


Figura 5.19: Nube de Puntos a 3 m ejemplo 2

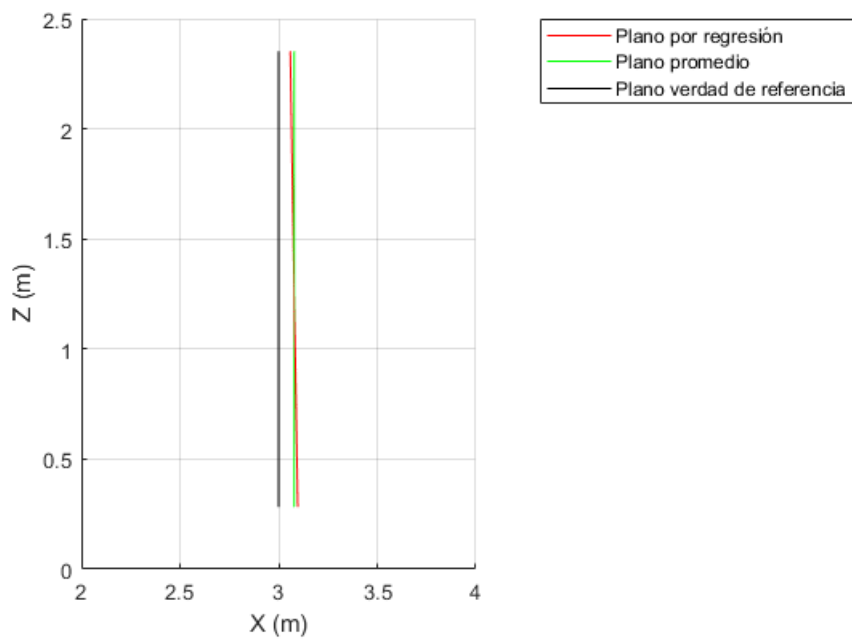


Figura 5.20: Comparación de planos a 3 m ejemplo 2

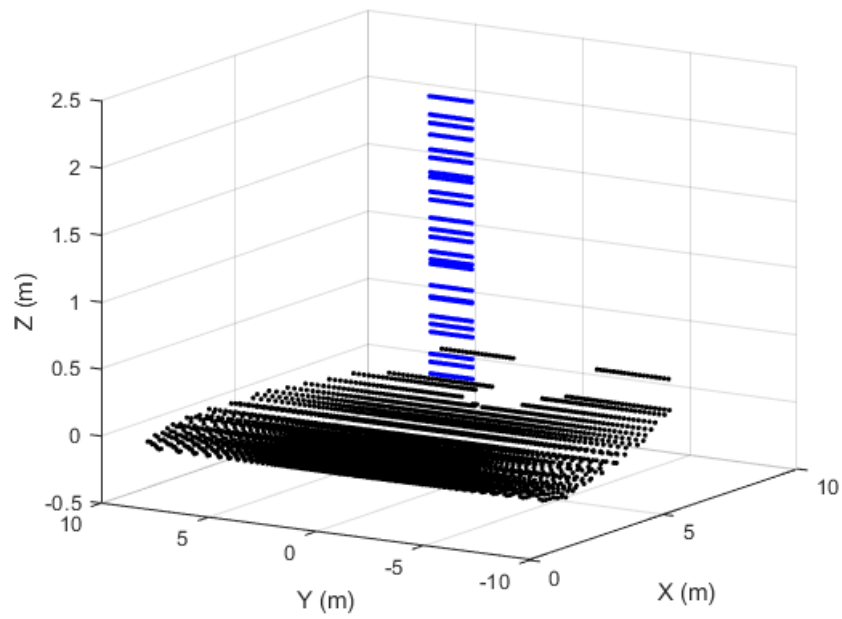


Figura 5.21: Nube de Puntos a 5 m ejemplo 1

