

**PONTIFICIE UNIVERSIDAD CATÓLICA DEL PERÚ**  
**ESCUELA DE POSGRADO**



**DISEÑO DE UN SISTEMA DE CONTROL BASADO EN  
LINEALIZACIÓN POR REALIMENTACIÓN PARA UN ROBOT  
MÓVIL TIPO ACKERMAN CON VELOCIDAD VARIABLE Y  
MOVIMIENTO EN DOBLE SENTIDO DESCRIBIENDO  
TRAYECTORIAS ÓPTIMAS**

Tesis para optar el grado de  
Magíster en Ingeniería de Control y Automatización

**Autor: Ing. Luis Angel Velasco Mellado**

**Asesor: PhD. Antonio Morán Cárdenas**

**Miembros del Jurado:**

MSc. Francisco Cuéllar Córdova

MSc. Pedro Crisóstomo Romero

Lima – Perú

2017

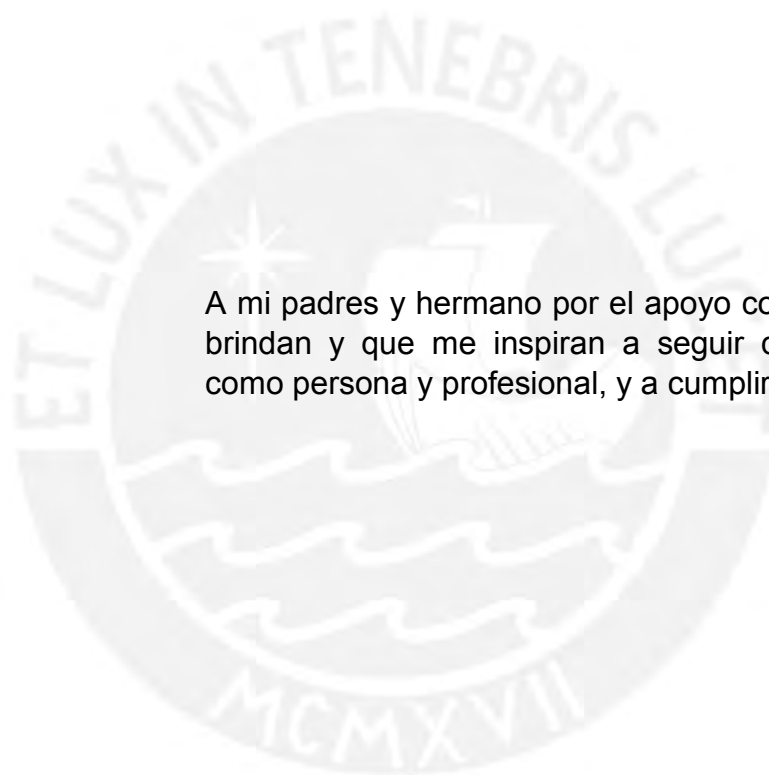
## RESUMEN

Desde varios años, las investigaciones en el tema del uso de robots móviles han ido en aumento con el objetivo de poder utilizar los robots móviles para múltiples actividades que suelen ser difícil para una persona realizar; es por eso que, se ha ido desarrollando técnicas de control para distintos tipos de robots móviles, así como también en la mejora de su desempeño en trayectorias a recorrer frente a entornos al cual se aplique. El desarrollo se enfoca principalmente en el control del robot móvil, ya que existen varios métodos para controlarlos, ya sea de manera lineal o de manera no lineal, dependiendo sea el caso, llevándolos a llegar a su objetivo siguiendo ciertas trayectorias específicas o trayectorias más cortas a su objetivo. Para los casos de trayectorias específicas, suelen definirse formas geométricas conocidas y fáciles de realizar; sin embargo, para el caso de trayectorias con la distancia más corta, aún sigue habiendo más investigación a nivel aplicativo. Es por eso que se busca desarrollar un método para la generación de trayectoria mínima enfocado para un robot móvil tipo Ackerman, que permita llevarlo a su objetivo, considerando que el robot móvil funcionará con movimiento en avance y retroceso en cualquier momento.

Esta investigación busca aplicar una técnica de control no lineal, para así trabajar en todo el espacio de funcionamiento del vehículo, que le permita al robot móvil describir trayectorias óptimas, la cual va a ser diseñada en base al análisis de movimiento que tiene un tipo Ackerman, considerando que se puede cambiar de velocidad en una misma trayectoria; para lo cual se ha realizado la obtención del modelo del tipo Ackerman y así definir correctamente el controlador a utilizar, mientras que a su vez se desarrolla el algoritmo de generación de trayectoria mínima y al final realizar la corroboración del funcionamiento en base a simulación y luego con un prototipo.

El desarrollo de la investigación se ha realizado analizando diferentes técnicas de control, decidiendo al final utilizar un control basado en linealización por realimentación utilizando el método de flatness diferencial, el cual permite introducir de manera práctica una trayectoria deseada a seguir por el robot móvil en sus variables de posición. Dicha trayectoria deseada ha sido diseñada mediante un análisis geométrico compuesto por segmentos de rectas y curvas de circunferencia, basado en la teoría de Dubins y Reed & Shepp, permitiendo describir la ruta más corta posible desde el punto de origen a su punto de destino. Las pruebas realizadas de su funcionamiento han sido simuladas bajo un software que permitió validar el controlador y la correcta generación de una ruta mínima; y también se realizó pruebas en un prototipo para corroborar que puede ser aplicable en un sistema real

Al final de la investigación se muestra como el controlador permite llevar al robot móvil a su objetivo describiendo trayectorias óptimas, generando la velocidad y el ángulo de dirección adecuados que permitan guiarlo por dicha ruta; considerando también un sistema de navegación para la aplicación real del sistema.



A mi padres y hermano por el apoyo constante que me brindan y que me inspiran a seguir desarrollándome como persona y profesional, y a cumplir mis metas.

## Agradecimientos

Agradezco al Dr. Antonio Morán por el gran asesoramiento que me ha brindado para el desarrollo de esta tesis.

El autor de este trabajo de tesis, *Luis Angel Velasco Mellado*, agradece la subvención de FONDECYT-CONCYTEC a través del convenio 2015-034 FONDECYT, en el marco del cual se desarrolló la presente tesis: *“Diseño de un Sistema de Control Basado en Linealización por Realimentación para un Robot Móvil tipo Ackerman con Velocidad Variable y Movimiento en Doble Sentido Describiendo Trayectorias Óptimas”*.

## Índice General

<b>CAPÍTULO 1. PRESENTACIÓN DEL PROBLEMA Y ESTADO DEL ARTE.....</b>	<b>1</b>
1.1. Introducción.....	1
1.2. Análisis de los Robots Móviles .....	2
1.2.1. Conceptos Básicos.....	2
1.2.2. Estructuras .....	3
1.3. Estado del Arte para Planeamiento de Trayectoria .....	5
1.4. Estado del Arte de Controladores No Lineales para Robots Móviles .....	8
1.5. Objetivos de la Tesis .....	12
1.5.1. Objetivos Generales.....	12
1.5.2. Objetivos Específicos .....	13
<b>CAPÍTULO 2. ANÁLISIS DEL MODELO DEL ROBOT TIPO ACKERMAN.....</b>	<b>14</b>
2.1. Modelo General del Robot Móvil.....	14
2.2. Modelo Cinemático del Robot Móvil Tipo Ackerman .....	15
2.3. Modelo Cinemático del Robot Móvil Tipo Diferencial .....	18
2.4. Esquema General para el Control y Navegación .....	18
<b>CAPÍTULO 3. PLANEAMIENTO DE TRAYECTORIA DESEADA.....</b>	<b>20</b>
3.1. Diseño de Ruta Óptima .....	20
3.1.1. Introducción.....	20
3.1.2. Cálculo de Líneas Rectas y Arcos de Círculos para Ruta Mínima .....	24
3.2. Generación de Trayectoria para Velocidad Variable .....	30
3.2.1. Función Polinómica .....	30
3.2.2. Cálculo de los Parámetros del Polinomio.....	32
3.2.3. Selección del Grado del Polinomio .....	34
3.2.4. Generación del Vector de Puntos Utilizando Función Polinómica .....	37
3.3. Generación de Trayectoria para Tiempo Mínimo .....	39
3.3.1. Fundamentos Teóricos de Control Óptimo.....	39
3.3.2. Problema de Control Óptimo para Tiempo Mínimo .....	42
3.3.3. Generación del Vector de Puntos Utilizando la Función Switching.....	45
3.4. Conclusiones Preliminares .....	48
<b>CAPÍTULO 4. DISEÑO DEL SISTEMA DE CONTROL DEL ROBOT MÓVIL .....</b>	<b>50</b>
4.1. Introducción.....	50
4.2. Linealización por Realimentación .....	51

4.3.	Flatness Diferencial .....	52
4.4.	Control de Trayectoria .....	54
4.4.1.	Esquema de Control .....	55
4.4.2.	Cálculo de las Ganancias del Controlador .....	56
4.5.	Simulación del Sistema de Control usando Trayectoria Basada en Polinomio .....	58
4.6.	Simulación del Sistema de Control usando Trayectoria Basada en la Función Switching .....	64
4.7.	Conclusiones Preliminares .....	69
 <b>CAPÍTULO 5. DISEÑO DEL SISTEMA DE NAVEGACIÓN PARA EL ROBOT MÓVIL .....</b>		<b>71</b>
5.1.	Introducción .....	71
5.2.	Sensores Inerciales .....	73
5.2.1.	Acelerómetro .....	73
5.2.2.	Giroscopio .....	74
5.3.	Magnetómetro .....	75
5.4.	Calibración de los Sensores .....	78
5.5.	Encoder .....	84
5.6.	Fusión de Sensores Utilizando Filtro de Kalman .....	87
5.6.1.	Fundamentos Teóricos del Filtro de Kalman .....	88
5.6.2.	Estimación del Ángulo de Orientación .....	90
5.6.3.	Estimación de la Posición .....	93
5.7.	Conclusiones Preliminares .....	96
 <b>CAPÍTULO 6. IMPLEMENTACIÓN DEL SISTEMA DE CONTROL Y NAVEGACIÓN .....</b>		<b>98</b>
6.1.	Introducción .....	98
6.2.	Estructura y Hardware del Robot Móvil .....	98
6.3.	Implementación del Código en C++ .....	101
6.4.	Resultados del Sistema de Control y Navegación .....	103
6.5.	Conclusiones Preliminares .....	106
 <b>CONCLUSIONES .....</b>		<b>107</b>
<b>BIBLIOGRAFÍA .....</b>		<b>109</b>
<b>ANEXOS .....</b>		<b>114</b>

## Índice de Figuras

Figura 1.1: Centro de Rotación de un Robot Móvil.....	2
Figura 1.2: Restricción No-Holonómica de un Robot Móvil .....	3
Figura 1.3: Categoría de Trayectorias.....	5
Figura 1.4: Esquema de Planeamiento de Trayectoria.....	6
Figura 1.5: Ejemplo de Curvas Bezier Usando Interpolacion de Puntos.....	6
Figura 1.6: Ejemplos de Trayectorias de Dubins.....	7
Figura 1.7: Diagrama de un Controlador Lógico Difuso.....	9
Figura 1.8: Diagrama de una Red Neuro-Difusa .....	9
Figura 2.1: Diagrama del Robot Móvil Tipo Ackerman .....	16
Figura 2.2: Modelo Basado en Tipo Bicicleta para un Tipo Carro.....	16
Figura 2.3: Robot Móvil Tipo Diferencial .....	18
Figura 2.4: Esquema de Control y de Navegación Propuesto .....	19
Figura 3.1:Tipos de Direcciones de Dubins.....	21
Figura 3.2: Tipos de Direcciones de Reeds y Shepp (Straight), R(Right) .....	21
Figura 3.3: Posición Inicial (rojo) y Final (Azul) del Robot Móvil .....	22
Figura 3.4: Posible Mínima Trayectoria a Calcular .....	23
Figura 3.5: Circunferencias Formada en la Posición Inicial y la Posición Final.....	24
Figura 3.6: Tipos de Rectas Tangentes entre Circunferencias.....	25
Figura 3.7: Generación de Todas las Rectas Tangentes Posibles .....	25
Figura 3.8: Doble Camino de Movimiento Posible en los Círculos .....	26
Figura 3.9: Trayectoria Mínima Obtenida de Todas las Trayectorias Posibles .....	28
Figura 3.10: Comparación de una Situación con 2 Trayectorias Óptimas .....	28
Figura 3.11: Comparaciones entre Trayectoria con Velocidad en un Sentido y con Ambos Sentidos.....	29
Figura 3.12: Categorías de Trayectoria.....	31
Figura 3.13: Trayectoria para la Posición, Velocidad y Aceleración con Polinomio de Orden 3 .....	35
Figura 3.14: Trayectoria para la Posición, Velocidad y Aceleración con Polinomio de Orden 5 .....	36
Figura 3.15: Diferencia de la Respuesta Polinomial entre el 3er y 5to Orden .....	36
Figura 3.16: Ejemplo de Trayectoria Multipunto .....	38
Figura 3.17: Recta Imaginaria para el Cálculo del Sentido de la Velocidad.....	45
Figura 3.18: Recta Imaginaria para el Cálculo del Sentido del Ángulo $\delta$ .....	46
Figura 3.19: Lazo de Estimación de Puntos de la Trayectoria Deseada.....	47
Figura 4.1: Esquema de Control de Trayectoria.....	56
Figura 4.2: Trayectoria con Función Polinómica – Movimiento en Avance .....	60
Figura 4.3: Señales de las Variables de Estado con Función Polinómica – Movimiento en Avance.....	60
Figura 4.4: Señales de las Variables de Control con Función Polinómica – Movimiento en Avance.....	61
Figura 4.5: Trayectoria con Función Polinómica – Movimiento en Retroceso.....	61
Figura 4.6: Señales de las Variables de Estado con Función Polinómica – Movimiento en Retroceso .....	62
Figura 4.7: Señales de las Variables de Control con Función Polinómica – Movimiento en Retroceso .....	62
Figura 4.8: Trayectoria con Función Polinómica – Movimiento Variable.....	63
Figura 4.9: Señales de las Variables de Estado con Función Polinómica – Movimiento Variable .....	63

Figura 4.10: Señales de las Variables de Control con Función Polinómica – Movimiento Variable .....	64
Figura 4.11: Trayectoria con Función Switching – Movimiento en Avance .....	65
Figura 4.12: Señales de las Variables de Estado con Función Switching – Movimiento en Avance.....	65
Figura 4.13: Señales de las Variables de Control con Función Switching – Movimiento en Avance.....	66
Figura 4.14: Trayectoria con Función Switching – Movimiento en Retroceso .....	66
Figura 4.15: Señales de las Variables de Estado con Función Switching – Movimiento en Retroceso .....	67
Figura 4.16: Señales de las Variables de Control con Función Switching – Movimiento en Retroceso .....	67
Figura 4.17: Trayectoria con Función Switching – Movimiento en Retroceso .....	68
Figura 4.18: Señales de las Variables de Estado con Función Switching – Movimiento Variable .....	68
Figura 4.19: Señales de las Variables de Control con Función Switching – Movimiento en Retroceso .....	69
Figura 5.1: Ángulos de Rotación en un Sistema de Coordenadas .....	72
Figura 5.2: Ángulos de Rotación de un Robot Móvil.....	72
Figura 5.3: Medición del Acelerómetro en Base a una Fuerza Generada .....	73
Figura 5.4: Giroscopio Mecánico Convencional .....	74
Figura 5.5: Campo Magnético de la Tierra .....	75
Figura 5.6: Declinación del Norte Geográfico con Respecto al Norte Magnético.....	76
Figura 5.7: Información de la Declinación .....	76
Figura 5.8: Ángulo de Azimut con Respecto a los Ejes del Magnetómetro.....	77
Figura 5.9: Inclinaciones del Magnetómetro en un Plano Horizontal .....	78
Figura 5.10: Módulo MPU-92/65 para el Análisis de Movimiento .....	78
Figura 5.11: Mediciones del Acelerómetro sin Calibrar .....	80
Figura 5.12: Mediciones del Acelerómetro Calibrado .....	81
Figura 5.13: Mediciones del Giroscopio sin Calibrar .....	81
Figura 5.14: Mediciones del Giroscopio Calibrado .....	82
Figura 5.15: Mediciones del Magnetómetro sin Calibrar .....	82
Figura 5.16: Mediciones del Magnetómetro sin Calibración 2D.....	83
Figura 5.17: Mediciones del Magnetómetro Calibrado 2D.....	84
Figura 5.18: Esquema General de un Encoder Óptico .....	85
Figura 5.19: Señales del Encoder en Cuadratura .....	86
Figura 5.20: Conexión del Encoder al Motor .....	86
Figura 5.21: Lazo del Proceso del Filtro de Kalman .....	90
Figura 5.22: Respuesta del Filtro de Kalman para una Señal Sinusoidal .....	91
Figura 5.23: Señales de las Mediciones del Giroscopio .....	92
Figura 5.24: Señal del Ángulo de Orientación Calculado con el Magnetómetro .....	92
Figura 5.25: Respuesta del Filtro de Kalman .....	93
Figura 5.26: Mediciones del Acelerómetro .....	95
Figura 5.27: Medición del Encoder. (a) Señal completa, (b) Zoom en un Tramo.....	95
Figura 5.28: Trayectoria Circular del Robot Utilizando Filtro de Kalman.....	96
Figura 6.1: Raspberry Pi 3 .....	99
Figura 6.2: Módulo Encoder FC-03.....	100
Figura 6.3: Servo SG-5010 .....	100
Figura 6.4: Módulo Driver L298.....	101
Figura 6.5: Prototipo de un Robot Móvil Tipo Ackerman .....	101
Figura 6.6: Trayectoria Real 1.....	104



Figura 6.7: Trayectoria Real 2.....	104
Figura 6.8: Trayectoria Real 3.....	105
Figura 6.9: Trayectoria Real 4.....	105



## Índice de Tablas

Tabla 1-1: Estructuras de Robots Móviles .....	4
Tabla 3-1: Distancia de Recorrido Utilizando la Teoría de Dubins o Reeds & Shepp .....	30
Tabla 6-1: Errores entre los Valores Finales Reales y los Deseados .....	106



# CAPÍTULO 1. PRESENTACIÓN DEL PROBLEMA Y ESTADO DEL ARTE

## 1.1. Introducción

Con el paso de los años, se ha ido observando un gran crecimiento en el uso de robots móviles tanto aéreo, terrestre y acuático en muchos ámbitos y áreas de investigación, con el fin de facilitar el acceso a lugares que resultan difíciles para un ser humano como transportar materiales en entornos complejos, analizar grandes áreas de manera autónoma, entre otras actividades.

El área de investigación, en el desarrollo de estos sistemas robóticos, se ha basado en el desarrollo de la estructura del robot de acuerdo a la aplicación que se le dará, la electrónica para controlar las partes y sensar el entorno en el cual se encuentra el robot y el aspecto más importante y de más desarrollo investigativo, la técnica de control. Esta última es la que permite al robot tener un comportamiento autónomo conociendo su entorno, y logrando su objetivo describiendo trayectorias óptimas para alcanzar su objetivo final, de esta manera no requiere el control manual de una persona que lo esté guiando a lo largo de su trayecto.

Tomando el caso particular de los robots móviles terrestres o conocidos también como UGV (Unmanned Ground Vehicle), las técnicas de control que se desarrollan parten desde linealizar el modelo del robot y trabajar en una vecindad específica de trabajo con controladores lineales, para continuar con controladores no lineales que permiten trabajar en una mayor área de alcance, para lo cual existen varios métodos que combinan diferentes técnicas de control dependiendo de la estructura del robot.

Los UGV se clasifican dependiendo de su estructura de locomoción, lo que lleva a que haya distintos métodos de control en función de las variables que definen el movimiento del robot. Enfocándose en un caso particular como son los robots “tipo carro” o “Ackerman” que funcionan con 4 ruedas (2 ruedas de tracción y 2 ruedas de dirección), las investigaciones se han dado principalmente en el control del ángulo de giro de las ruedas de dirección y poco en el control de velocidad, ya que algunos trabajan con velocidad constantes por cuestiones prácticas.

Debido a que en el caso real se desea lograr que un robot móvil se pueda mover libremente en múltiples direcciones y llegar a su destino describiendo la trayectoria

más rápida, se plantea realizar el control de un robot móvil que permita llegar a su punto de destino enfatizando la generación de velocidad variable y cambio en el sentido de la velocidad (avance y retroceso), ya que de esta manera el robot se podrá movilizar describiendo la trayectoria más corta a su destino. Al integrar ambos sentidos, el robot tendrá más opciones de ruta para llegar a su destino en el tiempo y distancia más corta.

## 1.2. Análisis de los Robots Móviles

### 1.2.1. Conceptos Básicos

#### □ Tipos de Ruedas:

- Rueda Motriz: la que proporciona fuerza de tracción al robot
  - Rueda Directriz: ruedas de direccionamiento de orientación controlable
  - Ruedas Fijas: solo giran en torno a su eje sin tracción motriz (Tipo Ackerman)
  - Ruedas Locas: ruedas orientables no controladas
- 
- Centro Instantáneo de Rotación (CIR): es el punto de intersección de todos los ejes de las ruedas cuando el ángulo de las llantas de dirección es mayor a cero como se muestra en la figura 1.1.

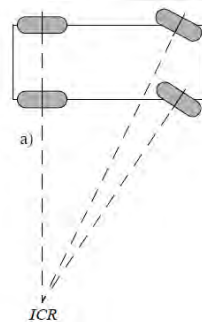


Figura 1.1: Centro de Rotación de un Robot Móvil (Siegwart, Nourbakhsh & Scaramuzza, 2011)

Esta característica es muy útil debido a que se puede conocer el radio de rotación partiendo del valor del ángulo de giro de las ruedas de dirección. Conociendo el ángulo máximo de giro de las ruedas de dirección, se puede calcular el radio mínimo de la circunferencia formada por el movimiento realizado del robot mediante la siguiente ecuación:

$$R_{min} = \frac{L}{\tan(\delta_{max})} \quad (1.1)$$

Siendo  $\delta_{max}$  el máximo ángulo de giro que pueden tener las ruedas de dirección.

- **Restricción No-Holonómica:** Significa que un robot puede moverse hacia adelante o hacia atrás, pero no de manera lateral debido a que las ruedas que utiliza no permiten un desplazamiento lateral como se muestra en la figura 1.2.



Figura 1.2: Restricción No-Holonómica de un Robot Móvil

### 1.2.2. Estructuras

Los robots móviles se diferencian, principalmente, por su tipo de locomoción: ruedas, patas, etc.; donde en este trabajo de investigación se analiza la locomoción por ruedas, generando varias estructuras, donde sus configuraciones parten desde un modelo básico en algunos casos o modelos surgidos en base a nuevas investigaciones.

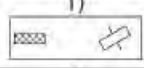
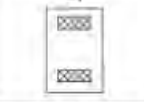

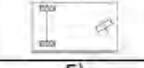
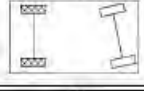
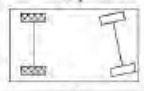

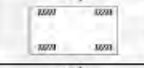

# ruedas	Estructura	Características	Ejemplos
2	1) 	Dirección delantera y tracción posterior.	Bicicleta
	2) 	Dos ruedas diferenciables	Cye robot
3	3) 	Dos ruedas centrales diferenciables y una tercera con punto de contacto.	smartRobot
	4) 	Dos ruedas traseras de tracción y una delantera de dirección.	Piaggio robots
	5) 	Dos ruedas traseras libres y una delantera de dirección y tracción	Neptune robot
4	6) 	Dos ruedas traseras de tracción y dos delanteras de dirección	Carros de conducción trasera
	7) 	Dos ruedas traseras de dirección y dos delanteras de tracción	Carros de conducción delantera
	8) 	Cuatro ruedas omnidireccionales	Carnegie Mellon Uranus
	9) 	Dos ruedas diferenciables y 2 ruedas con punto de contacto	Khepera

Tabla 1-1: Estructuras de Robots Móviles (Enciso,2015)

El robot a utilizar en esta investigación es el carro de conducción delantera o también conocido como robot tipo carro o Ackerman, el cual tiene una locomoción basada en 4 ruedas fijas con tracción trasera y conducción delantera. Esto quiere decir que las ruedas traseras son utilizadas para generar la fuerza de movimiento del vehículo, mientras que las ruedas delanteras sirven para dirigir el vehículo.

La diferencia en cada tipo de robot móvil parte principalmente en su estructura, pero lleva a analizar el tipo de movimientos que estos generan y la forma de controlarlos de acuerdo a sus características. Por ejemplo, los robots tipo CyeRobot y SmartRobot, conocidos también como diferenciales, tienen la peculiaridad de que pueden girar en su propio eje, ya que su movimiento se basa en el sentido de la velocidad que tiene cada rueda independiente de la otra, lo que lleva a un tipo de control distinto al que podría tener un tipo Ackerman, que no es capaz de girar en su propio eje debido a que sus llantas de tracción se encuentran conectados en un

mismo eje y sus movimientos son uniformes teniendo el sentido de la velocidad iguales para ambas ruedas de tracción.

### 1.3. Estado del Arte para Planeamiento de Trayectoria

Planeamiento o generación de trayectoria es el proceso de planear o generar el movimiento que debe seguir el movimiento del robot las cuales van dirigidas directamente a las variables de estado. La trayectoria deseada es directamente la entrada de referencia al controlador. El proceso de planeamiento se basa en tener las condiciones del sistema, restricciones que tiene el robot y restricciones de cinemática (Gasparetto, Boscariol, Lanzutti & Vidoni,2012). Existen varias categorías en el tema de trayectoria como se muestra en la figura 1.3.

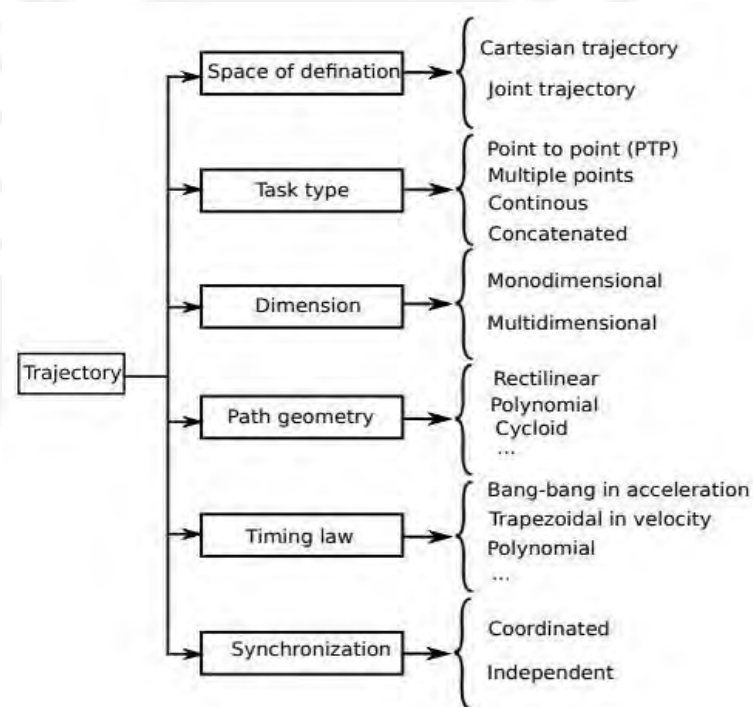


Figura 1.3: Categoría de Trayectorias (Zhao,2015)

Lo principal del planeamiento de trayectoria es la búsqueda de la mejor trayectoria posible que permita al móvil llegar a su objetivo en un corto camino. Existen múltiples técnicas para el planeamiento de trayectoria; entre los más prácticos tenemos las trayectorias de formas geométricas conocidas, hasta trayectorias

óptimas que describen la menor distancia posible y que depende de la locomoción del sistema (Gasparetto, Boscariol, Lanzutti & Vidoni, 2012; Murray & Sastry, 1993).

La trayectoria debe ser diseñada también en base al comportamiento que se quiere que tenga el vehículo, cumplir las condiciones iniciales y finales de posicionamiento, y también en base a las restricciones que pueda haber en el entorno de trabajo (obstáculos), generando así una función en el tiempo de la trayectoria deseada como se muestra en la figura 1.4.