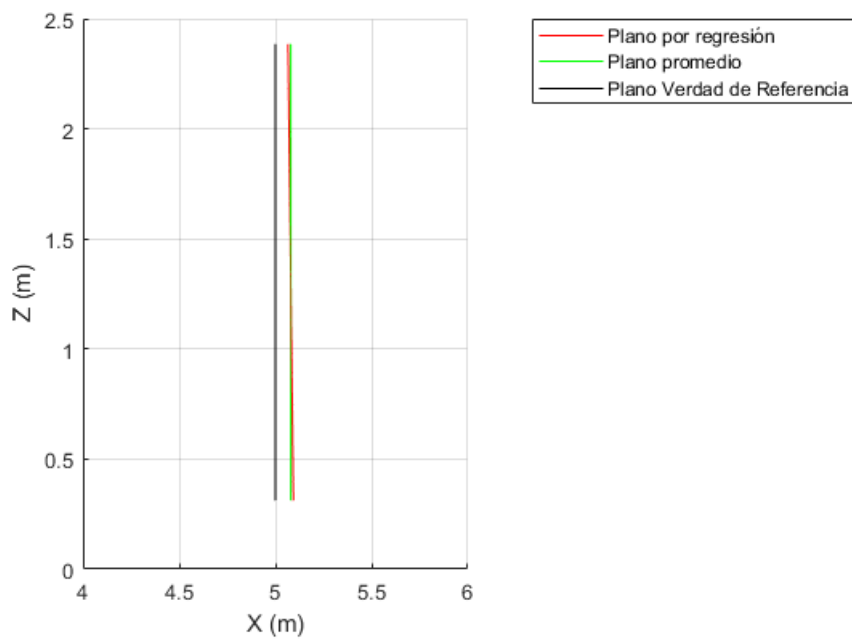


Figura 5.22: Comparación de planos a 5 m ejemplo 1

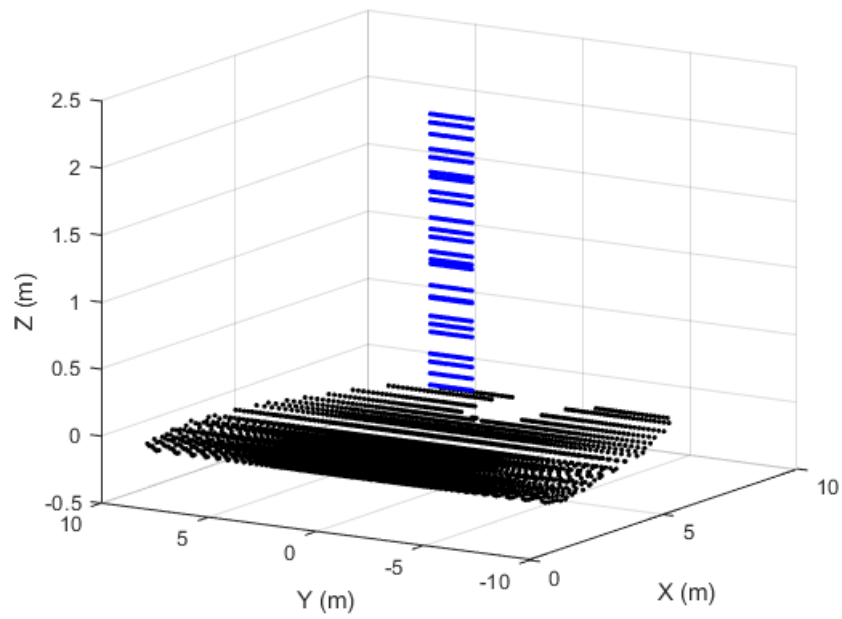


Figura 5.23: Nube de Puntos a 5 m ejemplo 2

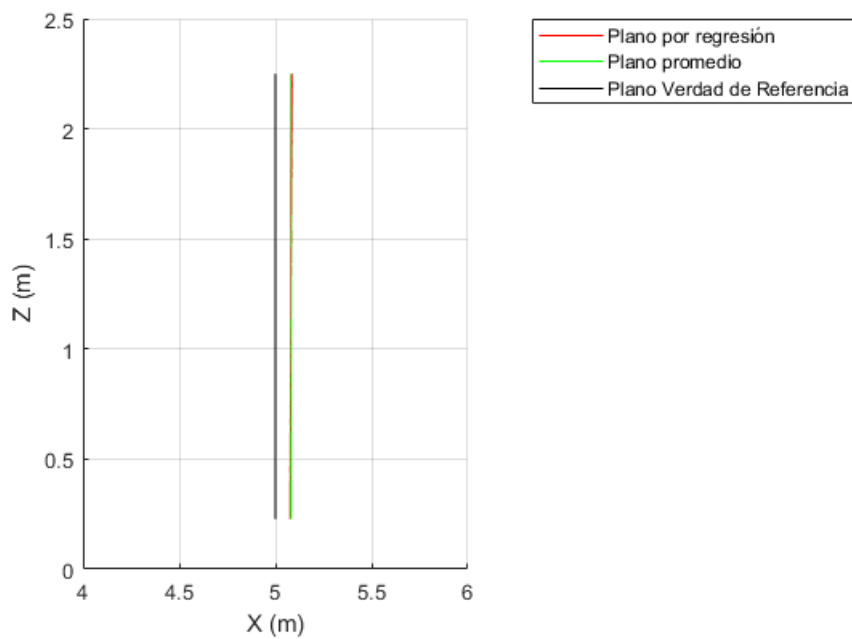