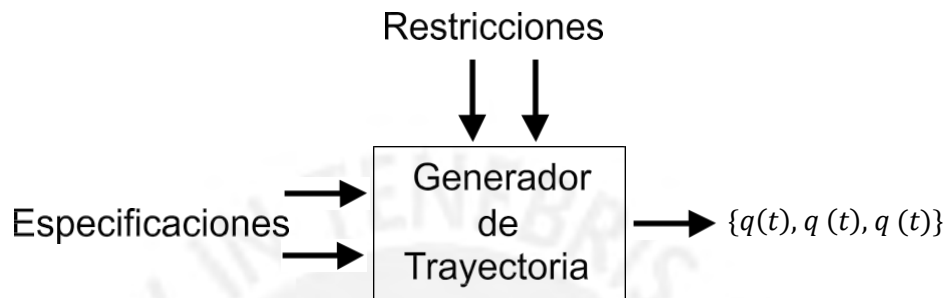


Figura 1.4: Esquema de Planeamiento de Trayectoria

Un método muy utilizado es la generación mediante interpolación de puntos o curvas de Bézier, donde la trayectoria es construida en base a varios segmentos de Bézier, cada uno interpolado entre 2 posiciones claves de la trayectoria. Este método es utilizado ampliamente en muchos casos de dibujo al momento de crear curvas (Gloderer & Hertle, 2010).

Las curvas de Bézier son curvas paramétricas polinómicas, las cuales buscan aproximarse mediante el método de interpolación a ciertos puntos dados tal como se muestra en la figura 1.5.

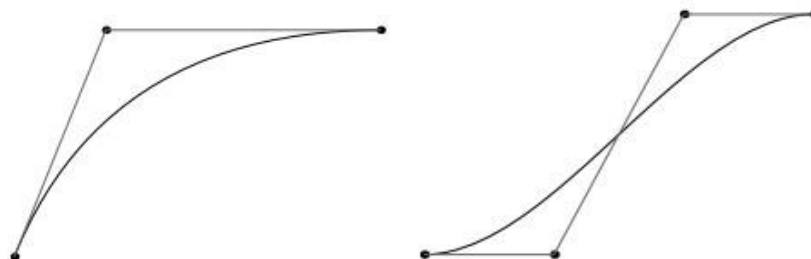


Figura 1.5: Ejemplo de Curvas Bezier Usando Interpolación de Puntos



A partir de este método, se han desarrollado otras técnicas de generación de trayectoria basada en interpolación de puntos y formas polinómicas, pero esta técnica no asegura una trayectoria óptima, únicamente una trayectoria posible que permita llevar al robot móvil a su objetivo final (Guo & Tang, 2008; Zhao, 2015).

En el tema de planeamiento de trayectoria óptima, es decir mínima trayectoria, para el caso de los robots móviles tipo ackerman, se utiliza la teoría de control óptimo para obtener una trayectoria mínima partiendo de minimizar el tiempo que se demore el vehículo en llegar a su objetivo y resolviendo la función de costo. Un algoritmo diseñado a nivel aplicativo, partiendo de este concepto, es el software DIDO, el cual resuelve todo tipo de problemas de control óptimo y funciona mediante el software Matlab.

También existe el análisis geométrico, sobre el movimiento que realiza un robot móvil, para resolver y generar una trayectoria mínima. El primer estudio fue realizado por Dubins, que demostró que un robot tipo carro con condiciones iniciales de posición y orientación puede llegar a su destino por el camino más corto con máximo 3 segmentos, que pueden ser arcos de círculos de mínimo radio y/o segmentos de recta realizando movimientos en avance (Lugo-Cárdenas, Flores, Salazar & Lozano, 2014; Furtuna & Balkcom, 2010). Mediante estos segmentos genera trayectorias óptimas, por ejemplo, como las que se muestran en la figura 1.6.

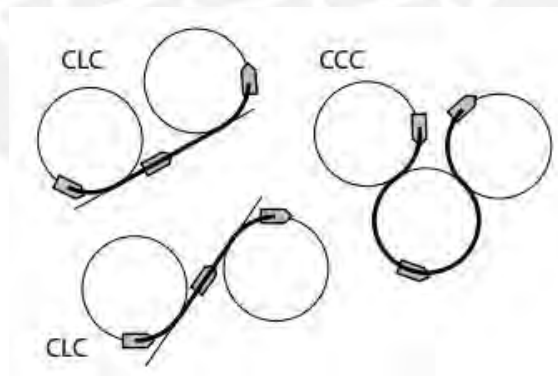


Figura 1.6: Ejemplos de Trayectorias de Dubins (Furtuna & Balkcom,2010)

En el método de Dubins se definen criterios ya establecidos para saber que ruta elegir de acuerdo al objetivo del vehículo, por lo que no requiere un análisis complejo. Sin embargo, Dubins solo considera velocidad en un sentido, por lo que luego, Reeds y Shepp resuelven un problema similar solo que aplicando un movimiento del vehículo en avance y retroceso (Fraichard & Scheuer, 2004), pero

que en este caso no se han definido criterios exactos considerando solo el objetivo del vehículo por lo que se plantean análisis geométricos para obtener la trayectoria mínima.

Se plantea utilizar el método de Reeds y Shepp para obtener la ruta mínima considerando ambos sentidos de movimiento en simultaneo en una misma ruta y elaborar un algoritmo que haga el cálculo automático aplicable para todo tipo de programación, partiendo de un análisis geométrico de cómo es el movimiento del robot móvil, para luego utilizar los criterios de curvas de Bezier y el concepto de la teoría de control óptimo para realizar el vector de puntos de la trayectoria deseada, destacando las ventajas de cada técnica de generación de puntos..

#### **1.4. Estado del Arte de Controladores No Lineales para Robots Móviles**

En la actualidad, existen varios métodos de diseño de controladores para robots móviles en base a las variables que se desean controlar, ya que para robots móviles terrestres las variables a controlar suelen ser velocidad y el ángulo de dirección, lo que conlleva en algunos diseños en optar por poner más énfasis al ángulo de dirección que a la velocidad. Por eso algunos controladores están diseñados más para el caso de velocidad constante y ángulo de dirección variable, siendo esta última la única variable de control del sistema. También los diseños de los controladores dependen mucho de si se desea trabajar en todo el rango de trabajo o alrededor de puntos de operación.

Los controladores más utilizados y recomendados, debido a su amplio margen de funcionamiento, son los controladores no lineales ya que estos permiten manipular al robot móvil en todo su rango de trabajo. Durante años, se han planteado varias técnicas de control no lineal entre las cuales destacan ciertos controladores ya que son los más utilizados.

Se tienen los controladores neuronales, difusos y neuro-difusos, que son controladores basándose en el aprendizaje como en el caso del controlador neuronal, basado en la experiencia como en los difusos y una mezcla de ambos que serían los controladores neuro-difusos. Con el avance de los años ha habido ciertas mejoras en cada tipo de técnica mezclada con otros controladores lineales para mejorar aún más el desempeño del control.

En la figura 1.7 se muestra un diagrama de un controlador difuso para el control de un vehículo.

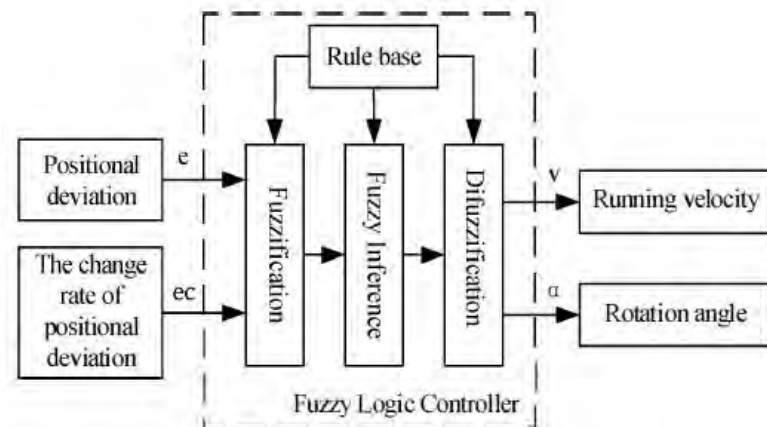


Figura 1.7: Diagrama de un Controlador Lógico Difuso (Fan, Yang, Li & Yan,2012)

Este controlador difuso permite calcular las variables de control de velocidad y ángulo de dirección en base al conocimiento del error de la posición y la variación del error entre muestras. Esta técnica es bien práctica y tiene buen funcionamiento ya que está basado en conocimientos previos del funcionamiento de un robot y que requiere de la experiencia del hombre para realizar el diseño.

La figura 1.8 muestra el diagrama de un controlador neuro-difuso.

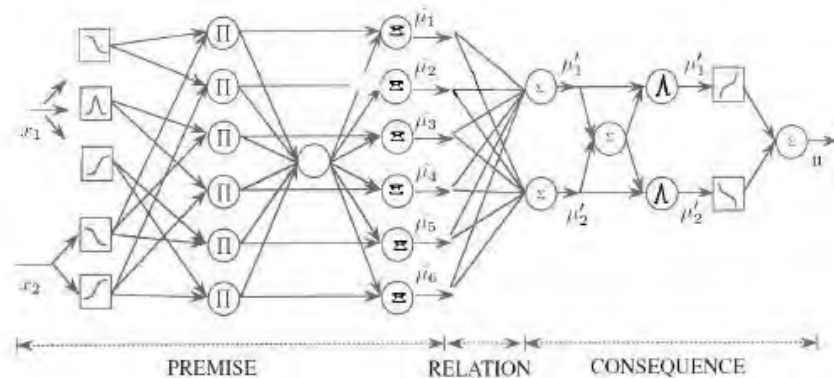


Figura 1.8: Diagrama de una Red Neuro-Difusa (Moran, Razuri, Sundgren & Rahmani,2013)

El controlador neuro-difuso es una combinación de la red neuronal y red difusa, que en algunos sistemas tienen un mejor desempeño del control del sistema, ya que otros sistemas ya tienen un desempeño óptimo solo con red neuronal o red difusa.

Cabe mencionar que en varios casos se mezclan este tipo de controlador con otros controladores lineales ya sea un PID, control adaptativo, etc.; y que sirven para casos lineales específicos en el funcionamiento.

Otra técnica de control utilizada ampliamente es el control mediante linealización por realimentación que conlleva a calcular la señal de control de una forma lineal, realizando transformaciones al sistema haciendo cambio de variables o un proceso de derivaciones al sistema. Entre estos métodos de transformaciones, para la linealización por realimentación, existen métodos como: Linealización Entrada-Salida, Forma de la Cadena, Flatness Diferencial, entre otros.

Para el caso de Linealización Entrada-Salida se aplica a un sistema no lineal de la forma:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{1.2}$$

Donde  $h: D \rightarrow \mathfrak{R}$  y  $f: D \rightarrow \mathfrak{R}$  sean diferenciable de orden suficiente en un subconjunto  $D \subset \mathfrak{R}^n$ . Se busca derivar la salida  $y$  hasta que aparezca la señal de control  $u$ , para esto se realiza el proceso (Rigatos,2015; Levine,2009):

$$y = \frac{\partial h}{\partial x} [f(x) + g(x)u] \stackrel{\text{def}}{=} L_f h(x) + L_g h(x) u \tag{1.3}$$

Dónde:

$$L_f h(x) = \frac{\partial h}{\partial x} f(x) \tag{1.4}$$

Es llamado *Lie Derivative* (Derivada Engañosa) de  $h$  con respecto a  $f$ . Teniendo así la derivada de orden  $k$  con respecto a  $h$  y  $f$  recursivamente:

$$L_f^k h(x) := L_f(L_f^{k-1} h(x)) \tag{1.5}$$

Se realiza este proceso hasta que se obtenga la señal de control  $u$ :

$$y^{(\rho)} = L_f^{(\rho)} h(x) + L_g L_f^{\rho-1} h(x)u \tag{1.6}$$

La ecuación 1.6 muestra claramente que el sistema es linealizable entrada-salida, si el control por realimentación es:

$$u = \frac{1}{L_g L_f^{\rho-1} h(x)} [-L_f^{(\rho)} h(x) + v] \quad (1.7)$$

Reduciendo el mapeo entrada-salida:

$$y^{(\rho)} = v \quad (1.8)$$

Siendo una cadena de  $\rho$  integraciones, siendo  $\rho$  el grado relativo del sistema.

Dentro del tema de métodos de transformación se encuentra la denominada Forma en Cadena o método de la cadena que trabaja en base a ecuaciones en cadena de las variables del sistema, la cual es considerada de la siguiente manera como un sistema de 2 entradas  $u = [u_1, u_2]$ :

$$\begin{cases} x_1 = u_1 \\ x_2 = u_1 x_3 \\ x_3 = u_1 x_4 \\ \vdots \\ \vdots \\ x_{n-1} = u_1 x_n \\ \{ x_n = u_2 \end{cases}$$

Donde el vector  $x = [x_1, x_2, x_3, \dots, x_n] \in R^n$  representan los estados del sistema en cadena. Con este método, se busca realizar una transformación del sistema a uno nuevo considerando las nuevas variables de estado y, de esta manera realizar, un control en base a estas nuevas variables de estado que serían definidas a un nuevo sistema dado como (Li, Xiao, Yang & Zhao, 2015):

$$z = u_1 A z + B u_2 \quad (1.9)$$

Donde  $z = [z_1, z_2, z_3, \dots, z_{n-1}] = [x_2, x_3, \dots, x_n] \in R^{n-1}$  es el estado del subsistema indicado anteriormente y tenemos que:

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ [0 & 0 & 0 & \dots & 0] \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

De esta manera es que el método de la cadena transforma un sistema a otro para controlarlo en base a las nuevas variables, y luego hacer la transformación inversa para obtener los valores del sistema real.

Un método similar es el flatness diferencial, el cual al igual que el método de la cadena, realiza un cambio de variables que lleva a un nuevo sistema el cual es fácilmente controlable. Este método, se basa en que teniendo un sistema como el de la ecuación (1.10):

$$x = f(x, u) \tag{1.10}$$

Donde dicho sistema dependiente del tiempo se elija una nueva variable  $y$  denominada salida flat, el cual debe satisfacer que todas las variables del sistema real puedan ser definidas en base a la salida flat que se eligen y sus derivas de igual manera. De esta manera, se consigue un sistema en base a las salidas flat que pueda ser controlable y, luego del control, realizar el proceso inverso para calcular los valores de mi sistema real. La selección de las salidas flat depende de la posibilidad de poder llegar a la señal de control en base a la nueva variable.

El método a utilizar es el de linealización por realimentación utilizando el flatness diferencial, ya que este método, que permite realizar un cambio de variables, es utilizado cuando se propone trabajar con trayectorias deseadas para ciertas variables de estado.

## **1.5. Objetivos de la Tesis**

### **1.5.1. Objetivos Generales**

Diseñar el sistema de control y navegación autónomo de un robot móvil mediante un controlador basado en linealización por realimentación para la conducción del móvil y que permita generar velocidad variable e integrar en una misma trayectoria la conducción en ambos sentidos.

### 1.5.2. Objetivos Específicos

Para poder alcanzar este objetivo es necesario realizar los siguientes trabajos de investigación y desarrollo:

- Obtener el modelo matemático del robot móvil.
- Diseñar una lógica de control basado en linealización por realimentación para que el robot pueda movilizarse correctamente por una trayectoria óptima para alcanzar su objetivo.
- Diseñar un sistema de control para determinar el ángulo del timón y velocidad del robot en cada instante.
- Diseñar un algoritmo de navegación para que el robot determine su posición dentro de su entorno de trabajo.
- Llevar el algoritmo de control y de navegación a un minicomputador para el control real de un robot móvil para verificar su implementación y viabilidad práctica.

## CAPÍTULO 2. ANÁLISIS DEL MODELO DEL ROBOT TIPO ACKERMAN

### 2.1. Modelo General del Robot Móvil

En el modelamiento de los robots móviles terrestres existen varias formas de calcular el modelo de cada vehículo dependiendo de su estructura de locomoción, pero que todos parten de un sistema básico. Principalmente, existen 2 tipos de modelo: Dinámico y Cinemático.

- **Modelo Cinemático:**

En este modelo, se parte del esquema básico del robot móvil, en el cual no se toma en cuenta efectos físicos que pueda tener el robot con el entorno ya sea el caso de fricción, torque, etc., y que este modelo se basa en el análisis del monociclo, obteniendo un modelo general de la forma:

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2.1)$$

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (2.2)$$

Donde  $v$  y  $\omega$  vienen a ser la velocidad lineal y velocidad angular respectivamente, siendo estas las entradas de control del sistema.

- **Modelo Dinámico:**

En este modelo, se toman en cuenta todas las reacciones físicas que puede tener el robot móvil con el entorno, como es: fricción, fuerza centrífuga, masa, etc., siendo así que se plantea un modelo como (Tang,2009):

$$Mq + Cq = E\tau - A^T\lambda \quad (2.3)$$

Dónde:



$$M = \begin{bmatrix} m_0 & 0 & -dm_0 \sin \theta \\ 0 & m_0 & dm_0 \cos \theta \\ -dm_0 \sin \theta & dm_0 \cos \theta & d^2m_0 + I_0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & -dm_0 \dot{\theta} \cos \theta \\ 0 & 0 & -dm_0 \dot{\theta} \sin \theta \\ 0 & 0 & 0 \end{bmatrix}$$

Donde  $M$  es la matriz de la masa,  $C$  es la matriz de Coriolis o de centrifuga,  $\tau$  es el torque de entrada y  $\lambda$  es la fuerza de restricción.

En aplicaciones comunes, se suele utilizar el modelo cinemático, el cual trabaja bien cuando se tienen velocidades bajas, mientras que el modelo dinámico se utiliza cuando ya se desea trabajar con mayores velocidades donde, principalmente, se produce el deslizamiento de las llantas dependiendo de la masa del robot y la velocidad con la que va.

## 2.2. Modelo Cinemático del Robot Móvil Tipo Ackerman

Para poder generar el modelo de un robot móvil primero se deben considerar las características a nivel estructural y funcionamiento del robot, así como también tener asunciones en las características que tendrá el robot móvil en movimiento.

Considerando las siguientes asunciones de funcionamiento:

- El robot móvil trabaja a baja velocidad (aproximadamente menor a 3 m/seg).
- Las ruedas del robot no deslizan ni resbalan, lo que lleva a que no existan temas de fricción durante el movimiento del vehículo y ahí este no se resbale o deslice.

Tomando estas asunciones, se presenta un modelo del robot móvil que funciona correctamente bajo estas asunciones, ya que, de ser el caso contrario, se requeriría un modelo dinámico, pues a mayor velocidad se genera fricción en los frenados o deslizamientos en los giros, lo que conlleva a tener un modelo más complejo.

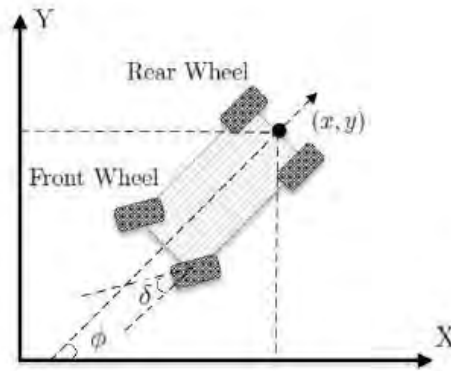


Figura 2.1: Diagrama del Robot Móvil Tipo Ackerman (Morán, Razuri, Sundgren & Rahmani,2013)

En la figura 2.1 se observa el diagrama de un robot móvil tipo Ackerman, donde se tienen las variables del sistema  $x, y$  que son las coordenadas del robot móvil ubicadas en la parte de tracción, que en este caso como se está trabajando con tracción posterior las coordenadas van ubicadas en la parte posterior,  $\theta$  que es el ángulo de inclinación del robot con respecto al eje  $x$ ,  $\delta$  es la inclinación de las ruedas de conducción o también se le conoce como el ángulo del timón,  $v$  que viene a ser la velocidad del vehículo.

Siendo su estructura un vehículo no holonómico, cumple la siguiente restricción:

$$x \sin \theta - y \cos \theta = 0 \quad (2.4)$$

El cálculo del modelo se basa en hacer el análisis en su forma simplificada como un modelo tipo bicicleta y tomando la referencia de su ICR, como se ve en la figura 2.2.

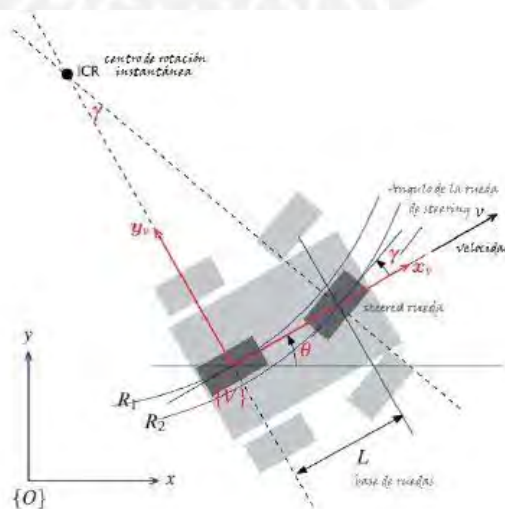


Figura 2.2: Modelo Basado en Tipo Bicicleta para un Tipo Carro (Corke,2011)

Mediante el cual facilita el cálculo de las ecuaciones del modelo, obteniéndose:  
(Corke,2011)

$$\begin{aligned}x &= v \cos(\theta) \\y &= v \sin(\theta) \\ \theta &= \frac{v}{L} \tan(\delta)\end{aligned}\tag{2.5}$$

La ubicación  $x$ ,  $y$  del vehículo representa la posición actual que se encuentra, siendo el punto central entre las ruedas de tracción. La orientación del vehículo es medida con respecto al eje horizontal. El vector de estado está definido por:

$$\underline{x} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}\tag{2.6}$$

Las variables de control son la velocidad del vehículo y el ángulo de dirección de las ruedas, cuyo ángulo es medido con respecto al ángulo de orientación del vehículo.

$$\underline{u} = \begin{bmatrix} v \\ \delta \end{bmatrix}, \quad \underline{u} \in U = \begin{cases} v: & v_{min} \leq v(t) \leq v_{max} \\ \delta: & \delta_{min} \leq \delta(t) \leq \delta_{max} \end{cases}\tag{2.7}$$

Donde los valores máximos y mínimos de las variables de control dependen de la estructura del robot móvil y en base a ciertas restricciones que se puedan dar para cada variable.

El modelo indicado en la ecuación 2.5, es un modelo válido para las asunciones dadas anteriormente, y es el que se utiliza para el control del robot móvil. De manera que se han identificado cuáles son las variables de salida del sistema y cuáles vendrían a ser sus entradas, y así con ese conocimiento se podrá diseñar el sistema de control respectivo.

### 2.3. Modelo Cinemático del Robot Móvil Tipo Diferencial

Como se mencionó los modelos de los robots móviles terrestres parten de un modelo básico presentado en la ecuación (2.1), y que difieren en base a la locomoción de cada uno. En el caso de un robot móvil tipo diferencial, mostrado en la figura 2.3.

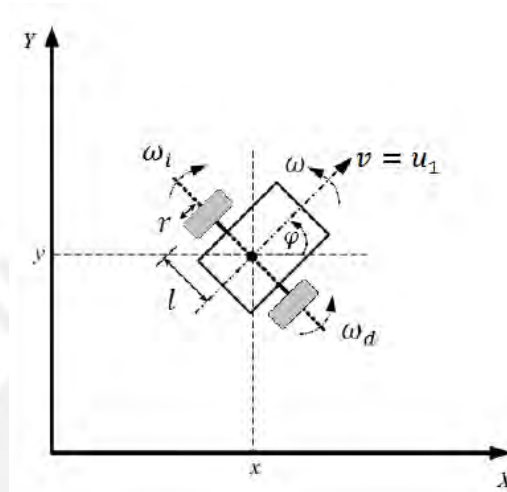


Figura 2.3: Robot Móvil Tipo Diferencial (Grisales,2013)

El cual trabaja con el modelo básico, pero sus señales de control están regidas por la velocidad de giro de las ruedas, teniendo así la siguiente transformación:

$$\begin{bmatrix} \theta_D \\ \theta_I \end{bmatrix} = \frac{1}{d} \begin{bmatrix} 1 & L \\ 1 & -L \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2.8)$$

Entre todos los modelos de robot móvil terrestre que hay, el modelo diferencial y el modelo tipo carro son los más utilizados, lo cual ha conllevado a decidir cuál de ambos tipos usar dependiendo del uso.

### 2.4. Esquema General para el Control y Navegación

Habiendo definido las variables de estado  $(x, y, \theta)$  y las variables de control  $(v, \delta)$  del sistema, se considera la estructura presentada en la figura 2.4 como esquema para el control y navegación del robot móvil tipo Ackerman.

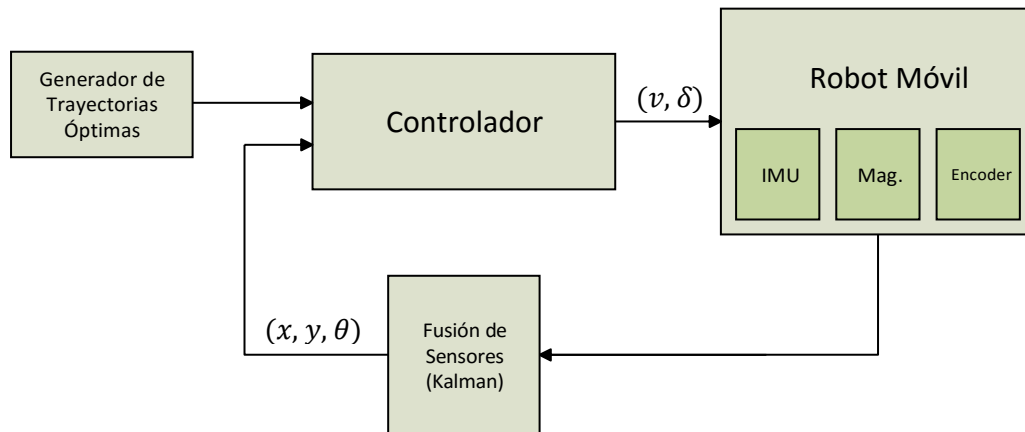


Figura 2.4: Esquema de Control y de Navegación Propuesto

En primer lugar, se realizará la generación de la trayectoria óptima deseada basada en un algoritmo que analice las especificaciones y restricciones que se dan para que el robot llegue a su destino. Para conocer la ubicación del robot móvil dentro de su espacio de trabajo, se toman las medidas de ciertos sensores como los inerciales, magnetómetro y el encoder. Estos sensores permitirán estimar la posición del vehículo y su ángulo de orientación mediante la fusión de sensores, basado en filtro de Kalman, para poder aumentar la precisión de las mediciones y disminuir el efecto del ruido de medición en los sensores.

Así el controlador, el cual está diseñado mediante control óptimo dentro de un esquema de linealización por realimentación, generará los valores de las variables de control correspondiente partiendo de las mediciones tomadas en cada instante de tiempo sobre la posición y orientación del vehículo, permitiendo que siga la trayectoria propuesta correctamente.

## CAPÍTULO 3. PLANEAMIENTO DE TRAYECTORIA DESEADA

### 3.1. Diseño de Ruta Óptima

#### 3.1.1. Introducción

En el planeamiento de la trayectoria a seguir por un vehículo móvil es una de las cosas principales en el tema de conducción autónoma de los robots móviles, siendo así que se han desarrollado algunas técnicas de planeamiento de trayectoria para que el vehículo móvil alcance su objetivo. Hablar de ruta óptima es hablar de una trayectoria que permita al vehículo recorrer la menor distancia posible, así como también en el menor tiempo posible, lo cual sucede al trabajar con la velocidad. Es así como se plantea primero el análisis que debe tener la selección de la ruta mínima en base a las condiciones iniciales y finales del vehículo, y como debe conformarse la trayectoria a seguir por el vehículo describiendo la menor ruta posible considerando el máximo desempeño realizable.

El diseño de ruta óptima depende de las señales de control disponible ya sea el caso de la velocidad de giro de las ruedas en un robot móvil tipo diferencial o velocidad del móvil y ángulo del timón para el caso de uno tipo carro. Por ejemplo, la manera más rápida de llegar a su objetivo, en el caso del tipo diferencial, es girar el móvil hasta que este frente al objetivo final y direccionarlo a ese punto, mientras que para el caso del robot móvil tipo carro se plantea otro análisis.

Dentro de las investigaciones en este tema se encuentra el realizado por Dubins (Furtuna & Balkcom, 2010; Gao, Hou, Zhu, Zhang & Chen, 2013) resuelto en 1957, demostrando que un robot móvil tipo carro puede llegar de su posición inicial a una posición deseada con exactamente 3 segmentos que consisten en curvas o arcos de circunferencia de radio mínimo posible y líneas rectas. En el modelo de Dubins se considera que el robot móvil se trasladara en un solo sentido (velocidad positiva o avance), existiendo únicamente 3 tipos de direcciones como se muestra en la figura 3.1.

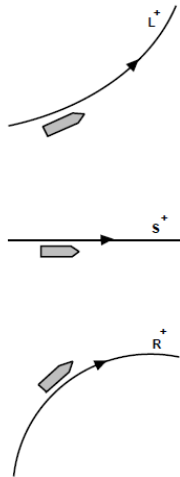


Figura 3.1: Tipos de Direcciones de Dubins (Sussmann & Tang, 1991)

En la figura 3.1 se muestran 3 ejemplos de trayectorias de Dubins, describiendo movimientos hacia la derecha (R - right), izquierda (L - left) y recto (S - straight), y se observa también que solo se realiza movimiento en avance (velocidad positiva).

Luego Reeds y Shepp resuelve un problema similar pero ahora considerando que el vehículo puede moverse en adelante y reversa utilizando de igual manera solamente las condiciones iniciales y finales del vehículo. A diferencia del caso de Dubins que solo existen 3 tipos de dirección, en Reeds y Shepp se tienen 6 tipos de dirección debido a la conducción en doble sentido como se muestra en la figura 3.2.

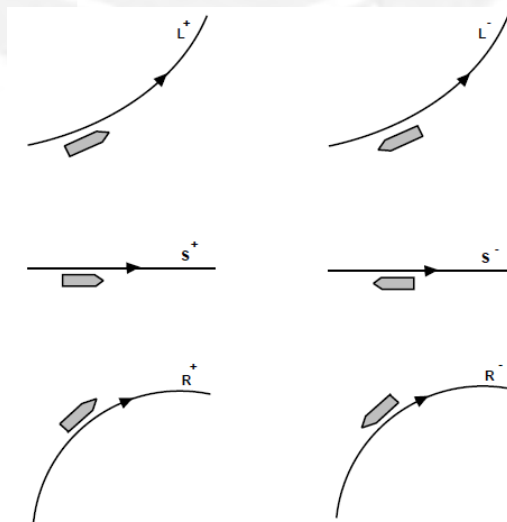


Figura 3.2: Tipos de Direcciones de Reeds y Shepp (Straight), R(Right) (Sussmann & Tang, 1991)

El uso de trayectoria de Dubins funciona cuando se realiza un control del vehículo con velocidad constante en un solo sentido durante toda la trayectoria, aplicable en algunos escenarios, pero para un caso general donde el vehículo pueda ir en ambos sentidos y la velocidad entra como una señal de control, la trayectoria Reeds y Shepp aplica. En Dubins se aplican métodos matemáticos usando las condiciones iniciales y finales del vehículo, además, se tienen formas ya establecidas de trayectorias, entonces en las aplicaciones simplemente se busca averiguar cuál de las posibles trayectorias aplica (tema que no se analizará en esta investigación). Sin embargo, en el caso de Reeds y Shepp existen varios ejemplos de trayectorias que se pueden generar, por lo que lo hace más difícil llevar a cabo un método principal.

En esta investigación se ha diseñado un algoritmo que permita generar todas las posibles trayectorias utilizando el análisis de Reeds y Shepp, y luego analizar desde la mínima ruta posible si se logra llegar al destino en la orientación correcta.

Para observar y analizar cómo se realiza el procedimiento para obtener la trayectoria óptima, se va a iniciar definiendo como ejemplo una posición inicial y final de un vehículo móvil, como se muestra en la figura 3.3, para así analizar los datos a considerar dentro de una mínima trayectoria y qué datos son los necesarios obtener al final del análisis.

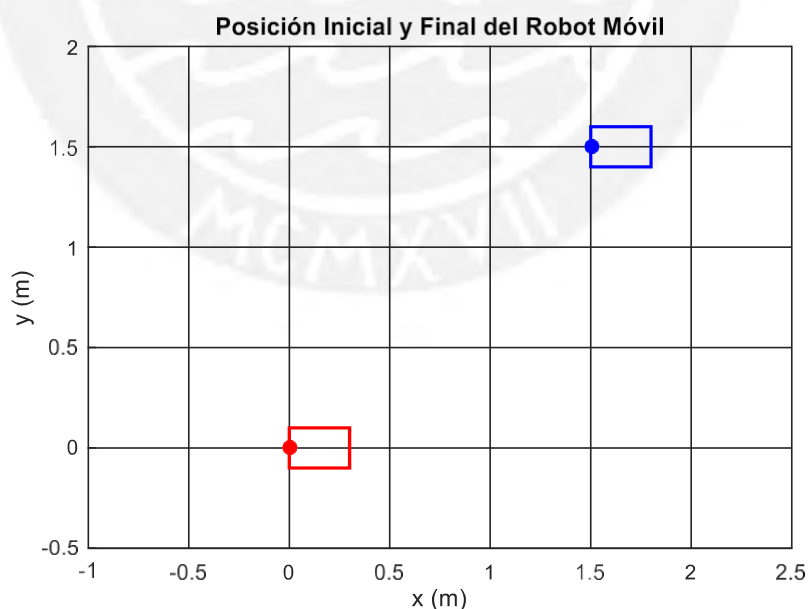


Figura 3.3: Posición Inicial (rojo) y Final (Azul) del Robot Móvil



Se inicia tomando en cuenta el CIR (Centro Instantáneo de Rotación), con lo cual se sabe que, al llevar el timón a su máximo valor, el robot se mueve formando una trayectoria circular con un radio  $R$  definido en la ecuación 1.1, que es el mínimo radio posible. Entonces, la idea es que el robot móvil gire por un arco de circunferencia, formado por su CIR, hasta llegar a un punto que permita generar una recta tangente a su círculo inicial y a la vez con el círculo final, el cual sería el CIR de la posición final.

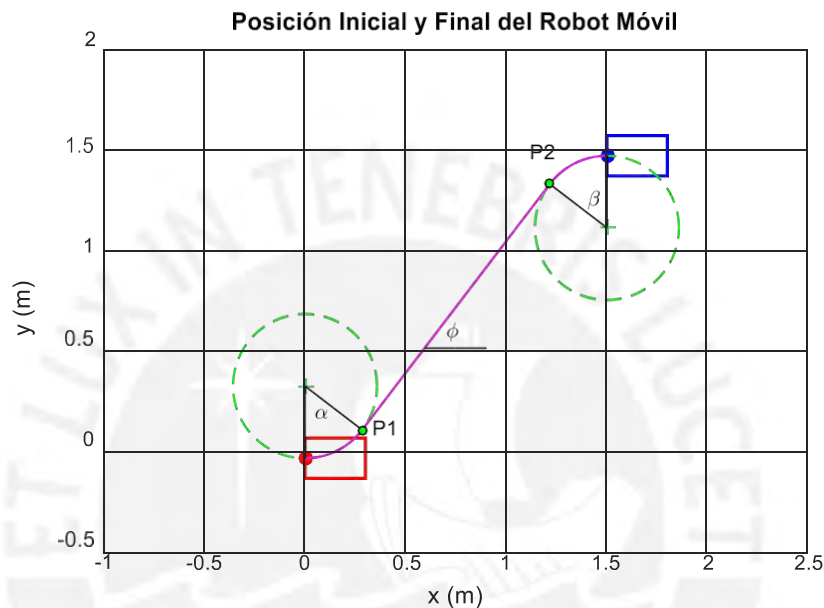


Figura 3.4: Posible Mínima Trayectoria a Calcular

Prácticamente, la trayectoria mostrada en la figura 3.4 es la mínima posible por la cual el robot móvil debería trasladarse para llegar a su punto final, la cual ha requerido el uso de 3 segmentos que serían los arcos en cada circunferencia (inicio y final) y la línea recta, siendo  $\alpha$  y  $\beta$  los ángulos de giro que realiza el robot en los arcos,  $P_1$  y  $P_2$  los puntos de unión entre las líneas rectas y las circunferencia inicial y final respectivamente, y  $\phi$  el ángulo de la línea recta. Puede existir el caso también donde la trayectoria mínima requiera de 2 segmentos o hasta de 1 solamente. El algoritmo debe ser capaz de generar dicha trayectoria, dando como resultado la dirección que debe tener el vehículo en cada segmento, así como también los puntos que divide cada segmento. Se deduce que la mínima trayectoria será generada considerando únicamente 3 estados del ángulo del timón:

$$\delta = \begin{cases} \delta_{max} \\ 0 \\ -\delta_{max} \end{cases} \quad (3.1)$$

Ya que, llevando el ángulo del timón a sus extremos, se obtiene el mayor desempeño posible que puede generar el robot móvil y de esta manera forzar a transcurrir por la mínima trayectoria posible.

### 3.1.2. Cálculo de Líneas Rectas y Arcos de Círculos para Ruta Mínima

El algoritmo de simulación que permite realizar el análisis de la mínima trayectoria se encuentra en el anexo A.1, donde el concepto parte en que el vehículo, tanto en su posición inicial como final, puede generar giros en ambos sentidos con un arco de radio mínimo generando así 2 circunferencias en cada posición como se muestra en la figura 3.5.

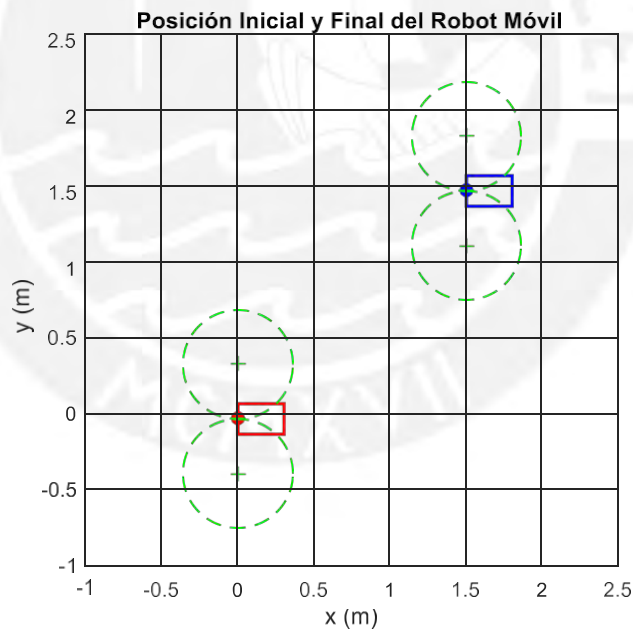


Figura 3.5: Circunferencias Formada en la Posición Inicial y la Posición Final

Primero se debe trazar las circunferencias que puede generar el móvil en su punto inicial y final, las cuales serían 2 en cada posición considerando el giro a la derecha y la izquierda. Luego, se deben trazar todas las rectas tangentes posibles entre los

círculos de la posición inicial y final, considerando que existen rectas tangentes internas y rectas tangentes externas como se observa en la figura 3.6:

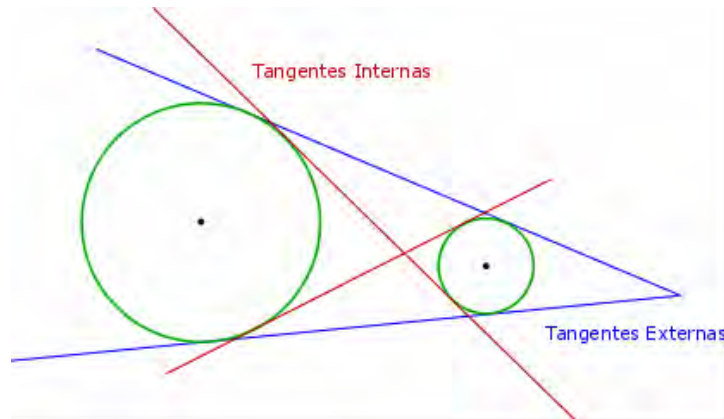


Figura 3.6: Tipos de Rectas Tangentes entre Circunferencias

El trazo de rectas tangentes es importante ya que estas representan la unión de las circunferencias, y los extremos de la recta representa los límites en cada circunferencia hasta el cual se requiere que el vehículo gire en su máximo desempeño, para luego, recorrer dicha recta hasta la circunferencia formada por la posición final. Esto lleva a poder generar 16 rectas tangentes en total como se puede apreciar en la figura 3.7:

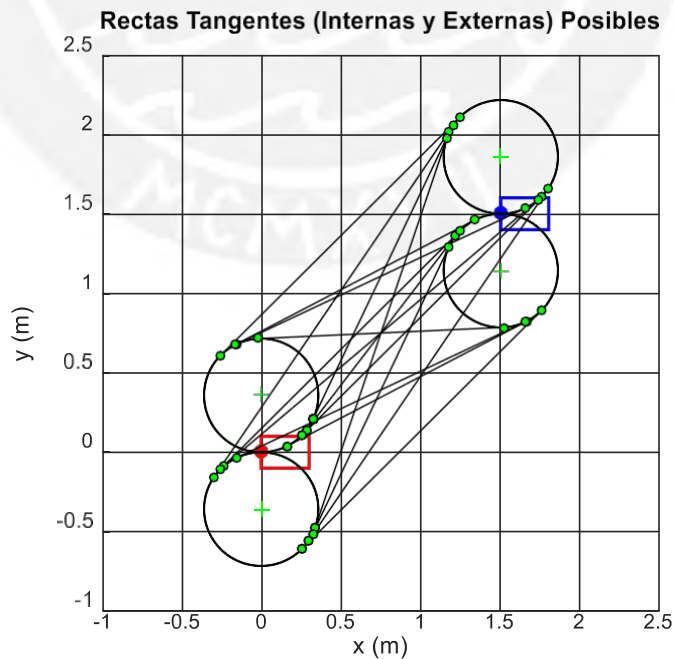


Figura 3.7: Generación de Todas las Rectas Tangentes Posibles

Utilizando las funciones en el anexo A.2, los cuales permiten generar las rectas tangentes internas y externas, se observan 16 rectas tangentes generadas en base a la interacción entre las circunferencias de la posición inicial con las de la posición final, mediante las cuales se consigue obtener 64 posibles rutas que puede tomar el vehículo para llegar a la posición final considerando únicamente las coordenadas  $(x, y)$  y considerando movimiento en ambos sentidos. Dentro de las 64 posibles rutas se considera que el robot puede llegar al punto tangente entre la circunferencia y la línea recta por 2 rutas:

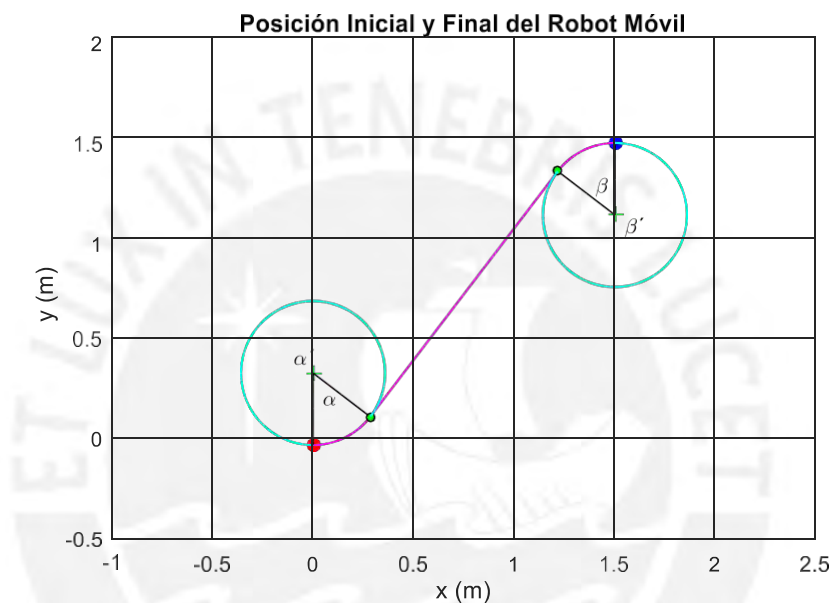


Figura 3.8: Doble Camino de Movimiento Posible en los Círculos

Como se observa en la figura 3.8, en el círculo formado por cada posición inicial y final se observa que existen 2 rutas de movimiento posible, siendo  $\alpha$  y  $\alpha'$  para la posición inicial, y  $\beta$  y  $\beta'$  para la posición final, donde ambos ángulos en cada caso son rutas que unen el punto tangente con la posición inicial y final en cada caso, pero diferentes en magnitud ya que:  $\alpha' > \alpha$  y  $\beta' > \beta$ . Se debe de analizar cuál de las 2 rutas se deben tomar, siendo la menor de estas la que se considera como el arco a seguir que lleve a la ruta optima, lo cual disminuye las posibles rutas desde el punto inicial al punto final a un total de 16 rutas posibles.

Utilizando la ecuación 3.2 que permite calcular el ángulo formado por una cuerda, donde dicha cuerda viene a ser la recta entre el punto inicial o final y la recta

tangente que corresponda, este mismo ángulo viene a ser el ángulo de giro en la circunferencia de justamente menor valor de las 2 posibles rutas de gira.

$$\gamma = \cos^{-1}\left(1 - \frac{C^2}{2R^2}\right) \quad (3.2)$$

Donde  $\gamma$  es el ángulo formado por los puntos de posición inicial o final y su punto tangente correspondiente,  $C$  es la longitud entre la posición y el punto tangente y  $R$  el radio de la circunferencia. Esta ecuación permite calcular el menor ángulo siendo normalmente  $\gamma \leq \pi$ .

Para realizar el análisis de la obtención de la trayectoria optima deseada, el algoritmo contiene los siguientes procedimientos principales:

- Generar las rectas tangentes internas y externas posibles entre las 4 circunferencias.
- Calcular todos los ángulos posibles de giro de los arcos en cada circunferencia
- Calcular la distancia de todas las trayectorias posibles y ordenarlas desde la menor distancia hasta la mayor.
- Evaluar en cada posible trayectoria que el robot móvil llegue a su punto final en el ángulo de orientación correcto.
- Almacenar los puntos tangentes, el ángulo de orientación de la recta tangente, el sentido de la velocidad y el ángulo del timón en cada segmento para realizar el diseño de la trayectoria deseada.

Cumpliendo los procedimientos indicados, el algoritmo permite calcular satisfactoriamente la ruta óptima del punto inicial al punto final como se muestra en la figura 3.9.

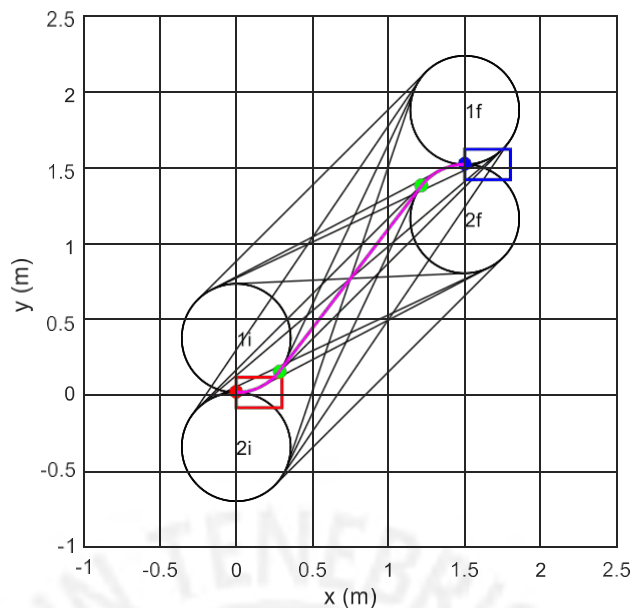


Figura 3.9: Trayectoria Mínima Obtenida de Todas las Trayectorias Posibles

Se muestra en línea morada la trayectoria óptima obtenida y la de color verde los puntos tangentes, entre la línea recta y las circunferencias formadas por la posición inicial y final, los cuales son puntos de referencia para realizar la generación del vector de puntos de la trayectoria, lo cual se mencionará más adelante.

Hay ocasiones donde la trayectoria resultante no es la única solución, ya que puede suscitarse que para la ruta de un punto inicial a un punto final existan 2 trayectorias óptimas posibles como se muestra en la figura 3.10.

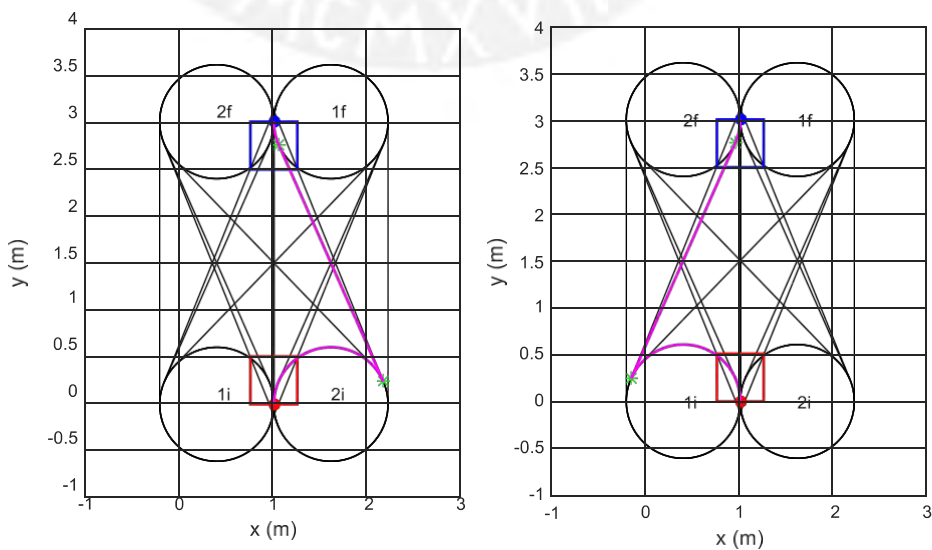


Figura 3.10: Comparación de una Situación con 2 Trayectorias Óptimas

Esto indica que no se tiene un solo resultado mínimo, por lo que sería un mínimo local y no un mínimo global.

A continuación, se muestra en la figura 3.11, la comparación de las trayectorias mínimas cuando se considera movimiento solo en avance y cuando se considera movimiento en avance y retroceso en simultaneo, considerando diferentes posiciones iniciales (rojo) y diferentes posiciones finales (azul), y así observar claramente la diferencia de trayectoria que se puede generar en ambos casos.

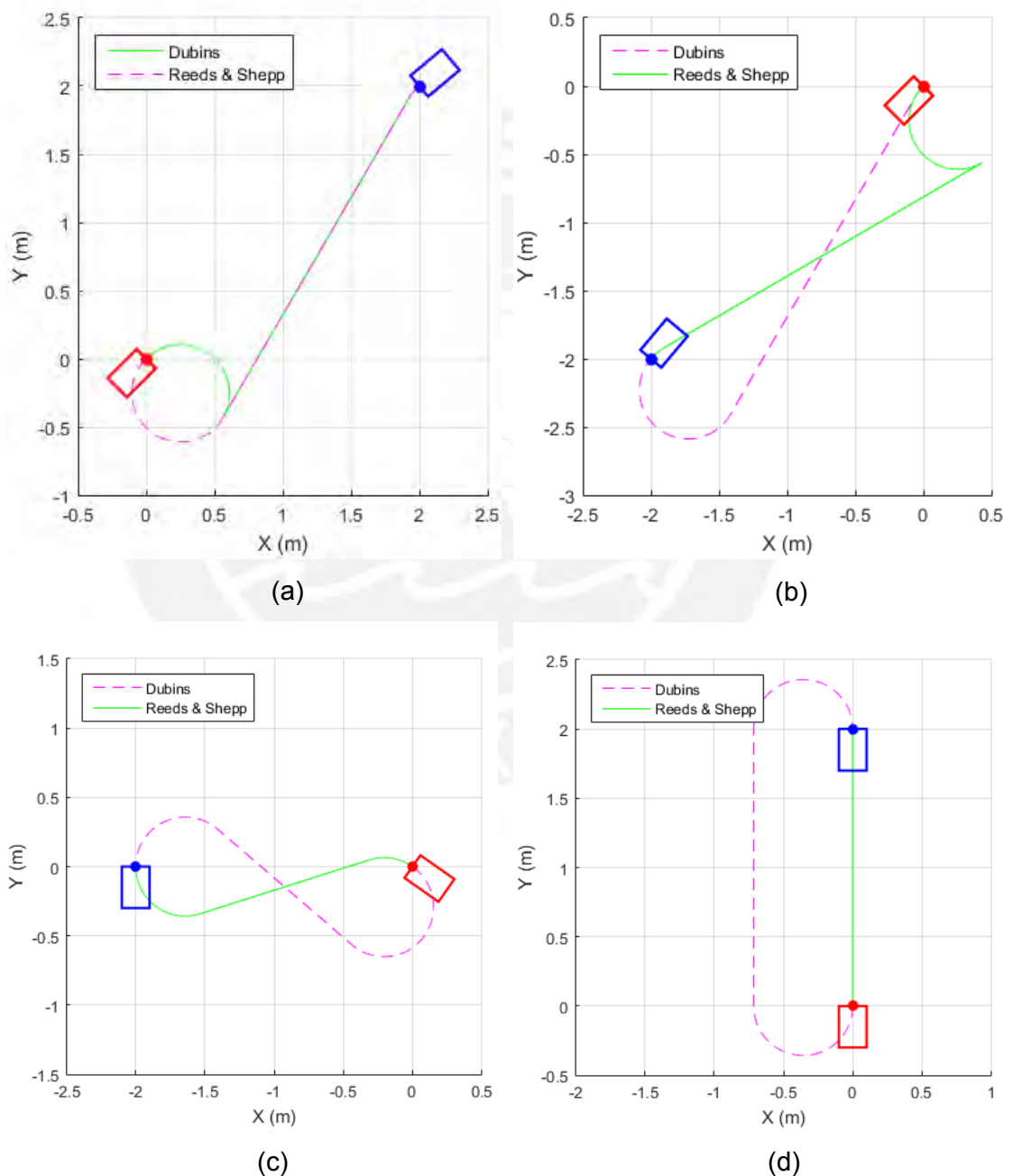


Figura 3.11: Comparaciones entre Trayectoria con Velocidad en un Sentido y con Ambos Sentidos

La generación de una trayectoria, considerando el uso de velocidad en ambos sentidos, permite generar la mínima trayectoria posible por la que puede moverse un vehículo a diferencia de cuando solo se considera velocidad en avance o en retroceso únicamente, donde se puede apreciar la diferencia en las distancias de cada trayectoria como se muestra en la tabla 3-1.

	Dubins	Reeds & Shepp
Figura 3.11a	4.0439 m	3.8587 m
Figura 3.11b	3.9808 m	3.8587 m
Figura 3.11c	3.2551 m	2.2812 m
Figura 3.11d	4.2464 m	2 m

Tabla 3-1: Distancia de Recorrido Utilizando la Teoría de Dubins o Reeds & Shepp

Este análisis realizado permite conocer la forma de la trayectoria y los puntos por los cuales pasara la trayectoria en cada segmento, y también conocer los sentidos de giro que tendrá el robot móvil, pero para poder generar correctamente el vector de puntos que permitan generar una trayectoria que considere la velocidad y el ángulo de dirección se plantean 2 técnicas de generación, una para el caso de velocidad variable y otras para el caso de tiempo mínimo

## 3.2. Generación de Trayectoria para Velocidad Variable

### 3.2.1. Función Polinómica

El planeamiento de trayectoria se basa en encontrar una relación entre tiempo y espacio del sistema al que se desea aplicar, para lo cual se plantea en este caso el uso de una función polinómica que permita generación el vector de puntos deseados desde la posición inicial hasta la posición final describiendo velocidad variable. Consecuentemente, la trayectoria es expresada como una función paramétrica que depende del tiempo, el cual provee en cada instante de tiempo el valor deseado de la posición (Biagiotti & Melchiorri, 2008).



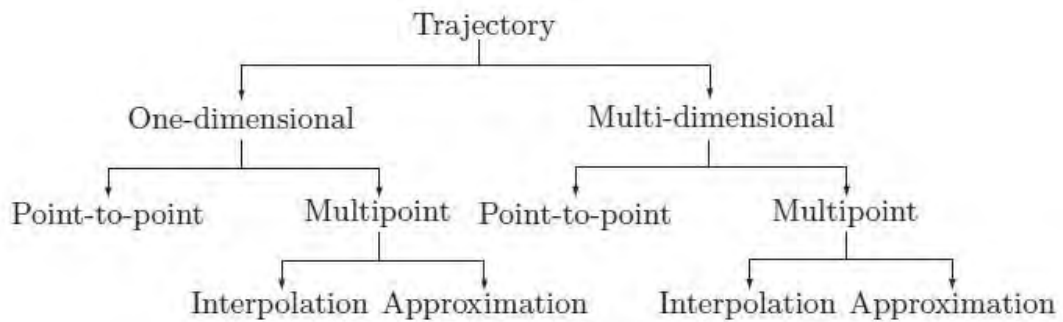


Figura 3.12: Categorías de Trayectoria (Biagiotti & Melchiorri,2008)

En la figura 3.12, se muestra las diferentes categorías que existen en el planeamiento de trayectoria, donde difieren en que sea uni-dimensional o multi-dimensional. En el primer caso (que es el utilizado en este proyecto), se define la posición para un sistema de un grado de libertad, mientras que para el segundo caso se considera un sistema multidimensional. Consecuente de la dimensión del sistema de trabajo, hay 2 divisiones que son trayectoria punto a punto o multipunto: en el primer caso, se definen únicamente el valor inicial y valor final de la trayectoria, mientras que, para el segundo caso, se define un vector de puntos por los cuales se desea que la posición pase (Biagiotti & Melchiorri, 2008).