Figura 5.24: Comparación de planos a 5 m ejemplo 2



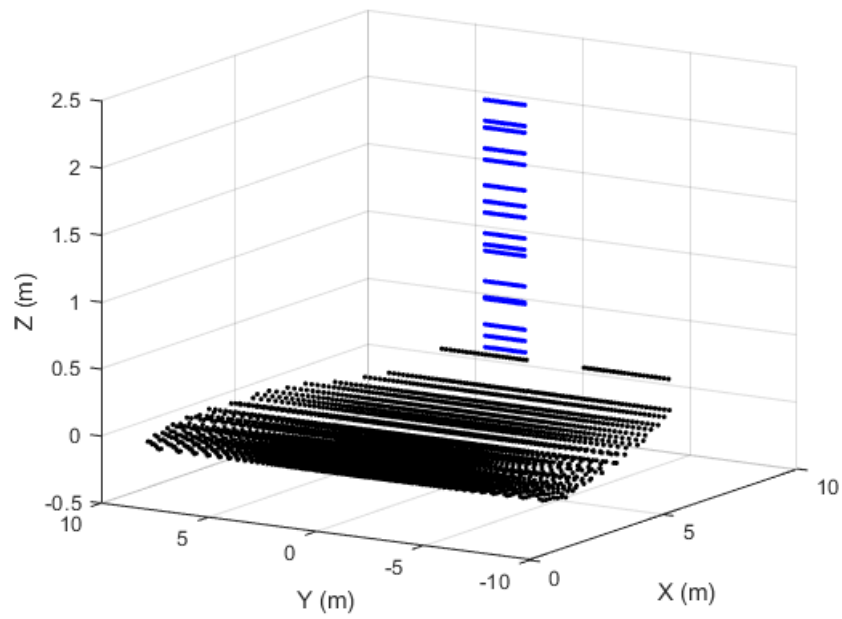


Figura 5.25: Nube de Puntos a 7 m ejemplo 1

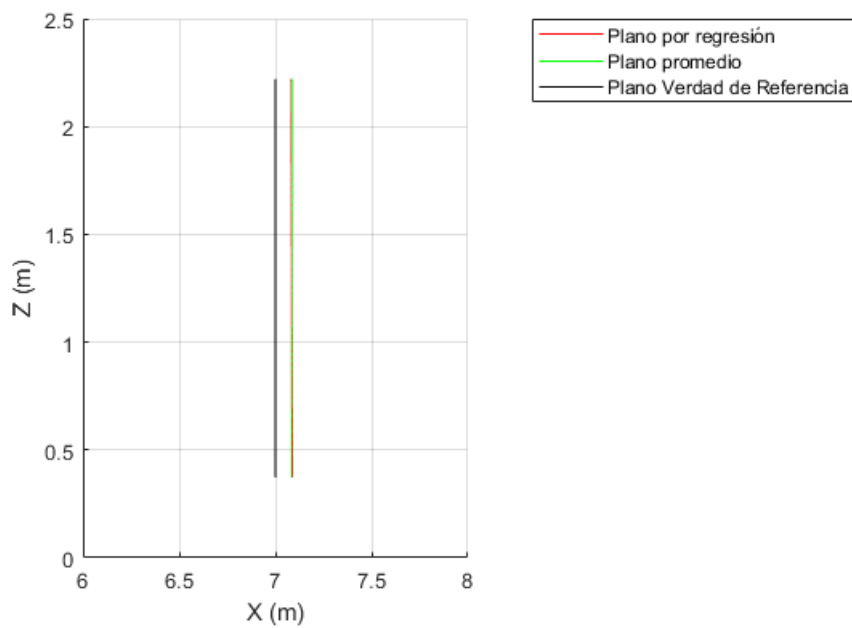


Figura 5.26: Comparación de planos a 7 m ejemplo 1

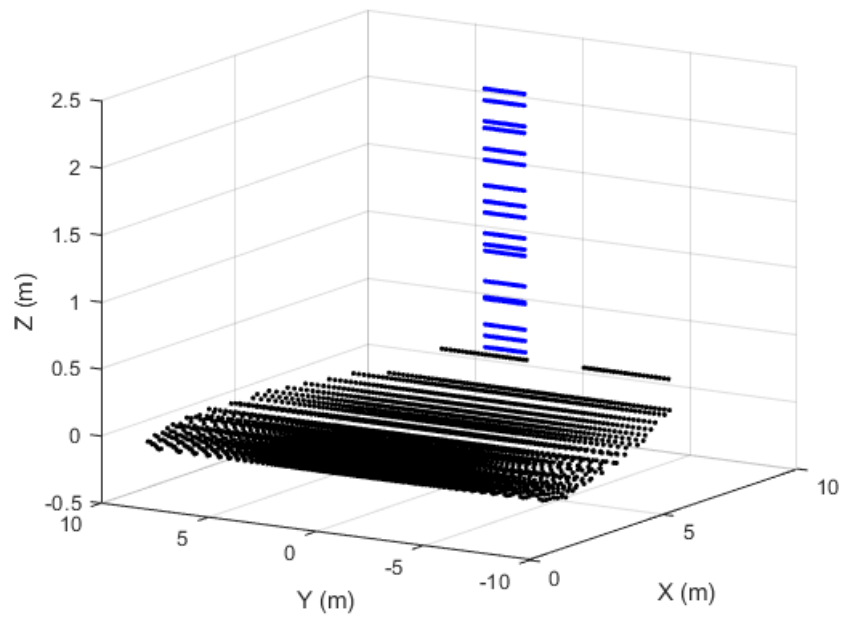


Figura 5.27: Nube de Puntos a 7 m ejemplo 2

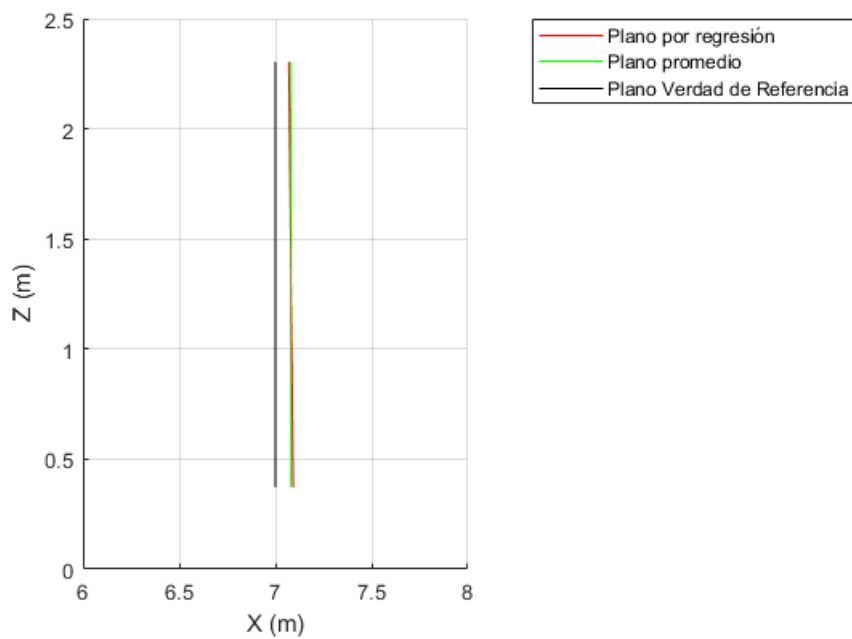


Figura 5.28: Comparación de planos a 7 m ejemplo 2

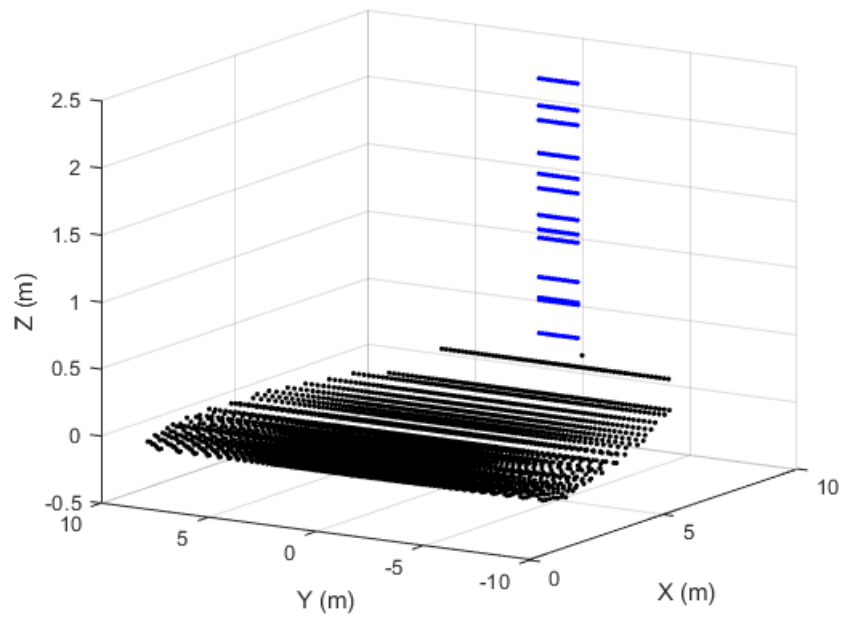