Dentro del planeamiento de trayectoria, que se definió como una función paramétrica dependiente del tiempo, se tienen 3 tipos de diseño de esta función, siendo estas: Polinómica, Trigonométrica y Exponencial; donde en este proyecto se trabajara con una función polinómica.

Se busca definir una función polinómica con respecto al tiempo:

$$q = q(t) \quad t \in [t_i, t_f]$$

Que satisfaga las condiciones dadas de tiempo, posición, velocidad y aceleración determinados en un tiempo inicial  $t_i$  y tiempo final  $t_f$ . En este problema se considera la función como un polinomio de la forma:

$$q(t) = a_0 + a_1t + a_2t^2 + \dots + a_nt^n \quad (3.3)$$

Donde los coeficientes  $a_k$ ,  $k = 1, 2, \dots, n$ , se determinan en base a cumplir las condiciones dadas de trayectoria, y donde el grado del polinomio  $n$  depende de cumplir las condiciones de cómo se desea que se den los cambios de las condiciones dadas en el tiempo, ya sean cambios suaves o cambios bruscos.

Debido a que la cantidad de condiciones límites suele ser par, el valor del grado de la función polinómica es impar, i.e. uno, tres, cinco, siete, etc (Biagiotti & Melchiorri, 2008).

Esta forma paramétrica dependiente del tiempo es muy útil ya que permite generar un vector de puntos partiendo de la señal de tiempo definida como un vector de puntos separados un  $\Delta t$  con punto inicial iguala  $t_i$  y punto final  $t_f$ , de tal manera que  $q(t)$  se pueda tomar como un vector de puntos en cada instante de tiempo.

El grado del polinomio refleja cómo se desea que varíe la trayectoria de la posición y sus respectivas derivadas que vendría a ser: velocidad y aceleración. En el caso del control del vehículo móvil, es claro que se trabaja con velocidad variable, por lo que se requiere que el polinomio cuente con orden de 3 a más, ya que, por ejemplo, al tener una función polinómica de grado 1, se tendría:

$$q(t) = a_0 + a_1 t$$

Y derivando para obtener velocidad y aceleración:

$$\begin{aligned} \dot{q}(t) &= a_1 \\ \ddot{q}(t) &= 0 \end{aligned}$$

Lo que muestra que se tendría una trayectoria que genera velocidad constante y que, para este trabajo, en el cual se enfatiza el diseño con variación de velocidad, no sería de utilidad y es por eso que se puede plantear una función polinómica de orden 3 a más, que debe ser elegida de acuerdo a la trayectoria que se quiere generar y para cumplir las condiciones límites.

### 3.2.2. Cálculo de los Parámetros del Polinomio

Para el cálculo de los parámetros de la función polinomial base, definida por la ecuación (3.3), se requiere plantear la función polinómica de orden “n” donde su k-esima derivada con respecto al tiempo tenga un valor en un tiempo específico, la cual matemáticamente se puede dar la siguiente manera (Biagiotti & Melchiorri, 2008):

$$q^{(k)}(t) = k! a_k + (k+1)! a_{k+1} t + \dots + \frac{n!}{(n-k)!} a_n t^{n-k} \quad (3.4)$$

Lo cual nos genera todas las formas polinómicas para cada derivada y que se puede definir en su forma matricial como:

$$Ma = b \quad (3.5)$$

Donde  $M$  es una matriz de  $(n+1) \times (n+1)$  no singular,  $b$  está compuesto por las  $(n+1)$  condiciones a satisfacer y  $a = [a_0, a_1, \dots, a_n]^T$  es el vector de los parámetros a calcular, por lo que, despejando el vector de parámetros de la ecuación, se obtiene:

$$a = M^{-1}b \quad (3.6)$$

Por ejemplo, se requiere calcular los parámetros del polinomio para satisfacer 2 condiciones: posición y velocidad, para lo cual se requiere un polinomio de orden 3 como mínimo (más adelante se explicará la selección del grado del polinomio a utilizar) y definiendo las condiciones iniciales y finales:

- Posición y velocidad inicial:  $x_i, v_i$
- Posición y velocidad final:  $x_f, v_f$

Por lo que se generan los siguientes polinomios de acuerdo a las condiciones a satisfacer:

$$\begin{aligned} x_i &= q(t_i) = a_0 + a_1 t_i + a_2 t_i^2 + a_3 t_i^3 \\ v_i &= q'(t_i) = a_1 + 2a_2 t_i + 3a_3 t_i^2 \\ x_f &= q(t_f) = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 \\ v_f &= q'(t_f) = a_1 + 2a_2 t_f + 3a_3 t_f^2 \end{aligned} \quad (3.7)$$

Con lo cual se arma la matriz  $M$  y los vectores  $a$  y  $b$  de la siguiente forma:

$$M = \begin{bmatrix} 1 & t_i & t_i^2 & t_i^3 \\ 0 & 1 & 2t_i & 3t_i^2 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \quad a = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \quad b = \begin{bmatrix} x_i \\ v_i \\ x_f \\ v_f \end{bmatrix}$$

Y con la ecuación matricial en 3.12, se calculan los parámetros del polinomio.

### 3.2.3. Selección del Grado del Polinomio

A continuación, se realizará las pruebas para cada orden del polinomio sugerido, para identificar las diferencias y elegir la mejor opción para la trayectoria del robot móvil. Para las pruebas se utilizará los siguientes valores iniciales y finales de trayectoria en un tiempo [0 - 10]:

$$\begin{aligned} x_i &= 2 & v_i &= 0 & a_i &= 0 \\ x_f &= 5 & v_f &= 0 & a_f &= 0 \end{aligned}$$

Y que en la función polinómica se representará la trayectoria de posición como  $q(t)$ , la trayectoria de velocidad como  $\dot{q}(t)$  y la trayectoria de aceleración como  $\ddot{q}(t)$ .

Para el caso de un polinomio de orden 3, se tiene:

$$\begin{aligned} q(t) &= a_0 + a_1t + a_2t^2 + a_3t^3 \\ \dot{q}(t) &= a_1 + 2a_2t + 3a_3t^2 \\ \ddot{q}(t) &= 2a_2 + 6a_3t \end{aligned} \tag{3.8}$$

Como se observa, la función para la aceleración representará una recta con pendiente igual a  $6a_3$ , y que el valor de la aceleración sería constante si la pendiente fuera cero, con lo cual la aceleración sería igual a  $2a_2$ , pero que son valores que no se conocen de antemano, ya que depende de los parámetros a calcular y sucede que no coincide con los valores límites que se colocarían para la aceleración y no se obtendría la trayectoria que se desea. Es por esto que para el cálculo de los parámetros se utilizan únicamente las funciones polinómicas de la posición y velocidad.

$$M = \begin{bmatrix} 1 & t_i & t_i^2 & t_i^3 \\ 0 & 1 & 2t_i & 3t_i^2 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix}, \quad b = \begin{bmatrix} x_i \\ v_i \\ x_f \\ v_f \end{bmatrix}$$

Obteniendo:

$$a_0 = 2 \quad a_1 = 0 \quad a_2 = 0.5625 \quad a_3 = -0.0938$$

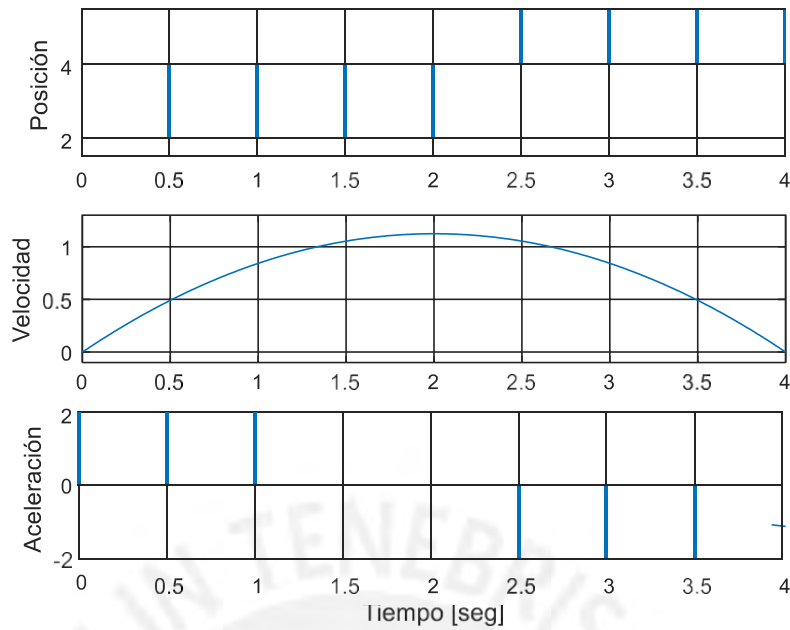


Figura 3.13: Trayectoria para la Posición, Velocidad y Aceleración con Polinomio de Orden 3

Para el caso de un polinomio de orden 5, se tiene:

$$\begin{aligned}
 q(t) &= a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \\
 \dot{q}(t) &= a_1 + 2a_2t + 3a_3t^2 + 4a_4t^3 + 5a_5t^4 \\
 \ddot{q}(t) &= 2a_2 + 6a_3t + 12a_4t^2 + 20a_5t^3
 \end{aligned} \tag{3.9}$$

Ahora se observa que la función para la aceleración tendrá una forma parabólica a través del tiempo, con lo cual si se puede definir con mayor seguridad los valores límites de la aceleración y que coincidan con la señal formada al calcular los parámetros. Es por esto que ahora se puede integrar también la función polinómica de la aceleración para el cálculo de los parámetros

$$M = \begin{bmatrix} 1 & t_i & t_i^2 & t_i^3 & t_i^4 & t_i^5 \\ 0 & 1 & 2t_i & 3t_i^2 & 4t_i^3 & 5t_i^4 \\ 0 & 0 & 2 & 6t_i & 12t_i^2 & 20t_i^3 \\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 & t_f^5 \\ 0 & 1 & 2t_f & 3t_f^2 & 4t_f^3 & 5t_f^4 \\ 0 & 0 & 2 & 6t_f & 12t_f^2 & 20t_f^3 \end{bmatrix}, \quad b = \begin{bmatrix} x_i \\ v_i \\ a_i \\ x_f \\ v_f \\ a_f \end{bmatrix}$$

Obteniendo:

$$a_0 = 2 \quad a_1 = 0 \quad a_2 = 0 \quad a_3 = 0.4688 \quad a_4 = -0.1758 \quad a_5 = 0.0176$$

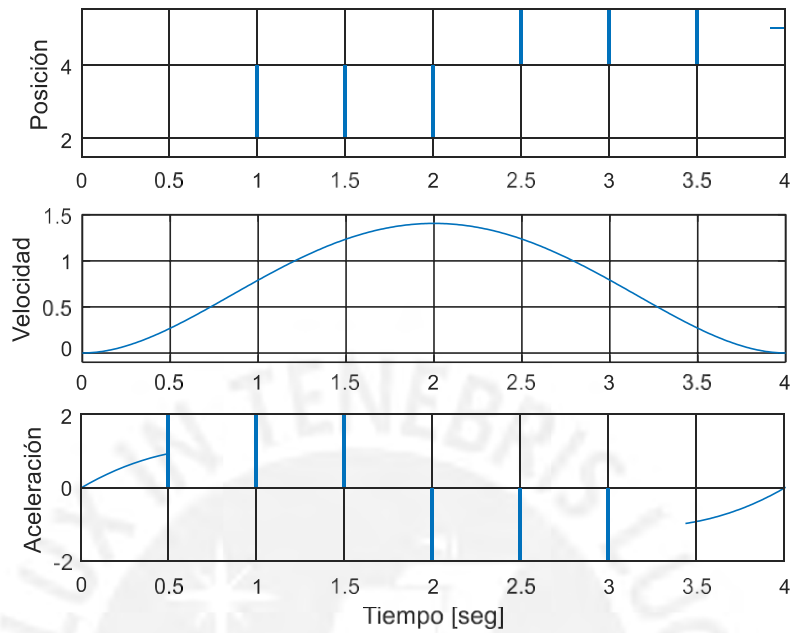


Figura 3.14: Trayectoria para la Posición, Velocidad y Aceleración con Polinomio de Orden 5

Comparando las respuestas de los polinomios de tercer y quinto orden para las mismas condiciones límites:

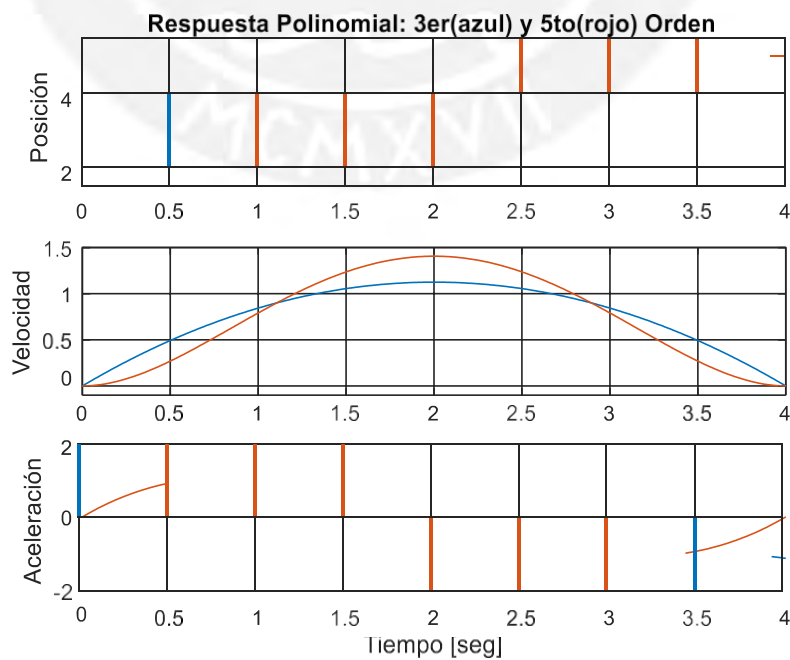


Figura 3.15: Diferencia de la Respuesta Polinomial entre el 3er y 5to Orden

Se observa que en la posición no se ve mucha diferencia, a comparación de las señales de velocidad y aceleración, pues en el caso de orden 3 se tiene que inicia con una aceleración diferente a cero y su velocidad al inicio no es tan suave a comparación de la velocidad con el orden 5, ya que en este al inicio su velocidad incrementa suavemente y esto debido a que la aceleración inicia con el valor de cero y oscila para ajustar la variación de la velocidad. Es por esta última diferencia que se tiene mejor respuesta para el caso del robot móvil con un polinomio de orden 5, ya que en la situación real un vehículo tiene una marcha lenta al inicio y al final que serían las situaciones en el arranque y en la parada.

Se demuestra también que el orden recomendable del polinomio depende de la cantidad de condiciones límites a satisfacer, por ejemplo, para conocer la respuesta de posición, velocidad y aceleración (3 condiciones a satisfacer), como se vio anteriormente, se requiere un polinomio de orden 5 como mínimo, en el cual se tenían 6 valores límites, lo que lleva a representar de manera matemática el polinomio base de orden mínimo como:

$$q(t) = a_0 + a_1t + a_2t^2 + \dots + a_{m-2}t^{m-2} + a_{m-1}t^{m-1} \quad (3.10)$$

Dónde:

$$m = \#valores\ limites - 1 \quad (3.11)$$

Se observa que la generación de trayectoria en base a polinomios ayuda a generar una buena ruta de la posición con solo introducir las coordenadas límites de posición, dando una buena señal de igual manera para la velocidad, que justamente tiene una respuesta como se desea; es decir, velocidad variante en el tiempo dependiendo de la posición del móvil en el transcurso de la trayectoria, ya que como se observa en la figura 3.14, la velocidad aumenta en la parte intermedia de la trayectoria y permite coincidir con el tiempo de llegada a la posición final.

#### **3.2.4. Generación del Vector de Puntos Utilizando Función Polinómica**

Ahora, se puede realizar la generación del vector de puntos de la trayectoria deseada utilizando el método de la función polinómica que permitirá obtener la data en cada instante de tiempo de la trayectoria. Como se mencionó anteriormente, se

plantea utilizar un polinomio de orden 5 debido a su mejor desempeño en la generación de curvas en comparación con la de orden 3, y también porque permite definir las condiciones iniciales y finales correctas hasta la segunda derivada de la posición que viene a ser la aceleración.

La generación del vector de puntos de la trayectoria deseada se realiza aplicando la función polinómica a cada segmento utilizando los puntos iniciales  $(x_i, y_i)$ , puntos finales  $(x_f, y_f)$  y los puntos intermedios  $P_1(x, y)$  y  $P_2(x, y)$ , lo cual también es llamado como generación de trayectoria multipunto como se muestra en la figura 3.16, donde se definen las condiciones que debe tener la trayectoria en cada punto.

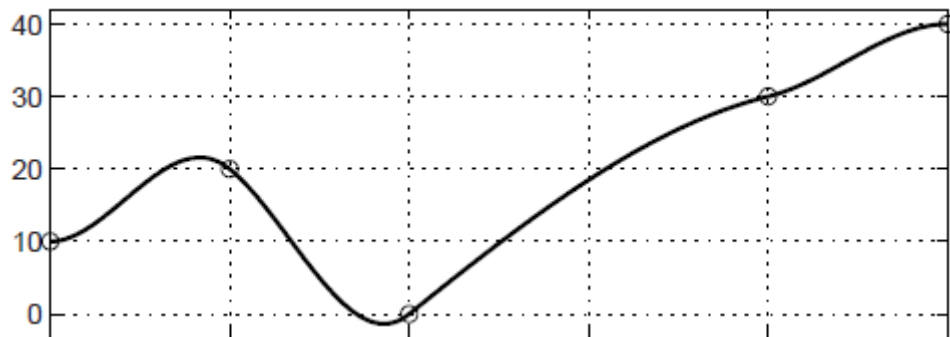


Figura 3.16: Ejemplo de Trayectoria Multipunto (Biagiotti & Melchiorri,2008)

Que, en este caso, donde máximo se tendrá 3 segmentos, conlleva a tener máximo 4 puntos principales en la trayectoria. Entonces, de la misma forma que se aplicó el diseño de la trayectoria basado en polinomios anteriormente, en este caso se definen las siguientes condiciones de cada punto en la trayectoria las cuales se utilizarán para calcular los parámetros del polinomio para cada segmento:

- 1er Punto:  $x = x_i$  ,  $y = y_i$  ,  $\theta = \theta_i$  ,  $v = 0$  ,  $\eta_v = 0$  ,  $\delta = 0$
- 2do Punto:  $x = P_{1x}$  ,  $y = P_{1y}$  ,  $\theta = \phi$  ,  $v = 0$  ,  $\eta_v = 0$  ,  $\delta = 0$
- 3er Punto:  $x = P_{2x}$  ,  $y = P_{2y}$  ,  $\theta = \phi$  ,  $v = 0$  ,  $\eta_v = 0$  ,  $\delta = 0$
- 4to Punto:  $x = x_f$  ,  $y = y_f$  ,  $\theta = \theta_f$  ,  $v = 0$  ,  $\eta_v = 0$  ,  $\delta = 0$



Siendo las condiciones iniciales y finales en cada segmento:

- 1er Segmento:
  - Condiciones iniciales → 1er Punto
  - Condiciones finales → 2do Punto
  
- 2do Segmento:
  - Condiciones iniciales → 2do Punto
  - Condiciones finales → 3er Punto
  
- 3er Segmento:
  - Condiciones iniciales → 3er Punto
  - Condiciones finales → 4to Punto

Para realizar la generación del vector de datos de la trayectoria se debe realizar los siguientes pasos:

- Calcular la mínima trayectoria posible utilizando el método de líneas rectas y arcos de círculos.
- Almacenar los valores de los puntos tangenciales, ángulo de la línea recta y sentidos de la velocidad en cada segmento.
- Generar trayectoria multipunto basada en polinomio tomando en cuenta las condiciones límites en cada segmento.

### **3.3. Generación de Trayectoria para Tiempo Mínimo**

#### **3.3.1. Fundamentos Teóricos de Control Óptimo**

La teoría del control óptimo se basa en la optimización de parámetros al encontrar los valores de  $m$  parámetros  $u_1, u_2, \dots, u_m$  que minimizan una función dada, dependiente de estos parámetros (Bryson, 1975).

$$L(u_1, u_2, \dots, u_m)$$

Donde se define el vector de parámetros de la forma:

$$u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{bmatrix}$$

Por lo que se puede escribir la función a minimizar como  $L(u)$ . Si no existen restricciones en los valores de  $u$ , y si la función  $L(u)$  tiene primera y segunda derivada, las condiciones necesarias para la minimización deben cumplir que:

$$\frac{\partial L}{\partial u} = 0 \quad (3.12)$$

En un caso más general donde se presentan problemas de optimización de parámetros en sistemas de control, involucra encontrar los valores del vector  $u$  que minimice una función de la forma:

$$L(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_m)$$

Donde los  $n$  parámetros  $x_1, x_2, \dots, x_n$  vienen a ser las variables de estado del sistema y los  $m$  parámetros  $u_1, u_2, \dots, u_m$  vienen a ser las variables de control, donde también se suelen presentar  $n$  restricciones de igualdad de la forma:

$$\begin{aligned} f_1(x; u) &= 0 \\ &\vdots \\ f_n(x; u) &= 0 \end{aligned} \quad (3.13)$$

Y se define como el vector de restricciones de igualdad.

$$f = \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix}$$

Por lo que el problema se puede definir de tal manera que: se debe buscar el valor del vector  $u$  que minimice la función  $L(x, u)$ , donde el vector de estado  $x$  está limitado por la restricción  $f(x, u) = 0$ .

Para poder calcular los valores de  $u$  que minimicen  $L(x, u)$  utilizando las restricciones que se presenten, se define la función a minimizar de la siguiente manera:

$$\mathbb{L}(x, u, \lambda) = L(x, u) + \sum_{i=1}^n \lambda_i f_i(x, u) = L(x, u) + \lambda^T f(x, u) \quad (3.14)$$

Donde  $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_n]$  vienen a ser el vector de multiplicadores de Lagrange.

Aplicando la teoría de optimización para un sistema continuo, se debe tener en consideración que tipo de restricciones se pueden presentar y como formular la función a minimizar. Para un sistema continuo dado, considerando un sistema no lineal de la forma:

$$\dot{x} = f[x(t), u(t), t]; \quad x(t_0) \text{ conocido}; \quad t_0 \leq t \leq t_f \quad (3.15)$$

Se define una función de costo de la forma:

$$J = \Phi[x(t_f), t_f] + \int_{t_0}^{t_f} L[x(t), u(t), t] dt \quad (3.16)$$

Donde,  $\Phi[x(t_f), t_f]$  viene a ser la función de restricción final del vector de estado.

El problema es encontrar el valor de la señal de control  $u(t)$  que minimice (o maximice) la función  $J$ . Integrando las restricciones al sistema, se adjunta los multiplicadores de Lagrange a la función de la ecuación (3.16), teniendo así:

$$J = \Phi[x(t_f), t_f] + \int_{t_0}^{t_f} [L[x(t), u(t), t] + \lambda^T(t)\{f[x(t), u(t), t] - \dot{x}\}] dt \quad (3.17)$$

Por conveniencia se define la función H (el Hamiltonian) de la siguiente manera:

$$H = H[x(t), u(t), \lambda(t), t] = L[x(t), u(t), t] + \lambda^T(t)f[x(t), u(t), t] \quad (3.18)$$

Que es la función base mediante la cual se trabaja la aplicación de la teoría de control óptimo a los sistemas continuos. El cálculo de los valores de la señal de control, que minimiza el sistema, se calcula en base al cumplimiento de ciertas condiciones necesarias de optimalidad, las cuales son:

$$\lambda_{t_f} = \frac{\partial \Phi(x_{t_f})}{\partial x_{t_f}} \quad (3.19a)$$

$$\lambda = - \frac{\partial H}{\partial x} \quad (3.19b)$$

$$\frac{\partial H}{\partial u} = 0 \quad (3.19c)$$

Resolviendo las condiciones de optimalidad, se llega a encontrar el valor de  $u(t)$  que minimice la función de costo cumpliendo los requisitos de restricción del sistema continuo.

### 3.3.2. Problema de Control Óptimo para Tiempo Mínimo

Como se indicó anteriormente, el control óptimo se basa en encontrar el valor del vector de control  $u(t)$  que minimice una función dada cumpliendo las restricciones del sistema, para lo cual se considera el rango de tiempo de trabajo  $t_0 \leq t \leq t_f$  y donde el rango del vector  $u(t)$  es libre.

Para un problema de tiempo mínimo, se busca minimizar el tiempo que le lleva al sistema llegar de su valor inicial a su valor final dado, por lo que en este caso se plantea una función de costo de la forma:

$$J = t_f - t_0 = \int_{t_0}^{t_f} dt \quad (3.20)$$

Con lo que se genera que:

$$\Phi = 0 \quad , \quad L = 1$$

El proceso de control para tiempo mínimo se obtiene resolviendo el TPBVP (Two-Point Boundary-Value Problem) dadas en las ecuaciones en (3.19), así como también se cumple para un caso de tiempo mínimo que:

$$(\lambda^T f)_{t=t_f} = -1 \quad (3.21)$$

También se debe de definir el valor de la variable de estado en  $t = t_0$  y en  $t = t_f$  sino no se podría resolver el problema para tiempo mínimo.

Así también en los casos generales de control, el vector de control  $u(t)$  se encuentra regido bajo restricciones de valores límites:

$$u_{min} \leq u(t) \leq u_{max} \quad (3.22)$$

Lo cual se transforma en una restricción de desigualdad tipo:

$$C(u, t) \leq 0 \quad (3.23)$$

Lo que conlleva a que el Hamiltonian tome una nueva forma al integrar esta restricción, obteniéndose:

$$H = L + \lambda^T f + \mu^T C \quad (3.24)$$

Que viene a ser lo mismo que la ecuación (3.18) con el requerimiento adicional que:

$$\mu \begin{cases} \geq 0, & C = 0 \\ = 0, & C < 0 \end{cases} \quad (3.25)$$

Si existen restricciones en las variables de control, es razonable esperar que la solución mínima requiera que las variables de control vayan de un punto a otro de los límites de las restricciones dadas:  $u_{min}$  y  $u_{max}$ , lo que genera el llamado control “bang-bang”, el cual parte de la resolución del Hamiltonian definido en (3.24). Por ejemplo, para el caso de tener un sistema continuo de la forma:

$$\dot{x} = f(x, t) + g(t)u, \quad u \in \mathcal{R}^{1 \times 1} \quad (3.26)$$

Se define el Hamiltonian correspondiente al control “bang-bang” :

$$H = 1 + \lambda^T (f(x, t) + g(t)u) \quad (3.27)$$

y para minimizar  $H$  con respecto a  $u$ , se requiere que:

$$u(t) = \begin{cases} u_{max}, & \lambda^T g(t) < 0 \\ 0, & \lambda^T g(t) = 0 \\ u_{min}, & \lambda^T g(t) > 0 \end{cases} \quad (3.28)$$

Que se puede expresar también como:

$$u(t) = -\text{sign}(\lambda^T g(t)) \quad (3.29)$$

Donde  $\lambda^T g(t)$  viene a ser la función switching que va a definir el valor de  $u(t)$ . Este caso trabaja para un sistema con una variable de control, pero explica claramente que se puede definir los valores de la señal de control basándose en una función que permita conmutar entre los valores límites de la señal de control o un valor intermedio iguala cero.

Aplicando este concepto al sistema de esta investigación que es el robot móvil, utilizando el sistema de la ecuación (2.2) donde se tienen 2 variables de control definidas por  $v$  y  $\omega$ .

$$x = f(x)v + g(x)\omega \quad (3.30)$$

Se busca obtener una función switching para cada variable de control. Se define  $\varphi(t)$  como la función switching de  $v$  y  $\psi(t)$  como la función switching de  $\omega$ , las cuales son funciones dadas por:

$$\varphi(t) = [\lambda(t), f(x)] \quad (3.31a)$$

$$\psi(t) = [\lambda(t), g(x)] \quad (3.31b)$$

Pero en el caso del robot móvil tipo Ackerman se trabaja con la señal de control  $\delta$ , por lo que la función switching de  $\omega = v u$ , va a depender de la velocidad para que de esta manera se obtenga el control directo de  $\delta$ , que conlleva a que  $\psi(t)$  dependa también de  $v$ .

$$\psi(t) = [\lambda(t), g(x), v] \quad (3.32)$$

Estas funciones permiten generar las siguientes condiciones de cambio en las señales de control, con las cuales se genera la trayectoria para tiempo mínimo.

$$v = \begin{cases} v_{max}, & \varphi(t) < 0 \\ 0, & \varphi(t) = 0 \\ v_{min}, & \varphi(t) > 0 \end{cases} \quad (3.33a)$$

$$\delta = \begin{cases} \delta_{max}, & \psi(t) < 0 \\ 0, & \psi(t) = 0 \\ -\delta_{max}, & \psi(t) > 0 \end{cases} \quad (3.33b)$$

Se plantea utilizar el concepto de la función switching aplicado a la generación de trayectoria basada en líneas rectas y arcos de circunferencia, ya que como se aprecia en (3.33b), se trabaja también con 3 estados para la señal de control  $\delta$  y en (3.33a) se aprecia los cambios de sentido de la velocidad en cada segmento según corresponda. Las funciones switching  $\varphi(t)$  y  $\psi(t)$  son diseñadas en base a la geometría que requiere la trayectoria tomando en cuenta los siguientes datos:  $\phi, P_1(x, y), P_2(x, y)$ ; que son obtenidos del análisis del algoritmo en la generación de los puntos clave de la trayectoria mínima.

El objetivo de utilizar las funciones switching, es para poder generar el vector de puntos de la trayectoria deseada correspondiente a las variables de estado  $x, y$ , que permitan generar el ángulo de dirección y la velocidad correspondiente en cada

segmento; para luego aplicar la ley de control utilizando dicha trayectoria donde se validará que se generen los valores límites de las variables de control en el control del sistema por dicha trayectoria.

### 3.3.3. Generación del Vector de Puntos Utilizando la Función Switching

Utilizando el concepto de las funciones switching  $\varphi(t)$  y  $\psi(t)$  indicadas en las ecuaciones (3.31a) y (3.31b) respectivamente, se define una función matemática para cada caso, utilizando los datos obtenidos en la generación de trayectoria basado en rectas líneas y arcos de circunferencia.

Para el caso de la función  $\varphi(t)$ , que corresponde a la velocidad, se define la siguiente función base:

$$\ell_v : \left( Y - X * \frac{1}{\tan \theta} - (y - x * \frac{1}{\tan \theta}) \right) * \text{sign}(\cos \theta) \quad (3.34)$$

La cual viene a ser la ecuación de una recta imaginaria que atraviesa al robot móvil de manera perpendicular como se muestra en la figura 3.17.

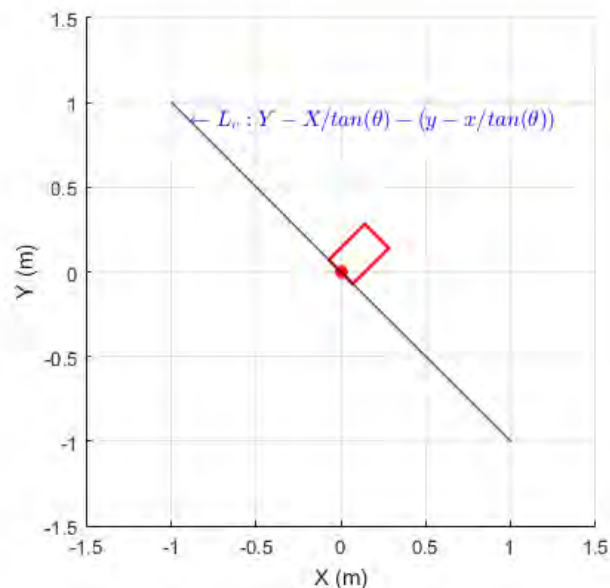


Figura 3.17: Recta Imaginaria para el Cálculo del Sentido de la Velocidad

Y a la cual se le multiplica por el signo del coseno del ángulo  $\theta$ , el cual cambiará de signo la función dependiendo de la orientación del vehículo. De tal manera que, aplicando el concepto de inecuaciones, se puede conocer si un punto se encuentra por debajo o encima de una recta si el valor de la ecuación es menor o mayor a cero. Esto permite conocer si los puntos de destinos en cada segmento se encuentran por delante o detrás del vehículo y así definir el sentido de la velocidad, siendo  $(x, y, \theta)$  las variables de estado en cada instante del vehículo y  $(X, Y)$  los puntos de destino en cada segmento, siendo estos  $P_1(x, y)$ ,  $P_2(x, y)$  o  $(x_f, y_f)$  según sea el caso.

Para el caso de la función  $\psi(t)$ , que corresponde al ángulo de dirección, se define la siguiente función base:

$$l_\delta: (Y - X * \tan \theta - (y - x * \tan \theta)) * \text{sign}(\cos \theta) \quad (3.35)$$

La cual viene a ser la ecuación de una recta imaginaria que atraviesa al robot móvil en la misma dirección de este como se muestra en la figura 3.18.

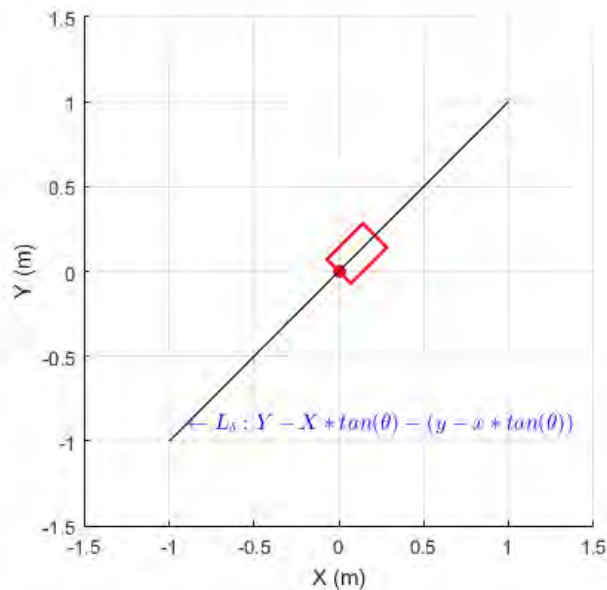


Figura 3.18: Recta Imaginaria para el Cálculo del Sentido del Ángulo  $\delta$

Y a la cual se le multiplica por el signo del coseno del ángulo  $\theta$ , el cual cambiará de signo la función dependiendo de la orientación del vehículo. De igual manera que el caso de la velocidad, se puede conocer si los puntos de destinos en cada segmento



se encuentran a la derecha o izquierda del vehículo y así definir el sentido de giro del vehículo.

Teniendo las funciones correspondientes para cada variable de control, se pasa a realizar la estimación de los puntos de la trayectoria como en el siguiente esquema:

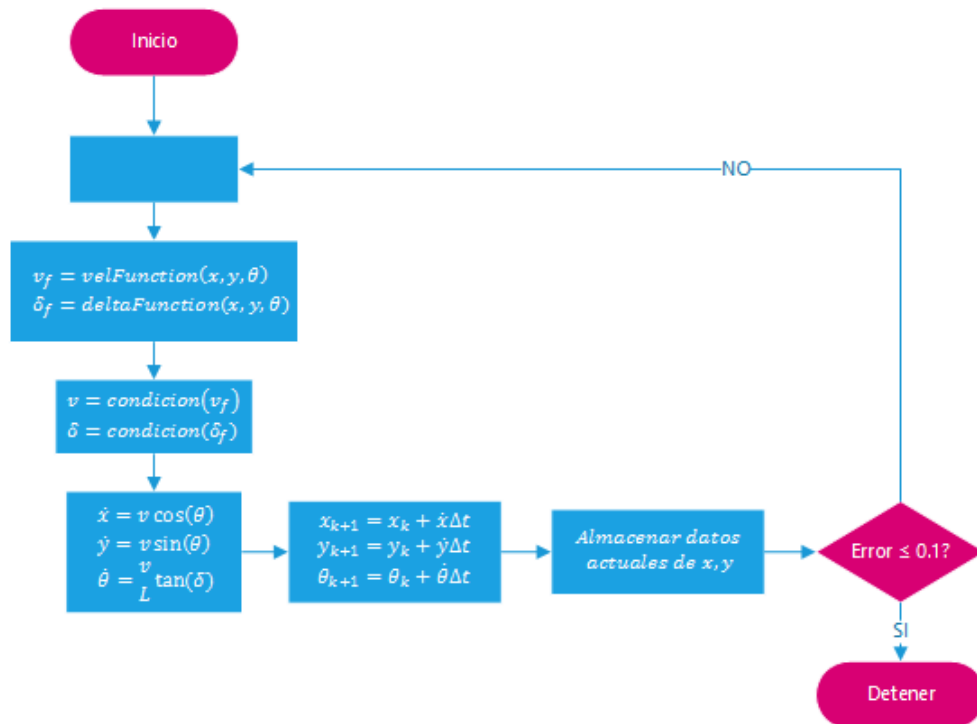


Figura 3.19: Lazo de Estimación de Puntos de la Trayectoria Deseada

En el esquema de la figura 3.19 se observa el lazo que permite calcular el vector de puntos de la trayectoria deseada, partiendo de obtener los valores de las funciones switching para cada variable de control y luego en base a las condiciones indicadas en las ecuaciones (3.33a) y (3.33b) definir el valor de cada variable. Después, introducimos los valores de las señales de control al modelo del robot móvil y actualizamos los valores actuales. Luego, se almacenan los datos  $x$ ,  $y$  actuales de cada lazo, siendo estos valores, los vectores de la trayectoria deseada. Finalmente, el lazo se detiene cuando el error medio cuadrático sea menor igual a 0.1 (este valor se puede variar).

Para obtener el vector de puntos de las siguientes 2 derivadas de cada variable, las mismas que también se calcularon en el método polinómico, se debe derivar hasta 2 veces los vectores de puntos obtenidos siguiendo el proceso de las funciones en (3.36) y (3.37).

$$x(k) = \begin{cases} 0, & k = 1 \\ \frac{x(k) - x(k-1)}{\Delta t}, & k > 1 \end{cases}, \quad k = 1, 2, \dots, N \quad (3.36)$$

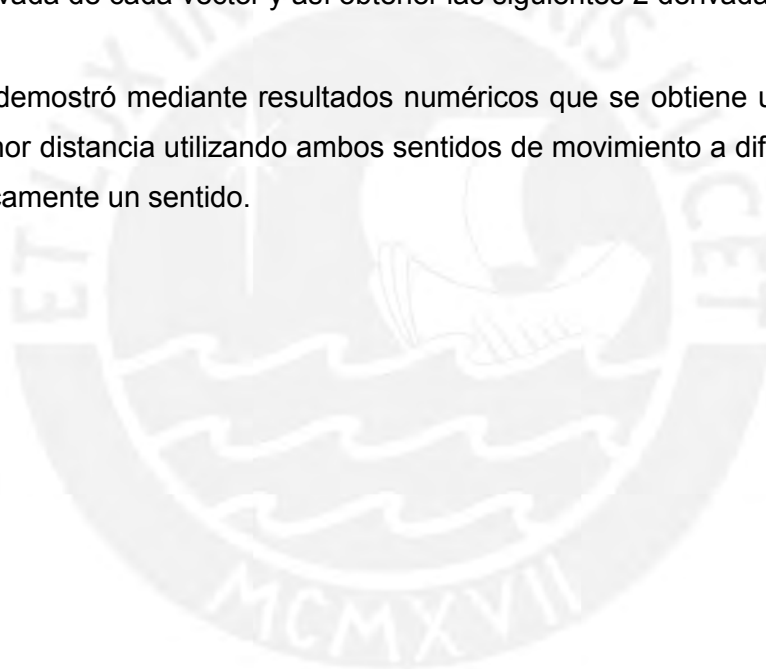
$$x(k) = \begin{cases} 0, & k = 1 \\ \frac{x(k) - x(k-1)}{\Delta t}, & k > 1 \end{cases}, \quad k = 1, 2, \dots, N \quad (3.37)$$

Donde  $N$  viene a ser el tamaño del vector de puntos principales y las funciones se aplican para el vector en  $y$  de igual manera.

### 3.4. Conclusiones Preliminares

- Se demostró que la trayectoria mínima posible que requiere un vehículo para llegar a su destino se puede basar en un análisis geométrico, donde se considera principalmente 3 estados de movimiento del robot móvil en la señal de control  $\delta$  y considerando un movimiento en avance y retroceso, lo que permite un gran margen de libertad en el movimiento del vehículo.
- Se plantearon 2 técnicas de generación de trayectoria basadas en la teoría de Reeds & Shepp sobre trayectoria mínima basado en un análisis geométrico del espacio de trabajo del robot móvil.
- Utilizando el método de la función polinómica, se demostró mediante simulaciones que es posible generar una trayectoria utilizando únicamente los puntos de los segmentos de la trayectoria obtenidos del análisis geométrico de la teoría de Reeds & Shepp y así automáticamente obtener los vectores de cada variable deseada y sus respectivas derivadas con velocidad variable.
- Utilizando la teoría de control óptimo, se definieron las funciones switching, diseñadas en base a los puntos clave generados en el análisis geométrico de la teoría de Reeds & Shepp, mediante las cuales se pudieron generar los vectores de las variables realizando un simple lazo de programa, que corresponden a trabajar únicamente con velocidad máxima o mínima.

- La ventaja del método mediante función polinómica es que permite generar automáticamente los vectores de puntos y sus derivadas mediante una ecuación matricial directamente, entregando un vector de puntos con una velocidad variable acorde a un comportamiento suave que requiere el robot móvil, mientras que su desventaja es que no genera trabajo para un funcionamiento de tiempo mínimo debido a la velocidad variable.
- La ventaja del método mediante función switching es que permite trabajar únicamente con los valores extremos de la velocidad lo cual conlleva a que el robot móvil llegue a su destino en el tiempo mínimo, mientras que su desventaja es que, para generar el vector de puntos de las variables se requiere de un lazo de programa para generar cada punto de la trayectoria y luego aplicar la derivada de cada vector y así obtener las siguientes 2 derivadas necesarias.
- Se demostró mediante resultados numéricos que se obtiene una trayectoria de menor distancia utilizando ambos sentidos de movimiento a diferencia de utilizar únicamente un sentido.



## CAPÍTULO 4. DISEÑO DEL SISTEMA DE CONTROL DEL ROBOT MÓVIL

### 4.1. Introducción

El objetivo de diseñar una ley de control es hacer que el vehículo vaya desde una posición inicial  $(x_i, y_i, \theta_i)$  hasta una posición final  $(x_f, y_f, \theta_f)$  siguiendo una trayectoria deseada  $(x_d, y_d, \theta_d)$ , es decir, el controlador debe permitir que el vehículo siga dicha trayectoria, bajo la condición de que el vehículo tenga como punto de inicio un punto dentro de la trayectoria o también cercano a esta, ya que el controlador buscará minimizar el error que hay entre la posición del vehículo y la trayectoria que se desea seguir.

El tema del control de robots móviles se ve enfocado principalmente en la no linealidad de estos, lo cual conlleva a analizar las opciones para su respectivo control que se realiza de acuerdo a si se desea tener un control en cierto rango de respuesta del sistema o en su rango completo. Mediante estas 2 opciones se define la selección del controlador a utilizar, ya que si se desea trabajar en un pequeño rango de operación se opta por utilizar un control lineal llevando a que se tenga que linealizar el modelo del sistema en un punto de operación, conocido también como linealización por aproximación, y de esta manera trabajar en la vecindad del punto de operación.

Para un sistema de la forma:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{4.1}$$

Considerando la linealización Jacobiana sobre un punto de operación del sistema de la siguiente forma:

$$\begin{aligned} \dot{x} &= \left[ \frac{\partial f(x_o)}{\partial x} + \frac{\partial g(x_o)}{\partial x} u_o \right] (x - x_o) + g(x_o)(u - u_o) \\ y - y_o &= \frac{\partial h(x_o)}{\partial x} (x - x_o) \end{aligned} \tag{4.2}$$

Puede ser representada por un sistema de espacio estado:

$$\begin{aligned}x &= Ax + Bu \\y &= Cx\end{aligned}\tag{4.3}$$

Y que es importante mencionar que este modelo es una representación exacta del modelo no lineal solo en la vecindad del punto de operación.

La otra opción, que es trabajar en todo el rango de operación del sistema, conlleva a utilizar un controlador no lineal. Es claro que realizar un control no lineal es más complejo, por lo que durante años se han planteado diferentes estrategias de control no lineal, tales como:

- BackStepping
- Linealización por Realimentación
- Superficies Deslizantes
- Redes Neuronales
- Lógica Difusa
- Redes Neuro-Difusas

En esta investigación se ha decidido utilizar la técnica de “Linealización por Realimentación” como controlador para el robot móvil.

## 4.2. Linealización por Realimentación

Esta técnica se basa en dos operaciones principales:

- Cambio de variables del sistema no lineal
- Realimentación utilizando las nuevas variables

La idea del cambio de variables es buscar una transformación  $z = T(x)$ , tal que permita transformar el modelo no lineal a una forma lineal:

$$\begin{aligned}z &= Az + Bv \\w &= Cz\end{aligned}\tag{4.4}$$

Para corroborar que la transformación representa correctamente al sistema no lineal principal, esta transformación debe ser considerada un difeomorfismo del sistema original; es decir, que puede ser un sistema invertible o, mejor dicho, que se puede regresar al sistema original a partir de la transformación hecha (Van Nieuwstadt,1997).

Luego de obtener la transformación del sistema, se realiza la realimentación utilizando el nuevo vector de variables  $z$ , a las cuales se les aplica la ley de control, donde dicho controlador viene a ser un controlador lineal. Para encontrar la correcta transformación de las variables del sistema, se usan diferentes técnicas, donde una de ellas es la técnica Flatness Diferencial, y es la que se utilizará en esta investigación (Oriolo, De Luca & Vendittelli,2002).

### 4.3. Flatness Diferencial

Debido a que el sistema no es estáticamente linealizable en realimentación, se propone el uso de la técnica Flatness Diferencial que es una técnica de transformación o cambio de variables del sistema. Esta técnica propone generar un modelo basado en salidas flat (flat outputs), las cuales llevan a poder controlar el sistema mediante estas nuevas variables (Rigatos,2015; Khan, Ali & Riaz,2014).

El flatness diferencial se define de la siguiente manera: Teniendo un sistema  $\dot{x} = f(x, u)$  con vector de estados  $x \in R^n$ , y vector de control  $u \in R^m$  y donde  $f(x, u)$  es flat diferencial si existe un vector  $y \in R^m$  de la forma (Rigatos,2015; Levine,2009):

$$y = h(x, u, \dot{u}, \dots, u^{(r)})$$

Tal que:

$$x = \phi(y, \dot{y}, \dots, y^{(q)})$$

$$u = \alpha(y, \dot{y}, \dots, y^{(q)})$$

Donde  $h, \phi, \alpha$  son funciones de derivadas continuas. Lo que significa que la dinámica del sistema no lineal puede ser expresada en función del nuevo vector de estado definido como salidas flat y sus derivadas. Por lo que al momento de elegir nuestras nuevas variables (salidas flat), se debe verificar que podemos definir todas

las variables de estado y de control de nuestro sistema en base a las salidas flat (Rigatos,2015).

Para el sistema no lineal del robot móvil de la ecuación 2.4, se definen las siguientes salidas flat (De Luca, Oriolo & Samson,1998):

$$\begin{aligned} F_1 &= x \\ F_2 &= y \end{aligned} \quad (4.5)$$

Para saber si se puede aplicar esta técnica al sistema, debemos corroborar que todas las variables puedan ser definidas en base a las salidas flat propuestas.

$$\begin{aligned} \theta &= \tan^{-1} \left( \frac{y}{x} \right), \quad v = \sqrt{x^2 + y^2} \\ \delta &= \tan^{-1} \left( L \frac{y \dot{x} - y \dot{x}_3}{(x^2 + y^2)^2} \right) \end{aligned} \quad (4.6)$$

Se observa que efectivamente todas las variables que comprenden al robot móvil pueden ser definidas en base a las salidas flat seleccionadas, con lo cual se concluye que se puede utilizar esta técnica y las salidas flat.

Ahora se debe derivar cada salida flat hasta su grado relativo; esto es, hasta que se obtenga alguna derivada en función de las variables de control.

Colocando el modelo del sistema de la siguiente manera por simplicidad:

$$\begin{aligned} \dot{x} &= v \cos(\theta) \\ \dot{y} &= v \sin(\theta) \\ \dot{\theta} &= v u \end{aligned} \quad (4.7)$$

Siendo:  $u = \frac{1}{L} \tan(\delta)$

Derivando con respecto al tiempo se tiene:

$$\begin{aligned} F_1 &= \dot{x} = v \cos(\theta) \\ F_2 &= \dot{y} = v \sin(\theta) \end{aligned} \quad (4.8)$$

Se ve que la primera derivada de las salidas flat solo dependen de la señal de control  $v$  y no de la señal  $u$ , para lo cual se introduce una nueva variable de control

$v = \eta_v$  y extendiendo el sistema a (Tang,2009; Tang, Miller, Krovi, Ryu & Agrawal,2011):

$$\begin{aligned}x &= v \cos(\theta) \\y &= v \sin(\theta) \\ \theta &= v u \\ v &= \eta_v\end{aligned}\tag{4.9}$$

Con lo cual volviendo a derivar  $F_1$  y  $F_2$  con respecto al tiempo:

$$F_1 = \eta_v \cos(\theta) - v^2 u \sin(\theta)\tag{4.10}$$

$$F_2 = \eta_v \sin(\theta) + v^2 u \cos(\theta)$$

Obteniendo hasta la segunda derivada de las salidas flat que dependen de las variables de control, con lo que  $F_1$  y  $F_2$  es lineal con respecto a las nuevas variables de control, con lo que se puede realizar el cálculo inverso de las señales de control mediante la segunda derivada de las salidas flat (Tang,2009; Tang, Miller, Krovi, Ryu & Agrawal,2011):

$$F = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} \eta_v \cos(\theta) & -v^2 \sin(\theta) \\ \eta_v \sin(\theta) & v^2 \cos(\theta) \end{bmatrix} \begin{bmatrix} u \end{bmatrix}\tag{4.11}$$

Se definen entonces todas las ecuaciones de las salidas flat con respecto a las variables del sistema:

$$F_1 = x, \quad F_2 = y$$

$$F_1 = x = v \cos(\theta), \quad F_2 = y = v \sin(\theta)\tag{4.12}$$

$$F_1 = \eta_v \cos(\theta) - v^2 u \sin(\theta), \quad F_2 = \eta_v \sin(\theta) + v^2 u \cos(\theta)$$

#### 4.4. Control de Trayectoria

Luego de haber definido las salidas flat y habiendo obtenido el orden derivativo de estas salidas que dependen de las señales de control, se observa que el control



debe ser realizado a  $F$ , para lo cual se aplica un control lineal con la una ley de control realimentado.

#### 4.4.1. Esquema de Control

La ley de control para las nuevas entradas puede ser definida mediante la siguiente ley de control lineal:

$$F_1 = F_1^a - p_1 \varepsilon_1 - p_0 \varepsilon_1 \quad (4.13)$$

$$F_2 = F_2^d - q_1 \varepsilon_2 - q_0 \varepsilon_2$$

Donde  $p_0, p_1, q_0, q_1$  son las ganancias del controlador, las cuales serán calculadas utilizando el método de Riccati, y  $\varepsilon_k, k = 1,2$ , es el error de las salidas flat definido como:

$$\varepsilon_1 = F_1 - F_1^d, \quad \varepsilon_2 = F_2 - F_2^d \quad (4.14)$$

$$\varepsilon_1 = F_1 - F_1^d, \quad \varepsilon_2 = F_2 - F_2^d$$

Donde los valores de  $F^d, F^d, F^d$  vienen a ser la salida flat deseada y sus respectivas derivadas, las cuales definirán la trayectoria a seguir deseada por el robot móvil.

De esta manera en base a las salidas flat deseadas (diseñado en base a la trayectoria deseada), se calculan las variables del sistema y se obtienen las señales de control del robot móvil que son  $v$  y  $\delta$ .

Obteniendo  $F = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$  se puede calcular  $v$  y  $u$  mediante la siguiente forma matricial:



$$\begin{bmatrix} F_2 \\ v \\ u \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -v^2 \sin(\theta) \\ \sin(\theta) & v^2 \cos(\theta) \end{bmatrix}^{-1} F \begin{bmatrix} 1 \\ F_2 \end{bmatrix} \quad (4.15)$$

O también mediante las siguientes ecuaciones:

—

$$\eta_v = F_1 \cos(\theta) + F_2 \sin(\theta)$$

$$u = \frac{1}{v^2} (-F_1 \sin(\theta) + F_2 \cos(\theta)) \quad (4.16)$$



Nótese que cuando  $v = 0$ , la matriz se vuelve singular, por lo tanto, el sistema trabaja en avance o retroceso, pero no continuamente en ambos sentidos para una misma trayectoria debido al cruce por cero y cuando el móvil parte del reposo y finaliza su trayectoria, por lo que para cuestiones prácticas se utilizará un valor bajo de la velocidad en estos 3 casos que la velocidad pasa por cero y utilizaremos un valor muy bajo  $v = 0.001$  y así evitar la singularidad del sistema.

Luego se puede calcular las señales de control del robot móvil realizando las conversiones correspondientes:

$$\begin{aligned} v &= \int \eta_v dt \\ \delta &= \tan^{-1} \left( \frac{u}{L} \right) \end{aligned} \quad (4.17)$$

Así estas señales ingresar al robot móvil y como salida obtendremos las variables de posición y ángulo de inclinación, con los cuales se realimentará el sistema y continuará el control hasta terminar el seguimiento de la trayectoria como se muestra en el siguiente esquema de control:

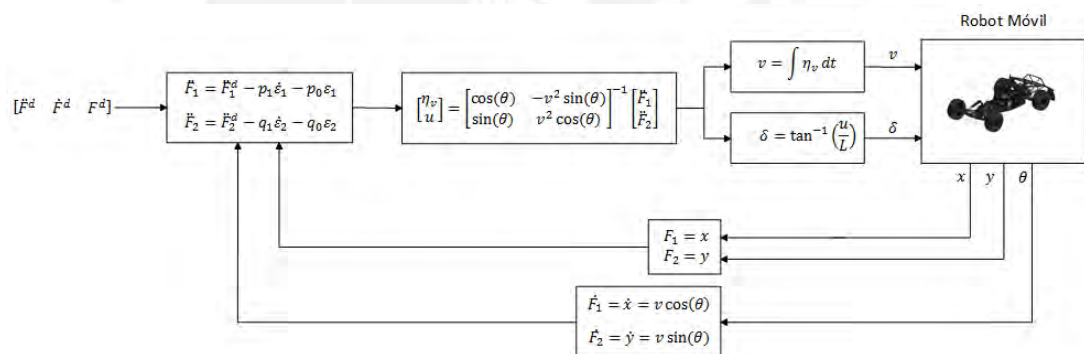


Figura 4.1: Esquema de Control de Trayectoria

#### 4.4.2. Cálculo de las Ganancias del Controlador

Usando la definición de la ley de control de la ecuación 4.9, se posible pasarlo a un sistema lineal de espacio estados de la forma:

$$\begin{aligned} \dot{h} &= Ah + Bw \\ w &= -Kh \end{aligned} \quad (4.18)$$

Donde,  $K$  viene a ser el vector de ganancias del controlador,  $h$  el vector de estado y  $w$  la señal de control. Armando la ley de control a la forma espacio estado, se tiene:

$$\begin{aligned}
 \begin{matrix} F_1 \\ F_2 \end{matrix} &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} F_1 \\ F_2 \end{matrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} [F_1] \\
 \begin{matrix} F_1 \\ F_2 \end{matrix} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} F_1 \\ F_2 \end{matrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{matrix} F_1 \\ F_2 \end{matrix} \\
 \begin{matrix} F_1 \\ F_2 \end{matrix} &= \begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} \\ K_{21} & K_{22} & K_{23} & K_{24} \end{bmatrix} \begin{matrix} F_1 \\ F_2 \end{matrix}
 \end{aligned} \tag{4.19}$$

Dónde:  $K_{11} = p_0, K_{13} = p_1, K_{22} = q_0, K_{24} = q_1$ .