Figura 5.29: Nube de Puntos a 9 m ejemplo 1

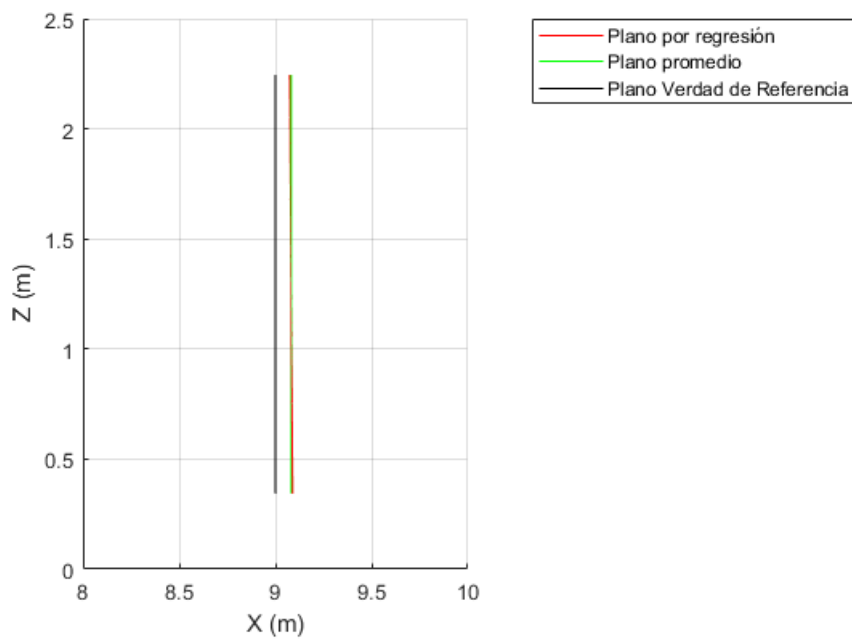


Figura 5.30: Comparación de planos a 9 m ejemplo 1

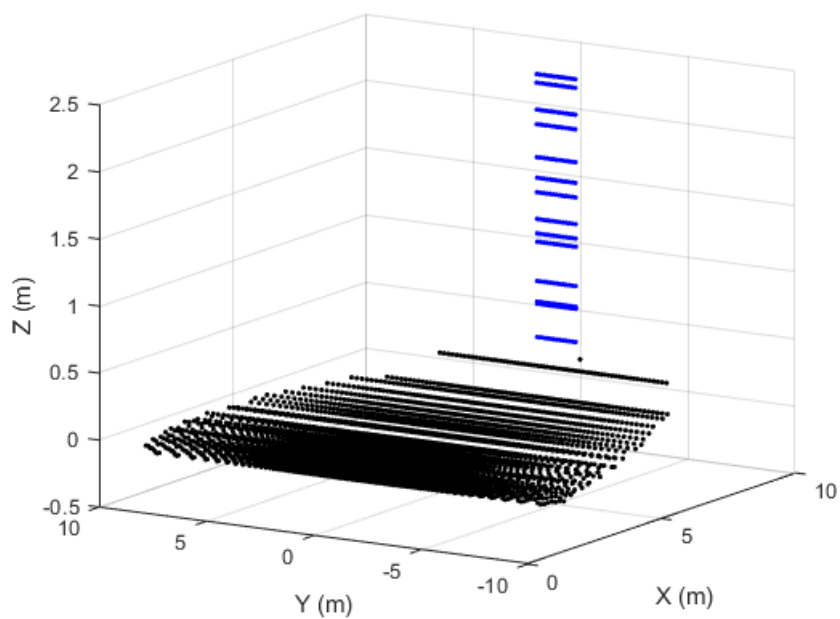


Figura 5.31: Nube de Puntos a 9 m ejemplo 2