Para verificar que el sistema es controlable, se analiza que la matriz de controlabilidad sea de rango completo ( $n=4$ ):

$$\begin{aligned}
 Co = [B \quad AB \quad A^2B \quad A^3B] &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \\
 rank(Co) &= 4
 \end{aligned}$$

Para obtener la matriz de ganancia se utiliza el método de Riccati:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \tag{4.20}$$

Donde se calcula la matriz  $P$  con la cual se calcula la matriz  $K$ :

$$K = R^{-1}B^T P \tag{4.21}$$

Donde  $Q$  es la matriz de pesos y  $R$  una matriz identidad. Eligiendo la matriz  $Q$  y  $R$  con los siguientes valores:

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 1000 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Para el resolver la ecuación de Riccati se utiliza el comando "are" de Matlab, el cual arroja la matriz de  $P$  automáticamente:

$$P = are(A, BR^{-1}B^T, Q) \tag{4.22}$$

Y reemplazando la matriz  $P$  en la ecuación 4.21, se obtiene la matriz de ganancia  $K$ :

$$K = \begin{bmatrix} 7.9517 & 0 & 31.6228 & 0 \\ 0 & 7.9517 & 0 & 31.6228 \end{bmatrix}$$

De lo cual obtenemos los parámetros de ganancia de la ley de control del lazo de control realimentado:

$$p_0 = 7.9517, \quad p_1 = 31.6228, \quad q_0 = 7.9517, \quad q_1 = 31.6228$$

Para corroborar que la matriz de ganancia  $K$  sea apropiada a utilizar, se debe calcular los valores propios en lazo cerrado del sistema espacio-estado de la ley de control y verificar que se encuentren en el semiplano izquierdo:

$$A_{CL} = A - BK = \begin{bmatrix} -7.9517 & 0 & -31.6228 & 0 \\ 0 & -7.9517 & 0 & -31.6228 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Donde sus valores propios son:

$$\begin{aligned} \lambda_1 &= -3.9764 + j3.974 \\ \lambda_2 &= -3.9764 - j3.974 \\ \lambda_3 &= -3.9764 + j3.974 \\ \lambda_4 &= -3.9764 - j3.974 \end{aligned}$$

Lo que corrobora que los parámetros de ganancia obtenidos para la ley de control son óptimos.

#### 4.5. Simulación del Sistema de Control usando Trayectoria Basada en Polinomio

Teniendo la trayectoria de referencia por la cual se desea que el robot móvil se transporte, se puede validar la técnica de control planteada, la cual permitirá guiar al robot móvil. Utilizando el algoritmo del anexo A.3, se puede realizar la simulación de todo el sistema de control en el cual se usa la trayectoria de referencia diseñada en base a la función polinómica.

Cuando se vio el tema de la generación del vector de puntos utilizando función polinómica en el capítulo anterior, se mencionaba que las condiciones iniciales y finales a tomar venían a ser la posición y sus derivadas para una trayectoria dada. Para aplicar ese método al sistema de control planteado utilizando las salidas flat, se debe utilizar como condiciones iniciales y finales para el diseño de la trayectoria, las salidas flat  $F_1$  y  $F_2$  y sus respectivas derivadas usando las formulas en las ecuaciones (4.5) y (4.6), de tal manera que utilizando las condiciones límites:

$$x(t_0), y(t_0), \theta(t_0), v(t_0), \eta_v(t_0), \delta(t_0)$$

$$x(t_f), y(t_f), \theta(t_f), v(t_f), \eta_v(t_f), \delta(t_f)$$

Donde,  $t_0$  refiere a los valores en el tiempo inicial y  $t_f$  refiere a los valores en el tiempo final, se puedan transformar a sus respectivas condiciones límites de las salidas flat como:

$$F_1(t_0), F_1'(t_0), F_1''(t_0), F_2(t_0), F_2'(t_0), F_2''(t_0)$$

$$F_1(t_f), F_1'(t_f), F_1''(t_f), F_2(t_f), F_2'(t_f), F_2''(t_f)$$

Por lo que cualquier trayectoria de  $F_1(t)$  y  $F_2(t)$  pueda ser armada en el espacio de las salidas flat, permitiendo que la trayectoria, mediante las salidas flat, cumplan con las condiciones iniciales y finales.

A continuación, se realizarán 3 pruebas para demostrar que el lazo de control permite trabajar en los extremos del ángulo del timón y con cambios de velocidad (avance y retroceso). Considerando las siguientes condiciones iniciales del vehículo:

### **Prueba1: Velocidad Positiva**

Condiciones iniciales y finales:

$$x_i = 0, y_i = 0, \theta_i = 0^\circ, v_i = 0.001, \eta_{v_i} = 0, \delta_i = 0$$

$$x_f = 2, y_f = 2, \theta_f = -45^\circ, v_f = 0.001, \eta_{v_f} = 0, \delta_f = 0$$

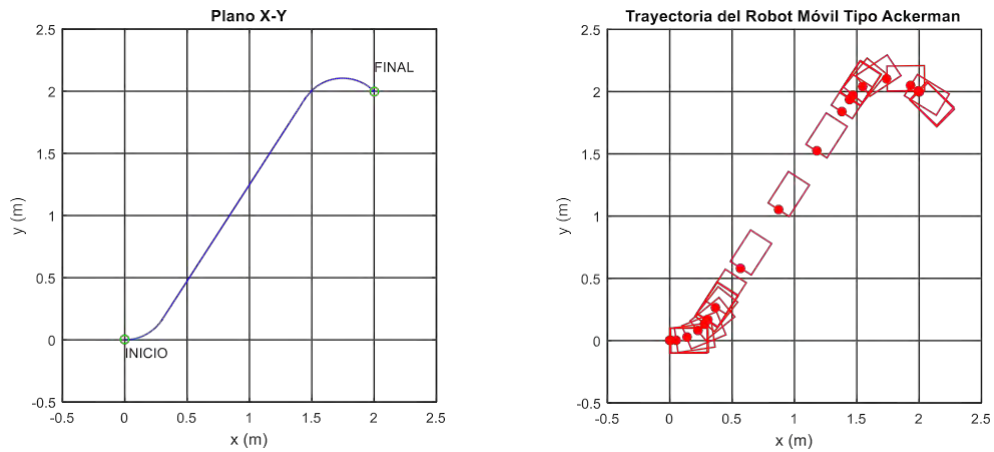


Figura 4.2: Trayectoria con Función Polinómica – Movimiento en Avance

Mostrándose en la figura 4.2 la trayectoria deseada generada en el lado izquierdo y la trayectoria recorrida por el robot móvil, mediante el controlador, en el lado derecho.

En la figura 4.3 se muestran las señales de las variables de estado y en la figura 4.4 se muestran las señales de las variables de control.

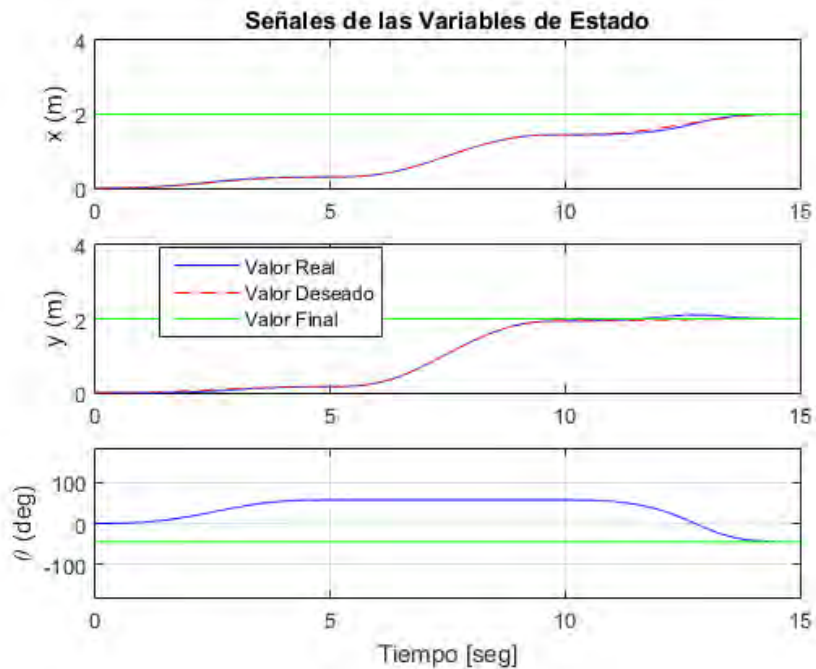


Figura 4.3: Señales de las Variables de Estado con Función Polinómica – Movimiento en Avance

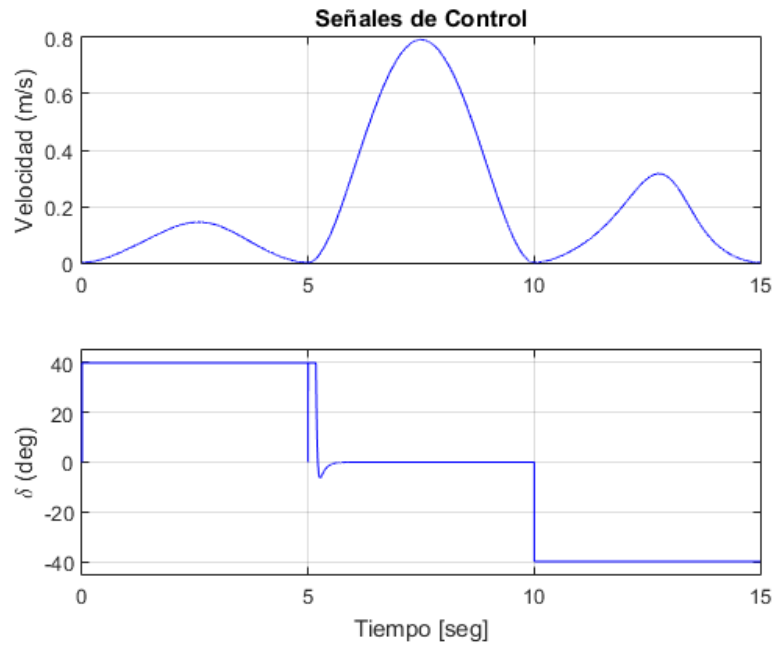


Figura 4.4: Señales de las Variables de Control con Función Polinómica – Movimiento en Avance

### Prueba 2: Velocidad Negativa

Condiciones iniciales y finales:

$$x_i = 0, y_i = 0, \theta_i = -175^\circ, v_i = 0.001, \eta_{v_i} = 0, \delta_i = 0$$

$$x_f = 2, y_f = 2, \theta_f = -160^\circ, v_f = 0.001, \eta_{v_f} = 0, \delta_f = 0$$

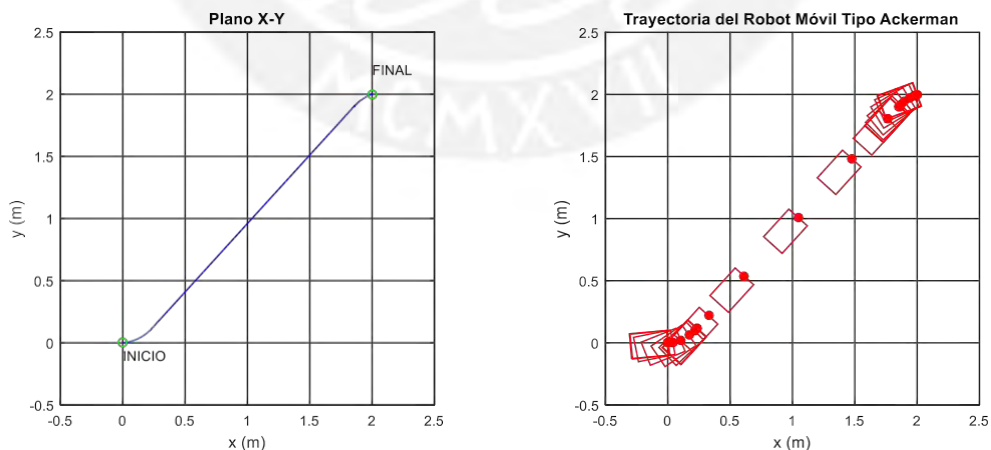


Figura 4.5: Trayectoria con Función Polinómica – Movimiento en Retroceso

En la figura 4.5 se muestra la trayectoria deseada generada en el lado izquierdo y la trayectoria recorrida por el robot móvil en el lado derecho.



En la figura 4.6 se muestran las señales de las variables de estado y en la figura 4.7 se muestran las señales de las variables de control.

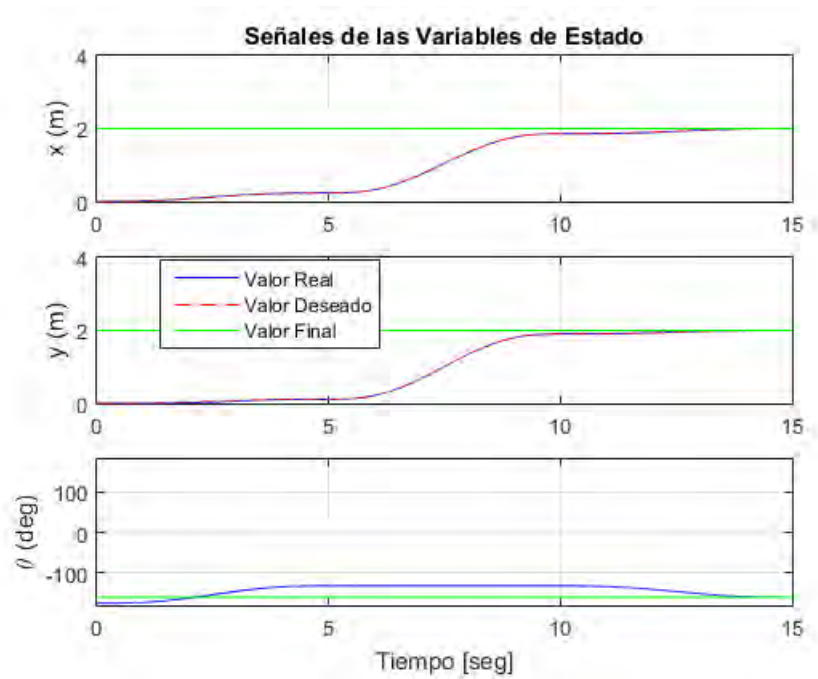


Figura 4.6: Señales de las Variables de Estado con Función Polinómica – Movimiento en Retroceso

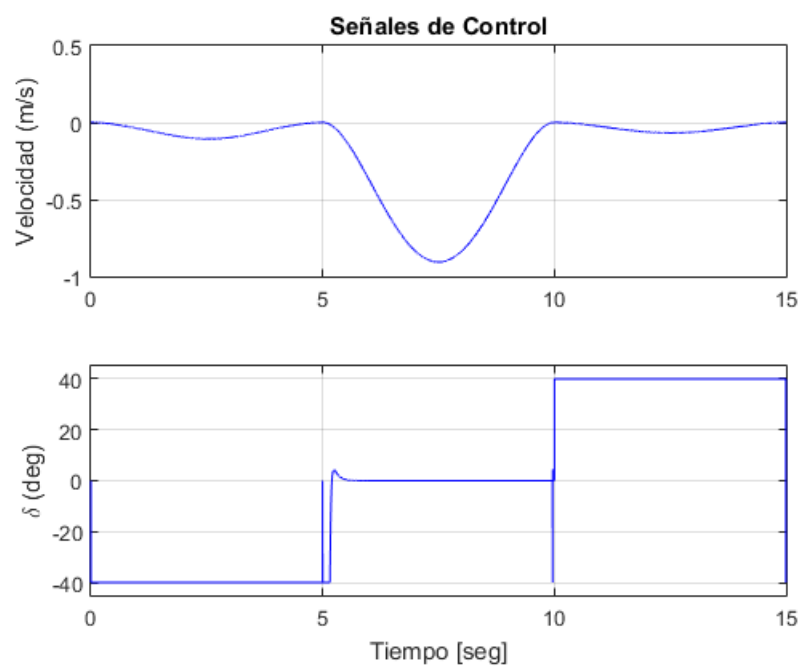


Figura 4.7: Señales de las Variables de Control con Función Polinómica – Movimiento en Retroceso

### Prueba 3: Cambio de Velocidad

Condiciones iniciales y finales:

$$x_i = 0, y_i = 0, \theta_i = -135^\circ, v_i = 0.001, \eta_{v_i} = 0, \delta_i = 0$$
$$x_f = -2, y_f = -2, \theta_f = 50^\circ, v_f = 0.001, \eta_{v_f} = 0, \delta_f = 0$$

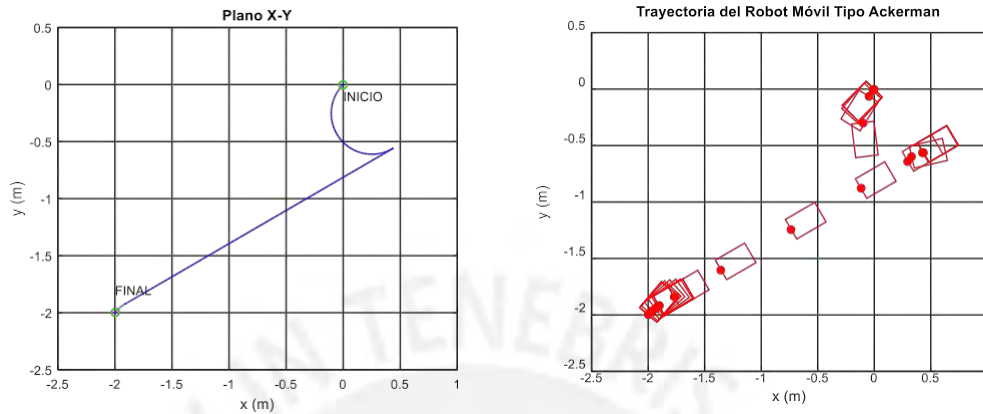


Figura 4.8: Trayectoria con Función Polinómica – Movimiento Variable

En la figura 4.8 se muestra la trayectoria deseada generada en el lado izquierdo y la trayectoria recorrida por el robot móvil en el lado derecho.

En la figura 4.9 se muestran las señales de las variables de estado y en la figura 4.10 se muestran las señales de las variables de control.

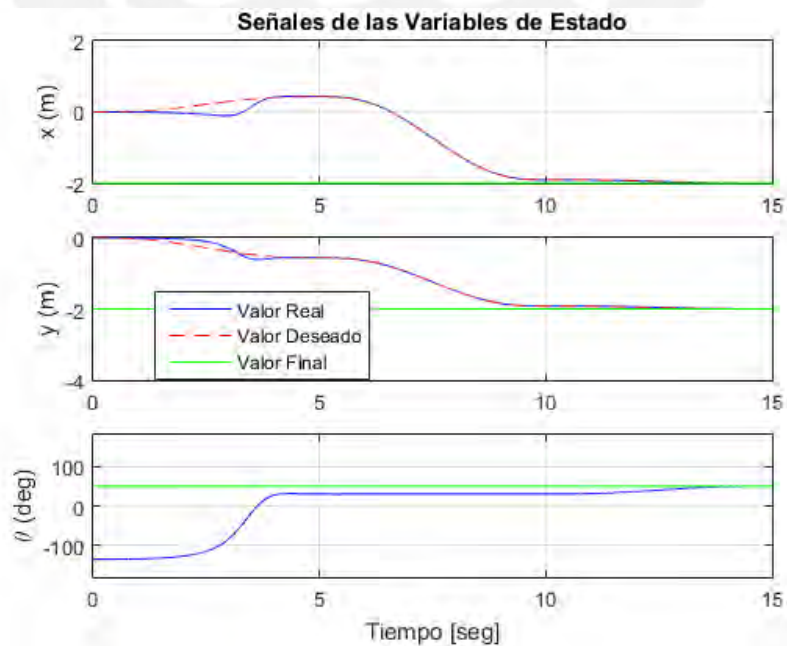


Figura 4.9: Señales de las Variables de Estado con Función Polinómica – Movimiento Variable

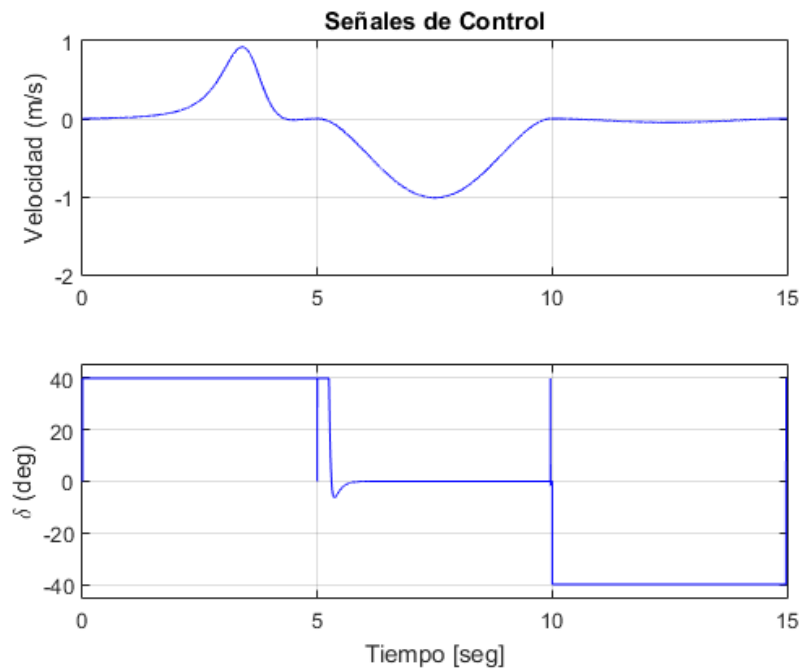


Figura 4.10: Señales de las Variables de Control con Función Polinómica – Movimiento Variable

#### 4.6. Simulación del Sistema de Control usando Trayectoria Basada en la Función Switching

Para este caso, se realizan las mismas pruebas que en el caso de la simulación utilizando trayectoria en base a polinomio, pero ahora utilizando las funciones switching definidas para cada variable de control, mediante el algoritmo descrito en el anexo A.4:

##### **Prueba1: Velocidad Positiva**

Condiciones iniciales y finales:

$$x_i = 0, y_i = 0, \theta_i = 0^\circ, v_i = 0.001, \eta_{v_i} = 0, \delta_i = 0$$

$$x_f = 2, y_f = 2, \theta_f = -45^\circ, v_f = 0.001, \eta_{v_f} = 0, \delta_f = 0$$

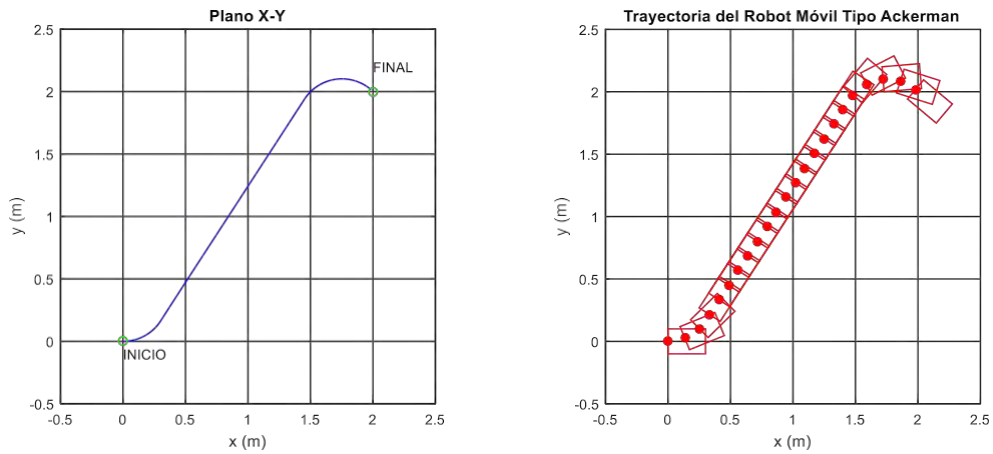


Figura 4.11: Trayectoria con Función Switching – Movimiento en Avance

En la figura 4.11 se muestra la trayectoria deseada generada en el lado izquierdo y la trayectoria recorrida por el robot móvil en el lado derecho.

En la figura 4.12 se muestran las señales de las variables de estado y en la figura 4.13 se muestran las señales de las variables de control.

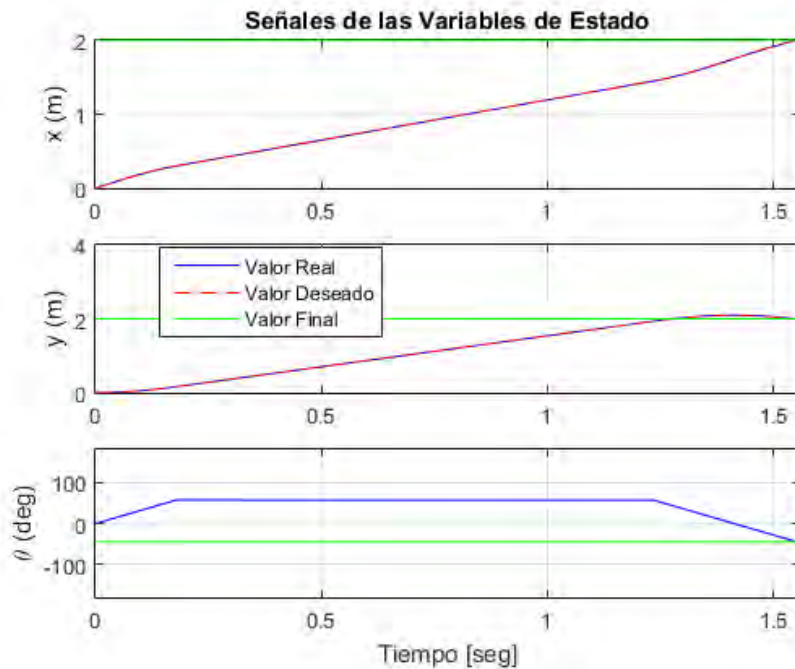


Figura 4.12: Señales de las Variables de Estado con Función Switching – Movimiento en Avance

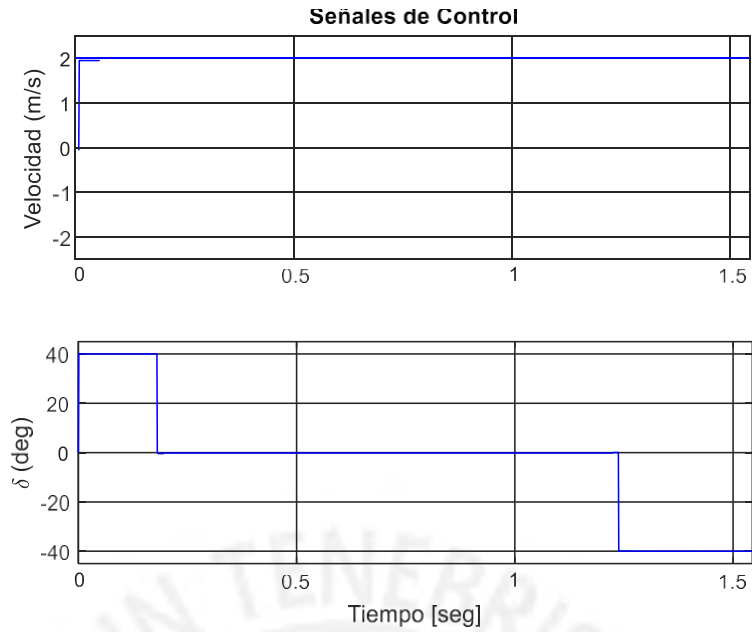


Figura 4.13: Señales de las Variables de Control con Función Switching – Movimiento en Avance

### Prueba 2: Velocidad Negativa

Condiciones iniciales y finales:

$$x_i = 0, y_i = 0, \theta_i = -175^\circ, v_i = 0.001, \eta_{v_i} = 0, \delta_i = 0$$

$$x_f = 2, y_f = 2, \theta_f = -160^\circ, v_f = 0.001, \eta_{v_f} = 0, \delta_f = 0$$

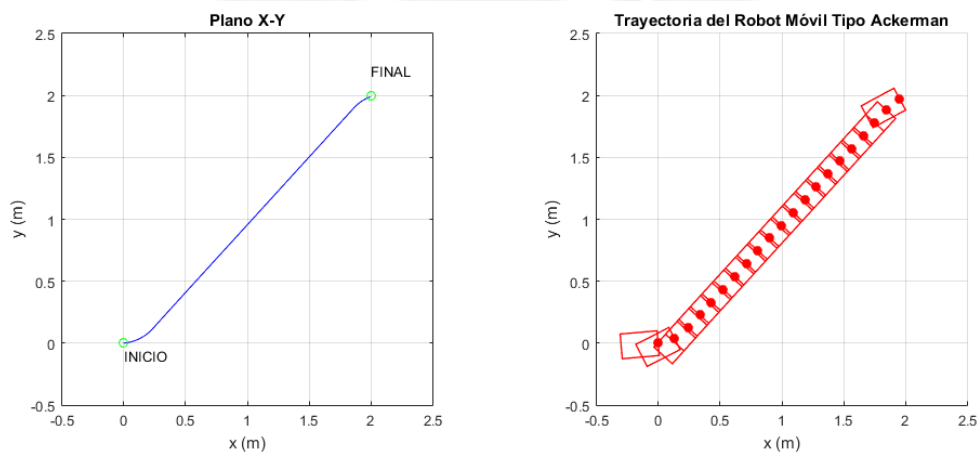


Figura 4.14: Trayectoria con Función Switching – Movimiento en Retroceso

En la figura 4.14 se muestra la trayectoria deseada generada en el lado izquierdo y la trayectoria recorrida por el robot móvil en el lado derecho.