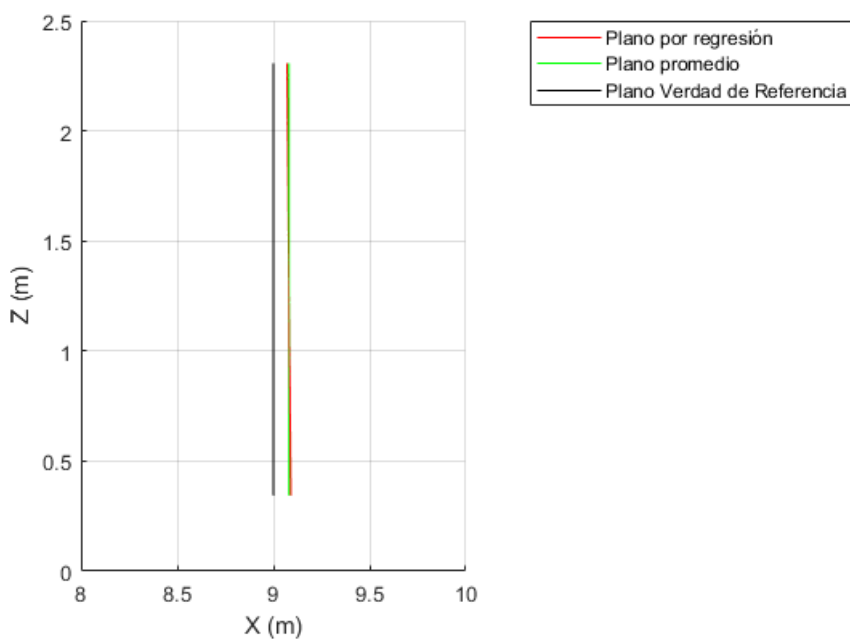


Figura 5.32: Comparación de planos a 9 m ejemplo 2

En la siguiente imagen se muestra la evolución del error conforme se va aumentando la distancia a la cual se encuentra la cara frontal de la caja.

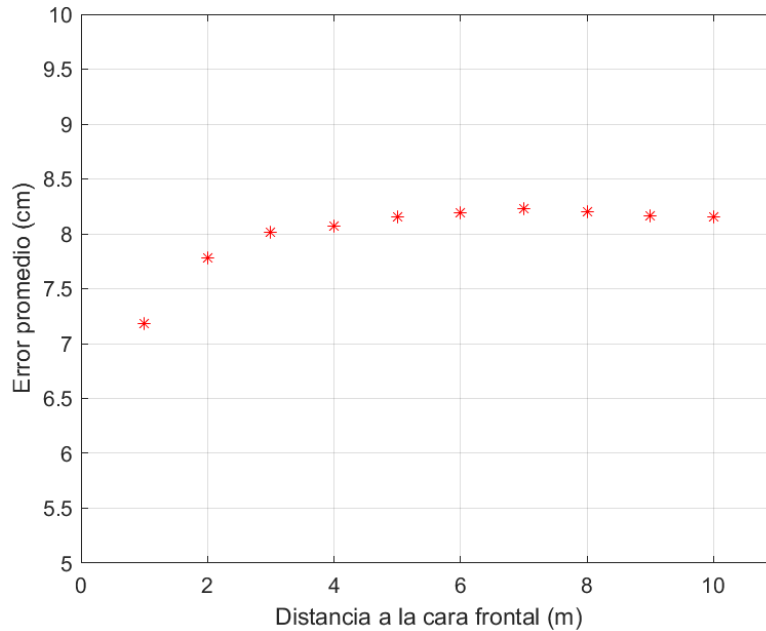


Figura 5.33: Error vs Distancia

## 5.4 Conclusiones del capítulo

En este capítulo se mostró la metodología seguida para la obtención de los resultados obtenidos en las pruebas conceptual y de desempeño. En la cual figuran las ecuaciones necesarias para obtener la información tridimensional de los escenarios de prueba y el proceso necesario para estimar el error que presenta el sistema al momento de realizar la construcción de una nube de puntos. También se observó el comportamiento del error obtenido, cuyo valor se encuentra entre 7.5 y 8 cm para distancias menores a los 10 m.

---

## CONCLUSIONES Y PERSPECTIVAS

---

### Conclusiones

En este proyecto de tesis, se desarrolló un sistema que permite obtener la reconstrucción de un entorno como una nube de puntos mediante el funcionamiento conjunto de un telémetro láser 2D y un dispositivo motriz. Con la información obtenida del sistema se crea la nube de puntos y así poder visualizar de manera gráfica dicha reconstrucción. Se realizó la simulación del sistema para evaluar su factibilidad y obteniendo la representación con una ligera incertidumbre debido al funcionamiento del servomotor y a las lecturas obtenidas del telémetro.

La reconstrucción 3D de un entorno representada como una nube de puntos se puede lograr de diferentes maneras, pero esta reconstrucción es el primer paso que se debe realizar para poder usar la nube de puntos y aplicarla para resolver alguna tarea específica. En base a los trabajos en el estado del arte se puede concluir que los telémetros láser 2D son muy utilizados en trabajos que tratan problemas de robótica móvil y visión por computadora, también se utilizan otros tipos de sensores en conjunto a uno más telémetros láser para buscar resolver problemáticas más complejas.

El software ANSYS nos permite poder simular el desempeño de una estructura bajo las condiciones especificadas, para poder saber tomar la decisión correcta para el diseño de los elementos que conformen un sistema. Mediante Gazebo y ROS se puede lograr la simulación de sistemas robóticos sin importar su grado de complejidad, y de esta forma poder tener una aproximación del funcionamiento del sistema en la realidad.

## Perspectivas

En base al análisis de los resultados obtenidos con este sistema, existen algunas mejoras que pueden ayudar al desempeño del sistema. Éstas se describen a continuación:

- Rediseñar el sistema manivela-biela-corredera mediante un modelo diferente que sea capaz de soportar las reacciones generadas y su velocidad de desplazamiento de tal forma que no afecte en gran medida el tiempo total para realizar el escaneo.
- Evaluar la incertidumbre en las otras dos dimensiones ancho (eje Y) y alto (eje Z) ya que solo se realizó para profundidad (eje X).