En la figura 4.15 se muestran las señales de las variables de estado y en la figura 4.16 se muestran las señales de las variables de control.

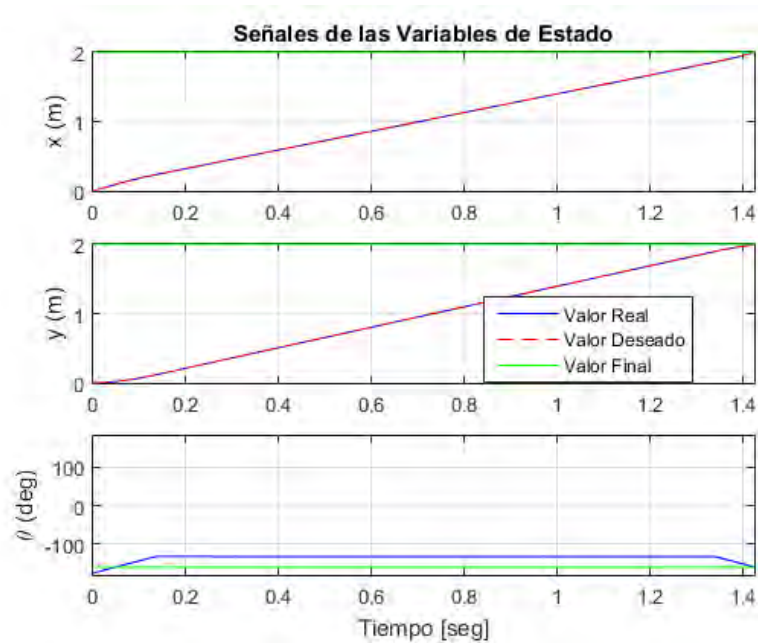


Figura 4.15: Señales de las Variables de Estado con Función Switching – Movimiento en Retroceso

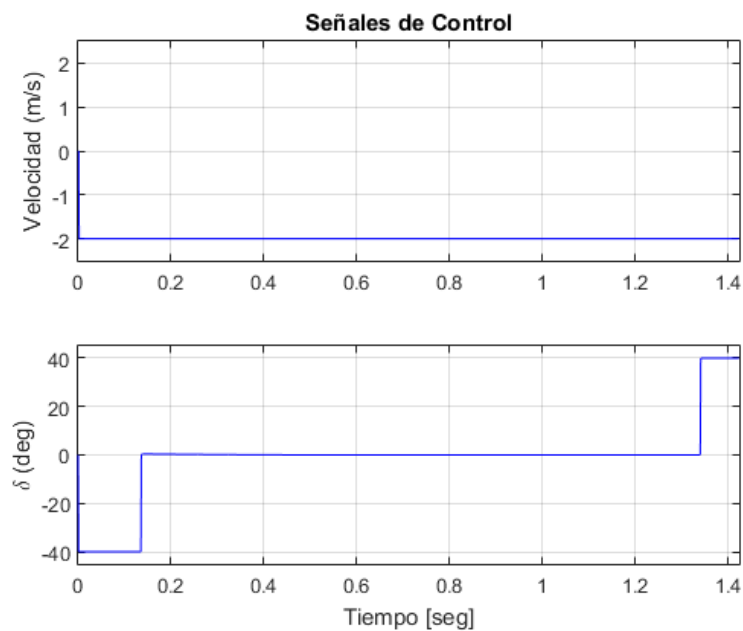


Figura 4.16: Señales de las Variables de Control con Función Switching – Movimiento en Retroceso

### Prueba 3: Cambio de Velocidad

Condiciones iniciales y finales:

$$x_i = 0, y_i = 0, \theta_i = -135^\circ, v_i = 0.001, \eta_{v_i} = 0, \delta_i = 0$$
$$x_f = -2, y_f = -2, \theta_f = 50^\circ, v_f = 0.001, \eta_{v_f} = 0, \delta_f = 0$$

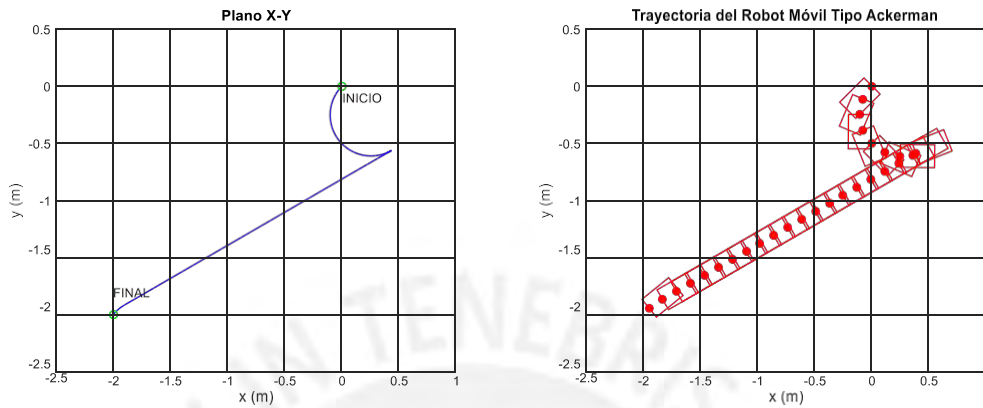


Figura 4.17: Trayectoria con Función Switching – Movimiento en Retroceso

En la figura 4.17 se muestra la trayectoria deseada generada en el lado izquierdo y la trayectoria recorrida por el robot móvil en el lado derecho.

En la figura 4.18 se muestran las señales de las variables de estado y en la figura 4.19 se muestran las señales de las variables de control.

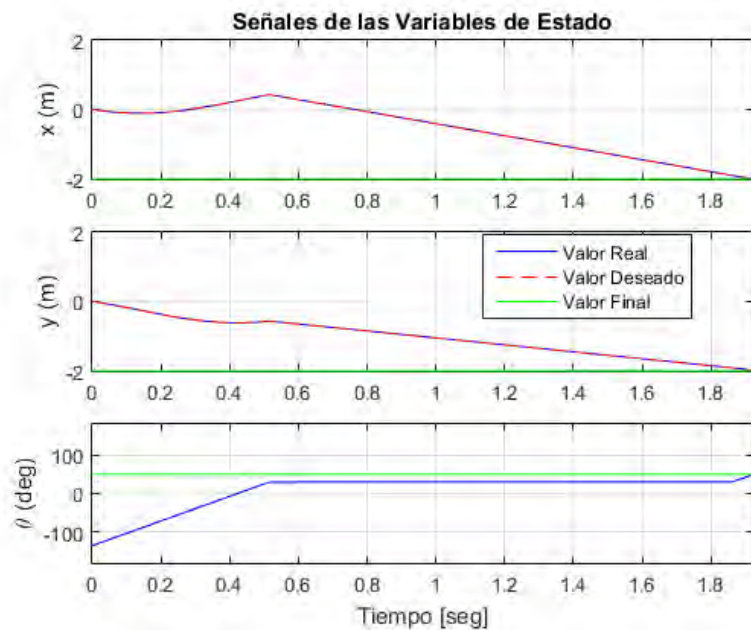


Figura 4.18: Señales de las Variables de Estado con Función Switching – Movimiento Variable

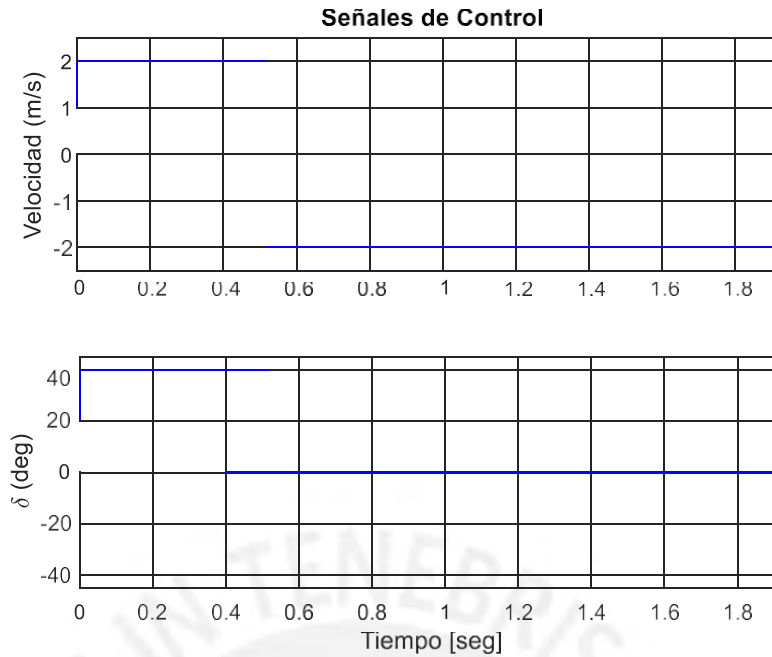


Figura 4.19: Señales de las Variables de Control con Función Switching – Movimiento en Retroceso

#### 4.7. Conclusiones Preliminares

- Se logró obtener un sistema de control aplicado a un sistema no lineal, partiendo de realizar una transformación al modelo del robot móvil para poder controlarlo en base a un controlador lineal.
- La técnica de flatness diferencial, resulta una técnica de transformación muy útil para sistemas que requieren de una trayectoria a seguir por ciertas variables del sistema, ya que la técnica flatness permite trabajar partiendo de variables seleccionadas del sistema que se desea controlar.
- Mediante las simulaciones realizadas para el seguimiento de trayectorias dadas, el sistema de control tiene un buen funcionamiento, permitiendo trabajar en una amplia zona de movimiento del robot móvil.



- La ley de control aplicada al sistema permite generar correctamente el valor de la velocidad, ya sea positivo o negativo, según sea el caso, cuando se realiza la transformación inversa del sistema flatnes.
- Pese a que al realizar la transformación del sistema utilizando la técnica flatness, se llega a una matriz que puede ser singular para una velocidad igual a cero, esta se puede evitar realizando un cambio de la velocidad igual a cero hacia un valor muy pequeño casi despreciable, pero útil para prevenir la singularidad.



## CAPÍTULO 5. DISEÑO DEL SISTEMA DE NAVEGACIÓN PARA EL ROBOT MÓVIL

### 5.1. Introducción

La navegación es la ciencia de calcular la localización y la velocidad de un vehículo en tierra, mar o en el espacio. Dependiendo del objetivo del vehículo, este es equipado con sensores que permiten realizar dicha tarea. Por ejemplo, algunos suelen utilizar un GPS el cual sería suficiente para viajes por tierra o mar, dependiendo del entorno, pero, por ejemplo, para vehículos aéreos se requiere realizar otras mediciones como altitud y velocidad del viento, lo que requiere de otros sensores para dichas mediciones. Sensores como magnetómetros también son utilizados para ciertos trabajos, pero últimamente se vienen utilizando la navegación inercial.

Los sistemas de navegación inercial utilizan los llamados Inertial Measurement Units (IMU's) los cuales son dispositivos microelectromecánicos conocidos como MEMS (MicroElectroMechanical Systems) y que son populares debido a su precisión en las mediciones. Aparte de su bajo consumo, estos sensores no requieren de ningún recurso externo para su medición. Existe 2 tipos de movimiento, traslación y rotacional, por lo que existen 2 tipos de IMU que son los acelerómetros y giroscopios que se usan para estimar el movimiento lineal y rotacional del vehículo a través del espacio de trabajo. (Bekir,2007)

Los acelerómetros y giroscopios son sensores inerciales debido a su principio de funcionamiento y van en mano para realizar la navegación del vehículo. Estos permiten medir aceleración, fuerza de gravedad, velocidad angular, pero que al fusionarlos permiten realizar también otras mediciones como inclinación y altitud del vehículo.

Existe también el llamado Sistema de Referencia de Altitud y Rumbo (AHRS - Attitude and Heading Reference System) que provee información de la referencia vertical (altitud) y la referencia con respecto al norte geográfico, que es usado mayormente para vehículos aéreos pero que provee información útil como el ángulo de rumbo para otros tipos de vehículo. Este tipo de sistema involucra, de igual manera, los sensores inerciales mencionados previamente, y además se utiliza un sensor representativo en este tipo de sistema que es el magnetómetro que trabaja

en base a los campos magnéticos de la tierra. Más adelante, se mencionará el motivo y ventaja del uso de cada tipo de sistema con respecto al otro.

En el tema de navegación se trabaja mucho con los ángulos orientación, llamados también ángulos intrínsecos o ángulos de Euler, los cuales vienen a ser los ángulos descritos por el movimiento rotacional de un cuerpo sobre un eje de coordenadas, siendo estos ángulos definidos como:  $\theta$  (*roll*),  $\phi$  (*pitch*) y  $\psi$  (*yaw*), como se muestran en la figura 5.1.

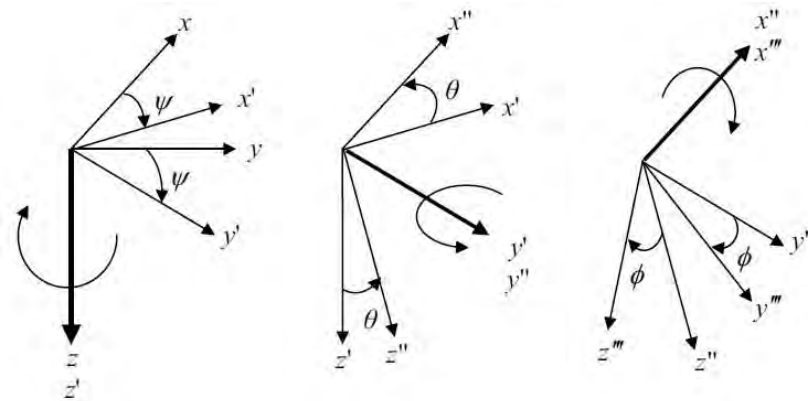


Figura 5.1: Ángulos de Rotación en un Sistema de Coordenadas (Bekir,2007)

Se muestran como cada rotación corresponde a un eje de coordenada siendo:  $\phi$  para el eje X,  $\theta$  para el eje Y,  $\psi$  para el eje Z; que visualizándolos en un robot móvil se tiene de la siguiente manera:

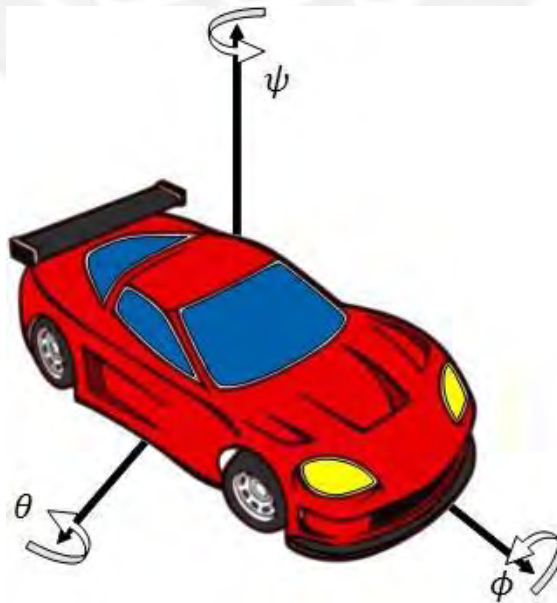


Figura 5.2: Ángulos de Rotación de un Robot Móvil

## 5.2. Sensores Inerciales

### 5.2.1. Acelerómetro

El acelerómetro es un dispositivo electromecánico que permite medir la fuerza de la aceleración que se genera en los ejes  $x$ ,  $y$ ,  $z$ . La aceleración que mide el dispositivo puede ser estática, como la fuerza de gravedad que se genera en un cuerpo en total reposo, o puede ser dinámica causada por el movimiento que se le realiza al acelerómetro.

Existen varias formas mediante las cuales están diseñados los acelerómetros. El método más típico utiliza el efecto piezoeléctrico que es la habilidad de algunos materiales de generar un voltaje eléctrico en respuesta a un estrés mecánico aplicado, por lo que el acelerómetro contiene una estructura cristalina microscópica al cual se le aplica el estrés mecánico.

Imaginar una esfera suspendida dentro de una estructura, a la cual no se le ejerce ningún movimiento ni tiene efecto de alguna fuerza externa no se produce ninguna aceleración en la esfera, por lo que el sensor dará como valores medidos el valor de cero en todos los ejes. Sin embargo, para un caso donde ya se involucra fuerzas externas y/o aceleraciones, el sensor arrojará las mediciones en cada caso.

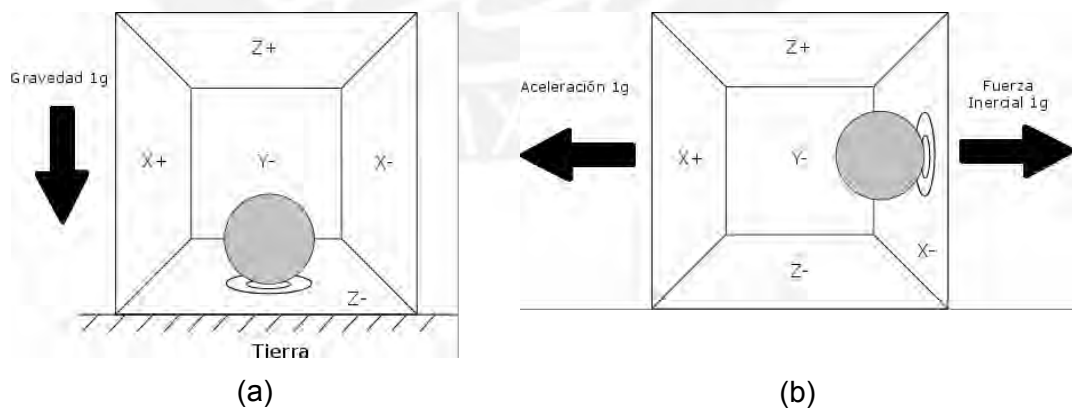


Figura 5.3: Medición del Acelerómetro en Base a una Fuerza Generada

Como se observa en la figura 5.3, en la imagen (a), el sensor ubicado sobre una superficie y sin movimiento alguno, debido a la presión de la esfera sobre la pared

del sensor generada por la fuerza de gravedad se genera una medición en el eje z dando como mediciones del sensor:  $X = 0$  ,  $Y = 0$  ,  $Z = -1g$  ; mientras que en el caso de la imagen (b) el sensor solo detecta un movimiento externo dirigido hacia la izquierda lo que fuerza a la esfera hacia la pared derecha y genera una fuerza en ese sentido y el sensor mide la aceleración que ahí se genera que sería:  $X = -1g$  ,  $Y = 0$  ,  $Z = 0$ .

Algunos acelerómetros pueden tener sus mediciones especificadas en  $g/s$  (gravedad =  $9.8 \text{ m/seg}^2$  ) o puede estar especificada directamente en  $\text{m/seg}^2$  ; aparte, depende de cada fabricante que tipo de interface tiene cada acelerómetro para realizar las mediciones de la aceleración; proporcionando así las aceleraciones en los 3 ejes de coordenadas:  $a_x, a_y, a_z$

### 5.2.2. Giroscopio

El giroscopio es un dispositivo que mide el movimiento rotacional o la velocidad de rotación alrededor de los ejes  $x$ ,  $y$ ,  $z$  utilizando el principio de conservación del momento angular.

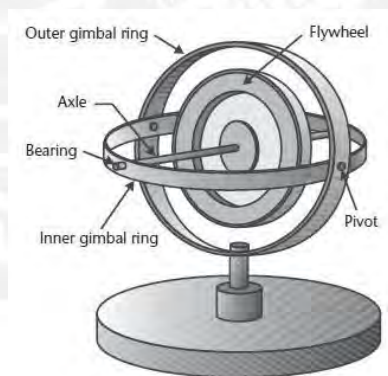


Figura 5.4: Giroscopio Mecánico Convencional (Woodman,2007)

Últimamente, se vienen usando principalmente los giroscopios vibracionales que sensan la velocidad angular a partir de la fuerza de Coriolis aplicado a un elemento vibracional. Es por esto que la precisión por la que se mide la velocidad angular puede variar dependiendo del material y la estructura que este diseñado el giroscopio. Los materiales típicos usados para el diseño de giroscopio son cristal o cerámica referido a los tipos piezoeléctricos.

Los giroscopios suelen tener sus mediciones especificadas generalmente en  $^{\circ}/seg$  y otros en  $rad/seg$ , lo cual depende del fabricante, proporcionando así las velocidades angulares en los 3 ejes de coordenadas:  $\omega_x, \omega_y, \omega_z$

### 5.3. Magnetómetro

El magnetómetro es un dispositivo que permite medir el campo magnético inducido en cada eje al cual está sometido el dispositivo por efecto de la tierra. Esto nos permite obtener referencia del norte magnético con lo cual se puede calcular el norte geográfico, dependiendo de la ubicación geográfica del dispositivo, y utilizarlo como referencia para conocer el ángulo de azimut del dispositivo. Este sensor cumple un papel importante cuando se trabaja en AHRS ya que en este tipo de sistema se requiere una referencia del rumbo del vehículo con respecto al norte geográfico, por lo que se utiliza el magnetómetro para calcular el norte magnético y utilizarlo como referencia.

El campo magnético de la tierra puede ser representado como se muestra en la figura 5.5.

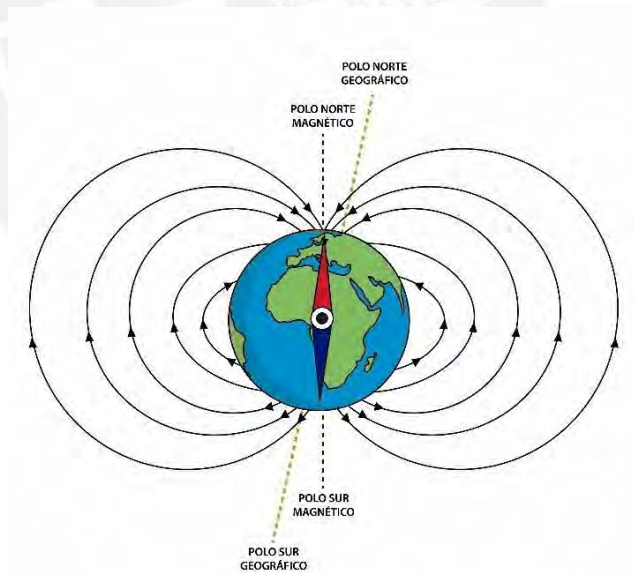


Figura 5.5: Campo Magnético de la Tierra

Utilizando el magnetómetro, conociendo la intensidad de campo que se produce en los 3 ejes, se puede estimar el norte magnético y en base a eso el norte geográfico.

El ángulo de diferencia entre el norte geográfico y el norte magnético se conoce como “declinación”, el cual se representa en la figura 5.6.

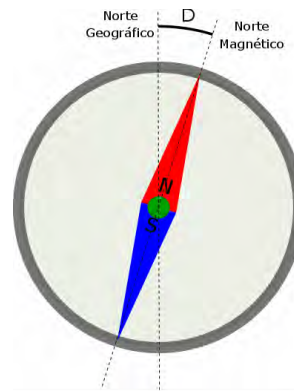


Figura 5.6: Declinación del Norte Geográfico con Respecto al Norte Magnético

Donde el valor de la declinación “D” depende de la ubicación geográfica que se encuentre el dispositivo localizado. Existen muchos links que proporcionan esta información como, por ejemplo, la página web <http://www.magnetic-declination.com/>, la cual permite brindar el valor de la declinación que corresponde a la ubicación en la cual se encuentra un dispositivo dentro del globo terrestre, ya que este valor varía mucho de acuerdo a la zona geográfica. En la figura 5.7 se puede observar un ejemplo de cómo se brinda la información del ángulo de declinación en base a una posición dada.



Figura 5.7: Información de la Declinación

Donde se observa que para la ubicación dada se tiene un ángulo de declinación igual a  $-1^{\circ} 47'$ . Este valor se le suma al ángulo de azimut, y así obtener el ángulo geográfico real.

$$\text{Ángulo Geográfico} = D + \alpha \quad (5.1)$$

El ángulo de azimut viene a ser el ángulo formado entre el norte magnético y el campo magnético en la dirección del vehículo, para lo cual se debe colocar el magnetómetro haciendo coincidir el eje X con la dirección frontal del vehículo, de tal manera que los ejes queden como en la figura 5.8.

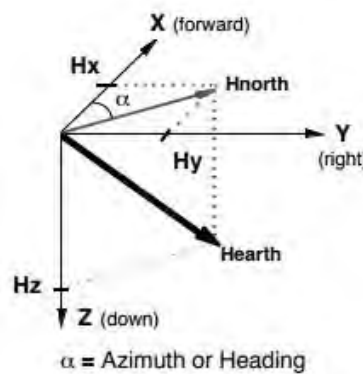


Figura 5.8: Ángulo de Azimut con Respecto a los Ejes del Magnetómetro (Caruso, 1997)

El cálculo del ángulo de azimut con respecto a las mediciones del magnetómetro, está descrita en la ecuación 5.2.

$$\alpha = \tan^{-1} \left( \frac{H_y}{H_x} \right) \quad (5.2)$$

La mayoría de los magnetómetros no suelen estar a un nivel plano o horizontal, ya sea por la estructura de aplicación o debido a que el chip no esté bien soldado a la tarjeta, por lo que existen inclinaciones en los ejes horizontales y pueden originar mediciones inexactas al momento de calcular el ángulo de azimut en base a la ecuación (5.2). En la figura 5.9 se observa las inclinaciones que se pueden generar en los ejes  $x$ ,  $y$ , y que deben ser considerados al momento de calcular el ángulo de azimut.



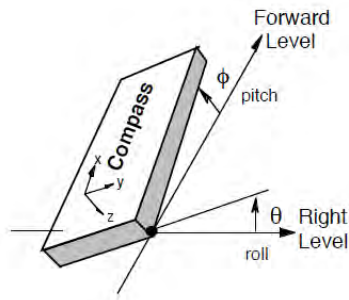


Figura 5.9: Inclinaciones del Magnetómetro en un Plano Horizontal (Caruso,1997)

Para compensar el cálculo del ángulo de azimut considerando las inclinaciones que se generan, se utilizan las siguientes ecuaciones:

$$H_x^* = H_x \cos \phi + H_y \sin \theta \sin \phi - H_z \cos \theta \sin \phi \quad (5.3a)$$

$$H_y^* = H_y \cos \theta + H_z \sin \theta \quad (5.3b)$$

La fórmula para el cálculo del ángulo de azimut estaría dada por:

$$\alpha = \tan^{-1} \left( \frac{H_y^*}{H_x^*} \right) \quad (5.4)$$

#### 5.4. Calibración de los Sensores

Para este sistema se utiliza el módulo MPU-92/65 que se muestra en la figura 5.10:



Figura 5.10: Módulo MPU-92/65 para el Análisis de Movimiento

Es un módulo multi-chip (MCM) compuesta de 2 empaques integrados en un solo empaque. Un empaque contiene un acelerómetro de 3 ejes y un giroscopio de 3 ejes, mientras que el otro empaque contiene el AK8963 que es un magnetómetro

de 3 ejes, lo cual convierte al MPU92/65 en un completo sensor de 9 ejes que combina los 3 ejes del acelerómetro, los 3 ejes del giroscopio y los 3 ejes del magnetómetro. También incluye un Procesador de Movimiento Digital (DMP – Digital Motion Processor) que permite la interconexión para el vfo de datos mediante el protocolo I2C.