---

## BIBLIOGRAFÍA

---

- [1] S. Fu, H. ying Liu, L. fang Gao, and Y. xian Gai, "SLAM for mobile robots using laser range finder and monocular vision," in *2007 14th International Conference on Mechatronics and Machine Vision in Practice*, pp. 91–96, 2007.
- [2] J. Morales, J. L. Martínez, A. Mandow, A. Pequeño-Boyer, and A. García-Cerezo, "Design and development of a fast and precise low-cost 3D laser rangefinder," in *2011 IEEE International Conference on Mechatronics*, pp. 621–626, 2011.
- [3] Y. k. Wang, J. Huo, and X. s. Wang, "A real-time robotic indoor 3D mapping system using dual 2D laser range finders," in *Proceedings of the 33rd Chinese Control Conference*, pp. 8542–8546, 2014.
- [4] F. Maurelli, D. Droschel, T. Wisspeintner, S. May, and H. Surmann, "A 3D laser scanner system for autonomous vehicle navigation," in *2009 International Conference on Advanced Robotics*, pp. 1–6, 2009.
- [5] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1, pp. 519–528, 2006.
- [6] A. I. Comport, E. Marchand, and F. Chaumette, "Robust model-based tracking for robot vision," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, pp. 692–697, 2004.
- [7] P. Fechteler and P. Eisert, "Adaptive colour classification for structured light systems," in *IET Computer Vision*, vol. 3, pp. 49–59, 2009.
- [8] A. Kirchner and T. Heinrich, "Model based detection of road boundaries with a laser scanner," in *Proc. IEEE International Conference on Intelligent Vehicles*, vol. 1, pp. 93–98, 1998.
- [9] N. Cornelis, B. Leibe, K. Cornelis, and L. Van Gool, "3D urban scene modeling integrating recognition and reconstruction," *International Journal of Computer Vision*, vol. 78, no. 2, pp. 121–141, 2008.



- [10] O. Wulf and B. Wagner, "Fast 3D scanning methods for laser measurement systems," in *International conference on control systems and computer science (CSCS14)*, vol. 1, pp. 312–317, 2003.
- [11] H. Surmann, K. Lingemann, A. Nüchter, and J. Hertzberg, "A 3D laser range finder for autonomous mobile robots," in *Proceedings of the 32nd ISR (International Symposium on Robotics)*, vol. 19, pp. 153–158, 2001.
- [12] D. Hahnel, D. Schulz, and W. Burgard, "Map building with mobile robots in populated environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, pp. 496–501, 2002.
- [13] Y.-S. Chou and J.-S. Liu, "A robotic indoor 3D mapping system using a 2D laser range finder mounted on a rotating four-bar linkage of a mobile platform," *International Journal of Advanced Robotic Systems*, vol. 10, no. 1, p. 45, 2013.
- [14] "Rieg1 laser measurement systems," 2001. Data sheet, LMS-Z210.
- [15] H. Surmann, A. Nuchter, and J. Hertzberg, "An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments," *Robotics and Autonomous Systems*, vol. 4, no. 3, pp. 181 – 198, 2003.
- [16] C. Brenneke, O. Wulf, and B. Wagner, "Using 3D laser range data for slam in outdoor environments," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, vol. 1, pp. 188–193 vol.1, 2003.
- [17] L. C. Goron, L. Tamas, I. Reti, and G. Lazea, "3D laser scanning system and 3D segmentation of urban scenes," in *2010 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, vol. 1, pp. 1–5, 2010.
- [18] P. Olivka, M. Mihola, P. Novák, T. Kot, and J. Babjak, "The 3D laser range finder design for the navigation and mapping for the coal mine robot," in *2016 17th International Carpathian Control Conference (ICCC)*, pp. 533–538, 2016.
- [19] Y. G. Moon, S. J. Go, K. H. Yu, and M. C. Lee, "Development of 3D laser range finder system for object recognition," in *2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1402–1405, 2015.
- [20] O. S. R. Foundation, "Gazebo licensed under apache2.0," 2002.
- [21] O. S. R. Foundation, "Robotic operating system ROS," 2007.
- [22] R. G. Budynas, J. K. Nisbett, *et al.*, *Shigley's mechanical engineering design*, vol. 8. McGraw-Hill New York, 2008.

- [23] J. Lentin, *Mastering ROS for Robotics Programming*. Packt Publishing Ltd  
Birmingham, 2015.