El chip contiene 3 conversores análogos digital que permiten la digitalización de los 3 sensores los cuales son enviados por el protocolo I2C. Permite también programar, de acuerdo al uso, el rango que tendrá cada sensor. Esto es muy útil debido a que podemos definir los límites de mediciones que tendrá el sensor, siendo así que para cada sensor tenemos las siguientes escalas programables:

- Acelerómetro:  $\pm 2g, \pm 4g, \pm 8g, \pm 16g$
- Giroscopio:  $\pm 250, \pm 500, \pm 1000, \pm 2000 \text{ }^\circ/\text{seg}$
- Magnetómetro:  $\pm 4800 \mu T$

Antes de utilizar y realizar los cálculos de posición y orientación utilizando el sensor, se debe realizar la calibración de los sensores debido a que probablemente el chip no se encuentre a un 100% en una posición horizontal con respecto a nuestro vehículo, esto debido a que el chip al ser soldado en el módulo tenga un error de contacto con la tarjeta y esto genere error en las mediciones. También, porque al colocar el módulo en nuestro vehículo, no se garantiza tener una completa precisión de estar colocando el sensor de manera horizontal con el vehículo.

Esta calibración se realiza en el chip modificando los registros de offset para cada sensor. Este offset se conoce también como “bias” que viene a ser en los sensores la diferencia entre el valor medido y el valor real, el cual se puede compensar con la calibración, pero a la larga genera un pequeño error acumulativo que más adelante mostrará su efecto al momento de estimar la posición y orientación.

De igual manera, se requiere generar una calibración para tener los sensores con valores nulos cuando el módulo se encuentre en un estado horizontal. Para el caso del acelerómetro y giroscopio el método de calibración es el mismo, mientras que para el magnetómetro es un proceso distinto. El algoritmo de calibración para cada sensor se encuentra en el anexo B.1.

La calibración para el caso del acelerómetro y del giroscopio, se basa en tomar cierta cantidad de muestras de las mediciones, luego filtrarlas y realizar un promedio de los datos y actualizar el offset, y seguir el proceso hasta que las mediciones sean las adecuadas.

- **Calibración del Acelerómetro**

En la figura 5.11 se tienen las mediciones de cada eje del acelerómetro en un estado de reposo sin calibración.

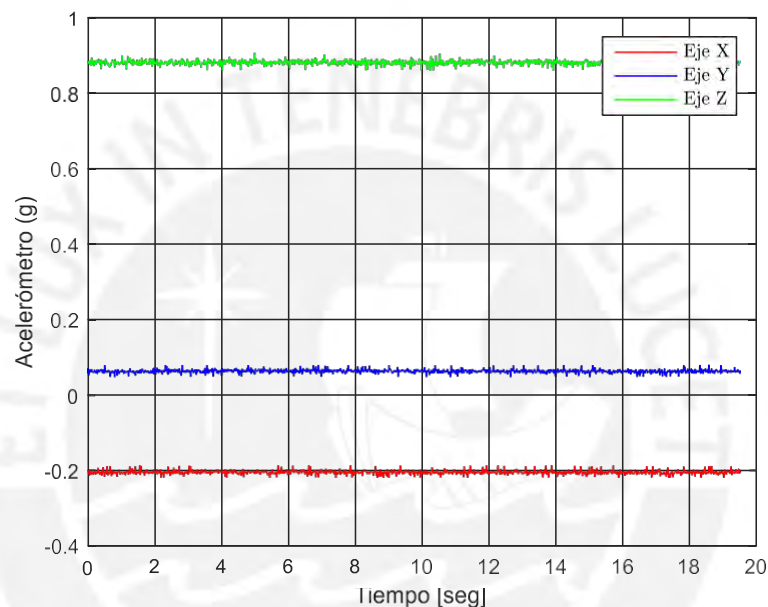


Figura 5.11: Mediciones del Acelerómetro sin Calibrar

Cuando el acelerómetro se encuentra en reposo, puesto en una superficie plana, la única medición que debe existir es de la gravedad con un valor de  $1g \approx 9.8 \text{ m/seg}^2$ .

La calibración del acelerómetro se realiza modificando los registros de offset con los valores adecuados, cuyos valores se pueden estimar haciendo un lazo de prueba y error hasta conseguir los valores de las mediciones deseados para el acelerómetro, o también utilizando unos cálculos recomendados según el fabricante.

Los valores de las mediciones que debe tener el acelerómetro en estado de reposo deben ser:  $a_x = 0$  ,  $a_y = 0$  ,  $a_z = 1g$  .Realizando la calibración se obtiene la data mostrada en la figura 5.12.

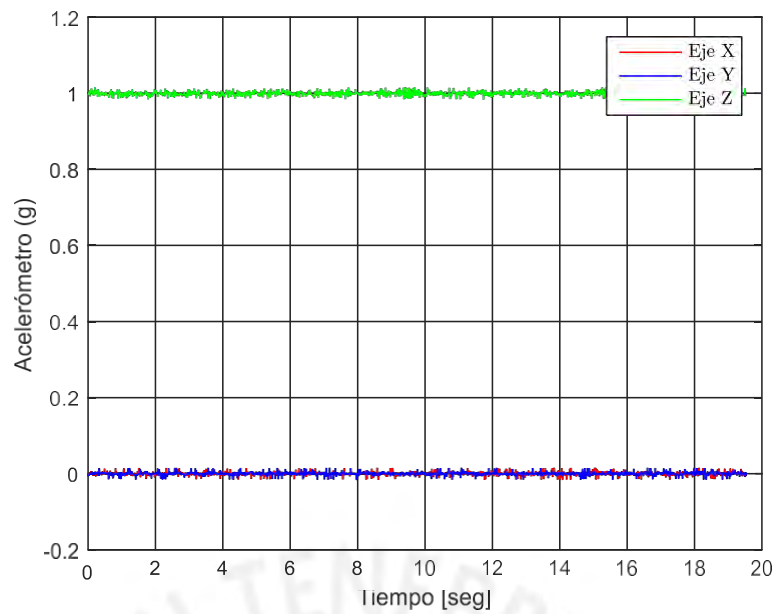


Figura 5.12: Mediciones del Acelerómetro Calibrado

- **Calibración del Giroscopio**

En la figura 5.13 se tienen las mediciones de cada eje del giroscopio en un estado de reposo sin calibración

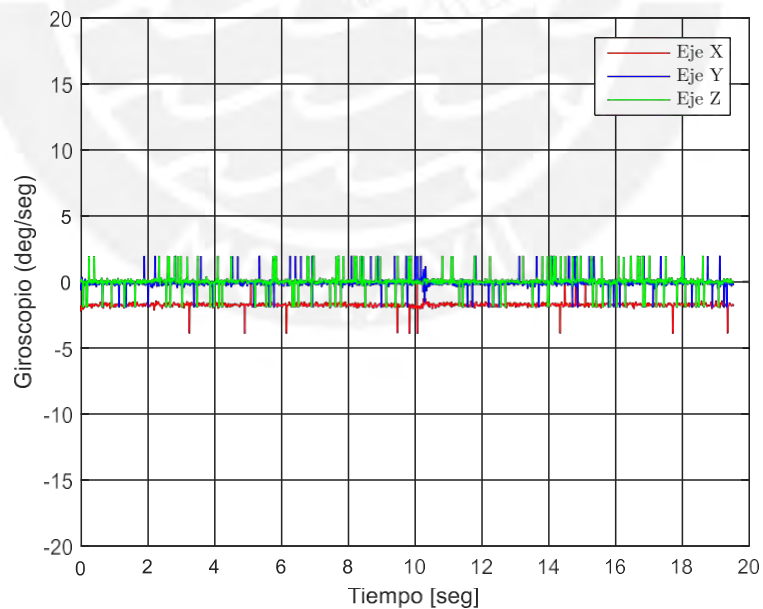


Figura 5.13: Mediciones del Giroscopio sin Calibrar

Al igual que el acelerómetro, para calcular el offset, que permita generar los valores deseados de las mediciones del giroscopio en estado reposo, se puede utilizar el

mismo método empleado en el acelerómetro, donde los valores deseados del giroscopio en estado de reposo deben ser:  $g_x = 0$ ,  $g_y = 0$ ,  $g_z = 0$

Realizando la calibración se obtiene la data mostrada en la figura 5.14.

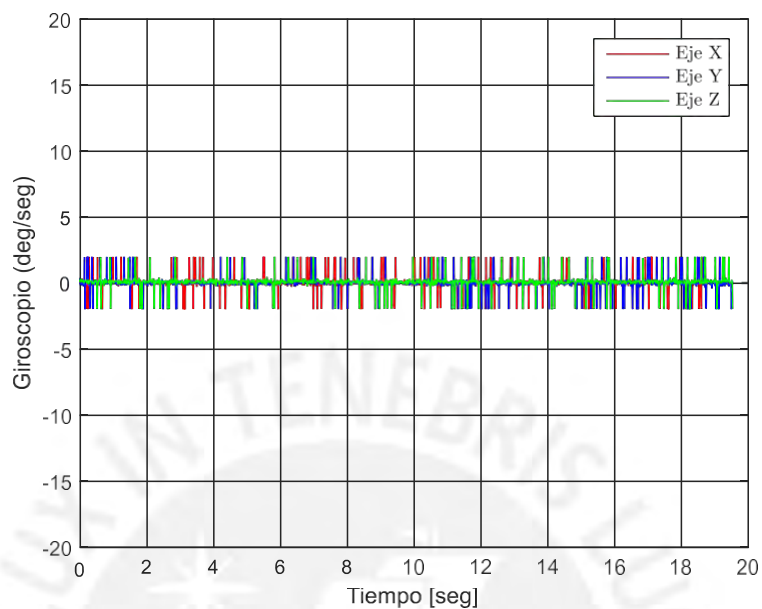


Figura 5.14: Mediciones del Giroscopio Calibrado

- **Calibración del Magnetómetro**

En la figura 5.15 se muestran las mediciones de cada eje del magnetómetro sin calibración.

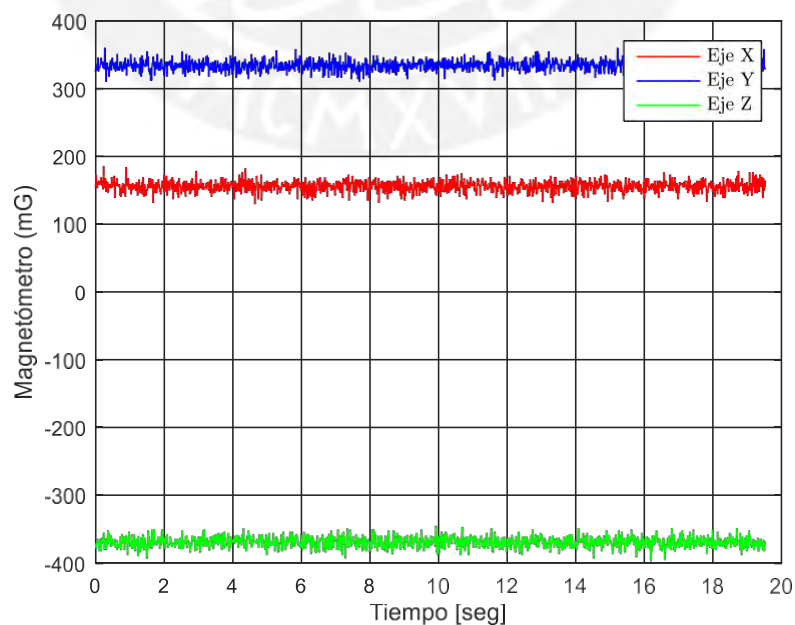


Figura 5.15: Mediciones del Magnetómetro sin Calibrar

Al igual que en los acelerómetros y giroscopios, el magnetómetro también tiene errores en las mediciones, ya sea el caso como error al momento de soldar el chip y el eje horizontal no coincida con el de la superficie que se trabajará o también mayormente por tema de distorsión. La calibración del magnetómetro para temas del eje horizontal es fácil, ya que el chip tiene registros que facilitan la rápida calibración. Mientras que, para el tema de distorsión, que es lo resalta más el error en las mediciones, se separan en 2 categorías denominadas hierro fuerte y hierro suave (hard iron y soft iron). Las distorsiones de hierro fuerte se producen debido a objetos que producen campo magnético como cualquier objeto magnetizado presente. Las distorsiones de hierro suave son consideradas como refracciones o alteraciones en el campo magnético existente. De estos 2 tipos de distorsiones, el que tiene un mayor efecto en el error de las mediciones es el de hierro fuerte (Caruso,2000).

Para poder realizar la calibración de las distorsiones, la forma más práctica de visualizar el error es generando gráficas 2D en base a las mediciones de los ejes de los magnetómetros, esperando que estas gráficas tengan una forma esférica y centrada en el origen de coordenadas. En la figura 5.16 se muestra las gráficas 2D del magnetómetro para cada juego de coordenadas.

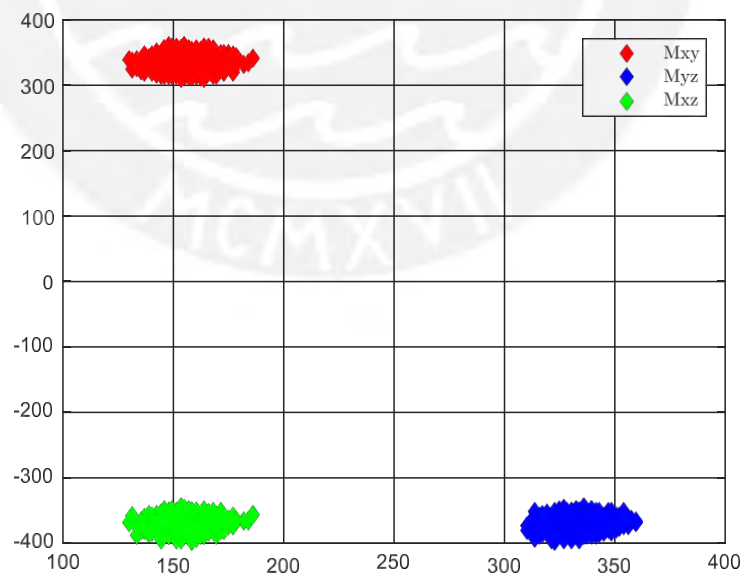


Figura 5.16: Mediciones del Magnetómetro sin Calibración 2D

Se ve que las esferas no están bien centradas es por eso que se aplica la corrección de error para el hierro fuerte y hierro suave. El criterio para la corrección de la distorsión es tomar cierta cantidad de muestras del magnetómetro, mientras que el sensor es movido lentamente en una forma como un ocho, y así ir rastreando los valores máximos y mínimos medidos el sensor en cada eje.

La calibración para el hierro fuerte hará que la esfera formada por las mediciones tenga un centrado en el origen de coordenadas, mientras que para el hierro suave hará que se equilibre la respuesta entre los 3 ejes de manera simétrica. Realizando esta calibración se obtienen los resultados mostrado en la figura 5.17.

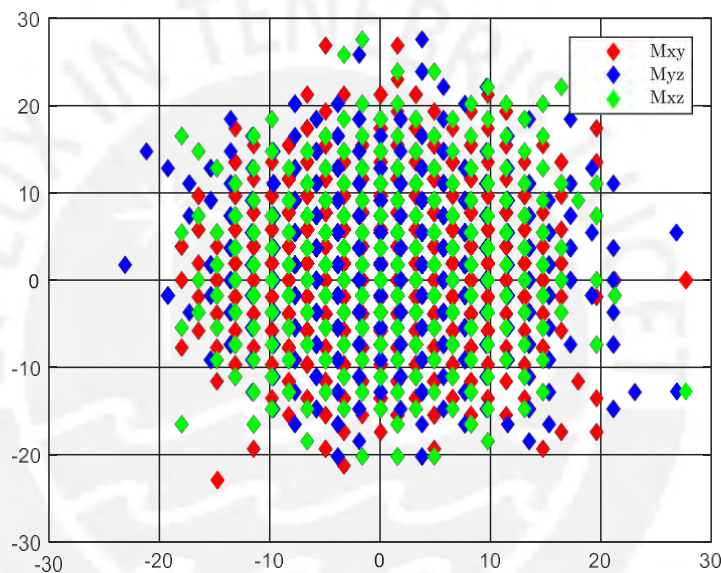


Figura 5.17: Mediciones del Magnetómetro Calibrado 2D

## 5.5. Encoder

El encoder es un dispositivo de sensado que permite convertir movimiento en una señal eléctrica y es utilizado en la gran mayoría de casos para estimar velocidad, posición o dirección. El encoder utiliza diferentes tipos de tecnología para crear la señal que se procesa para realizar las estimaciones, estos tipos pueden ser: mecánico, magnético, resistivo u óptico, siendo esta última la más utilizada, ya que provee señal basándose en la interrupción de la luz del sensor. El encoder óptico es llamado también sensor de posición angular siendo su esquema general el mostrado en la figura 5.18.

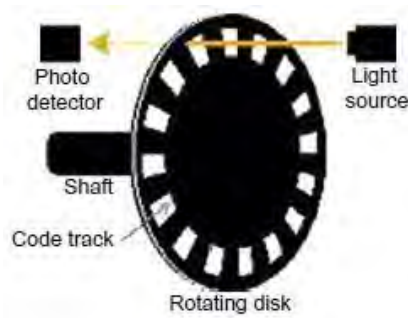


Figura 5.18: Esquema General de un Encoder Óptico (Tzafestas,2013)

El esquema del encoder óptico consta de 4 partes principales:

- Una fuente de luz que se mantiene prendida constantemente.
- Un disco de “n” numero de orificios separados de manera equidistante, el cual bloqueara o dejara pasar la luz de acuerdo a como se mueva el disco.
- Un foto-detector que recibirá la luz emitida por la fuente y generará una señal eléctrica de acuerdo a la intensidad de luz que le llegue.
- Tarjeta electrónica que estará compuesto de un circuito que analizará la señal eléctrica emitida por el foto-detector y emitirá una señal lógica que será recibida por el procesador.

Cabe mencionar que el utilizar un encoder óptico permitirá estimar el movimiento, pero únicamente en un sentido, ya que la salida del sensor enviará una señal lógica, siendo un “1” cuando la luz llega al receptor y un “0” cuando el disco bloquea la luz, esto solo nos permitirá, en base al tiempo de conmutación de la señal lógica y con la cantidad de agujeros que tenga el disco, saber la velocidad que se mueve el disco. Para saber el sentido del movimiento del disco se propone una medición en cuadratura, el cual está compuesto por 2 fuentes de luz y 2 receptores colocados a ciertas distancias, de tal manera que, generen señales desfasadas entre ambas, así como se muestra en la figura 5.19.



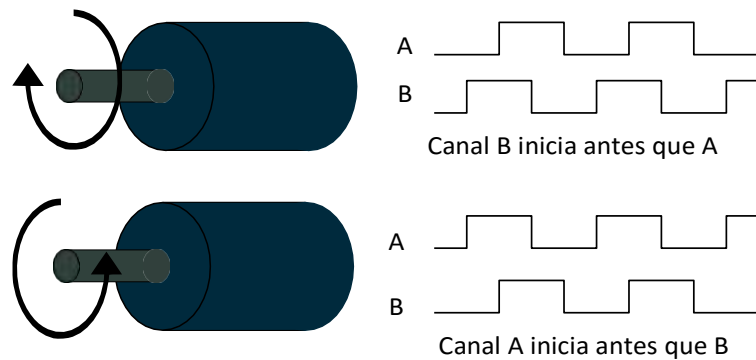


Figura 5.19: Señales del Encoder en Cuadratura

En esta figura se muestra que obteniendo la señal de ambos receptores y analizando la fase que tienen, se puede conocer el sentido con el que gira el motor.

El disco cuenta con una cantidad de agujeros que varía según la aplicación y de esta manera se puede estimar la velocidad realizando el siguiente cálculo:

$$rpm = \frac{60}{n * \Delta t} * pulsos \quad (5.5)$$

Dónde:

- $n$ : número de pulsos por vuelta
- $\Delta t$ : tiempo de muestreo
- $pulsos$ : cantidad de pulsos medidos en un intervalo de tiempo

La conexión que tendrá el encoder en el robot móvil será direccionada hacia el motor de las ruedas de tracción. De esta manera, podremos estimar la velocidad de giro de las ruedas del motor, que sería la velocidad con la cual avanza el robot móvil, en base al sistema óptico del encoder, donde se tendría una conexión al motor de las ruedas parecido a lo que se muestra en la figura 5.20.

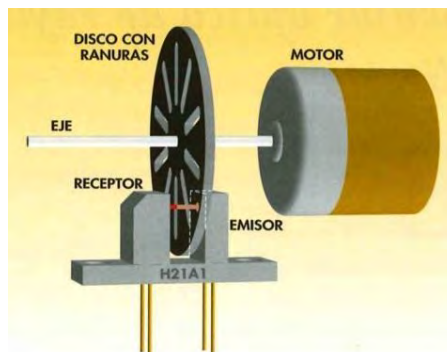


Figura 5.20: Conexión del Encoder al Motor

Fuente: <http://carolina-sp.blogspot.pe/2014/11/contador-de-pulsos-de-encoder-con.html> (Última revisión: 10/08/2017)

## 5.6. Fusión de Sensores Utilizando Filtro de Kalman

Los datos que se busca calcular para el sistema de navegación son: posición, y orientación del vehículo, siendo estos valores los que permitirán realizar correctamente el lazo de control del robot móvil. Estos datos pueden ser estimados teóricamente de manera directa de los sensores.

La posición puede ser calcula como la doble integración de la aceleración:

$$p = \int \int a dt \quad (5.6)$$

Para lo cual, de los varios métodos conocidos, se suele utilizar la regla del trapecio, que define para la velocidad:

$$v_{t2} - v_{t1} = \int_{t1}^{t2} a(t) dt \approx \frac{1}{2}(a_{t1} + a_{t2})\Delta t \quad (5.7)$$

Y de igual forma volviendo a aplicar el método para calcular la posición:

$$p_{t2} - p_{t1} = \int_{t1}^{t2} v(t) dt \approx \frac{1}{2}(v_{t1} + v_{t2})\Delta t \quad (5.8)$$

Y conociendo los valores previos de la velocidad y posición, se puede calcular los valores actuales de cada uno.

Para el caso de la orientación, siendo este el ángulo de giro con respecto al eje z, es posible calcular teóricamente la orientación utilizando el giroscopio, ya que integrando el valor medido en el eje z, se obtiene el valor de la variación en el ángulo y conociendo su valor anterior se puede calcular el ángulo en cada instante de tiempo (Woodman,2007):

$$\theta_{t+1} = \theta_t + \int \omega_z dt \quad (5.9)$$

Como se mencionó, estos métodos funcionan de manera teórica y tal vez cuando se trabajan con sensores muy precisos. Pero en el caso del acelerómetro y giroscopio, pese a haber realizado la calibración de cada uno, existen un offset llamado error *bias*, que es típico en este tipo de sensores y que se observa cuando el sensor esta en reposo, ocasionando que el sensor de un valor diferente al real y que al momento de integrar las medidas generen un error acumulativo en el tiempo,

así como un error en las estimaciones de los valores de posición y orientación. Por lo cual los valores estimados, que deben mantenerse constante y en un valor igual a cero, tienen un valor que crece linealmente o exponencial en el tiempo (Woodman & Harle, 2008; Woodman, 2007).

$$p(t) = \epsilon \frac{t^2}{2} \quad (5.10)$$

$$\theta(t) = \epsilon t \quad (5.11)$$

Siendo la ecuación (5.10) para el caso del acelerómetro y la ecuación (5.11) para el caso del giroscopio, donde  $\epsilon$  es el error acumulativo.

Una forma simple, pero no muy precisa, es tomando un valor promedio de varias mediciones en estado de reposo y utilizar el valor promedio para restarlo de las mediciones y mejorar el cálculo de los datos; otra forma es el método de fusión de sensores, el cual permite utilizar varios sensores para corregir el error y tener un mucho mejor resultado.

Para realizar la fusión de sensores, se utiliza el filtro de Kalman que permitirá utilizar las propiedades de ambos sensores para cada caso de estimación, así como también minimizar el ruido de medición que se genera al realizar la lectura de los datos de los sensores.

### 5.6.1. Fundamentos Teóricos del Filtro de Kalman

El filtro de Kalman es usado para estimar o predecir la siguiente etapa del sistema basado en un promedio de medidas afectadas por ruido, el cual es completamente impredecible. Requiere un modelo entre la entrada y salida del sistema que provea de señales realimentadas, pero que pueda seguir los cambios en la estadística del ruido.

Es esencialmente un conjunto de ecuaciones matemáticas que implementa un corrector de predicción tipo estimador que es óptimo en el sentido que minimiza la covarianza de error de estimación de un vector de estado. Se asume un proceso aleatorio a ser estimado, que puede ser modelado de la siguiente forma (Grover & Hwang, 1997):

$$x_{k+1} = \Phi_k x_k + w_k \quad (5.13)$$

Las mediciones de los procesos se asumen que ocurren en puntos discretos en concordancia con la siguiente relación:

$$z_k = H_k x_k + v_k \quad (5.14)$$

Donde  $w_k$  y  $v_k$  representan señales del ruido del sistema y ruido en la medición respectivamente, y que son considerados como ruido blanco, ruido gaussiano y con covarianza conocida, que están regidas por las matrices de covarianza (Bishop & Welch, 2001; Grover & Hwang, 1997):

$$Q = E[w_k w_k^T]$$

$$R = E[v_k v_k^T]$$

Donde Q corresponde a la covarianza del error del proceso y R corresponde a la covarianza del error de medición.

Se asume que se puede conocer la matriz de covarianza del error asociado a  $\hat{x}_k$  (vector de estado estimado a priori) y se puede definir el error de estimación como:

$$e_k^- = x_k - \hat{x}_k \quad (5.15)$$

Y su matriz de covarianza del error asociada es:

$$P_k^- = E[e_k^- e_k^{-'}] = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T]_k \quad (5.16)$$

En varios casos se inicia la estimación sin conocer mediciones a priori, por lo que se puede considerar que el valor inicial estimado es cero.

Con el cálculo del vector de estado estimado a priori  $\hat{x}_k$ , se busca utilizar la señal medida  $z_k$  para realizar a estimación a posteriori, siendo la ecuación de estimación definida como (Grover & Hwang, 1997):

$$\hat{x}_k = \hat{x}_k + K_k (z_k - H_k \hat{x}_k) \quad (5.17)$$

Dónde:

$\hat{x}_k$  = estimación actualizada

$K_k$  = Factor de Corrección

Por lo que se define el proceso señalado en la figura 5.21 para realizar la estimación del vector de estado:

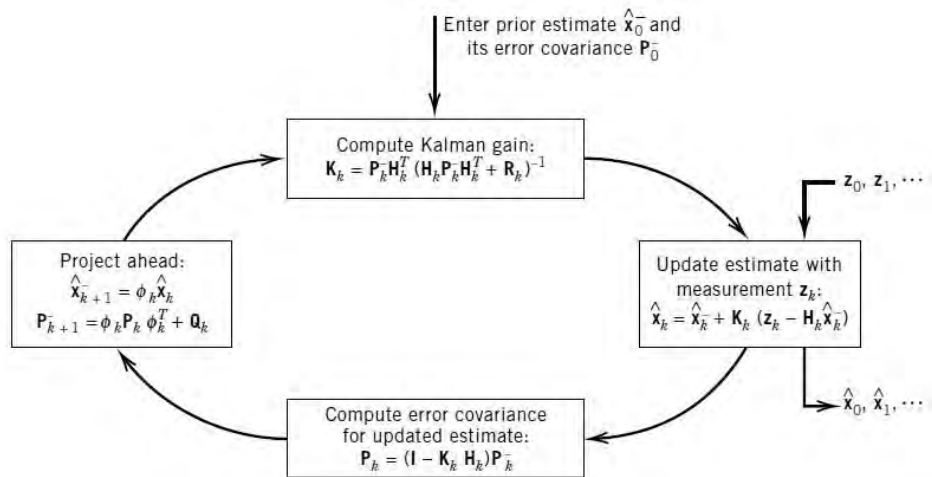


Figura 5.21: Lazo del Proceso del Filtro de Kalman (Grover & Hwang, 1997)

El proceso se puede definir en 2 etapas principales: predicción y corrección, donde en la primera etapa se calcula los valores estimados a priori del vector de estado, así como la matriz de covariancia del error, mediante las siguientes ecuaciones:

$$\hat{x}_{k+1}^- = \Phi_k \hat{x}_k \quad (5.18)$$

$$P_{k+1}^- = \Phi_k P_k \Phi_k^T + Q_k \quad (5.19)$$

Mientras que, en la segunda etapa, se corrige el valor del vector de estado en base al factor de corrección y las mediciones reales del proceso, obteniendo así el vector de estado estimado a posteriori, y luego se actualiza la matriz de covarianza del error, todo esto mediante las siguientes ecuaciones:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \quad (5.20)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-) \quad (5.21)$$

$$P_k = (I - K_k H_k) P_k^- \quad (5.22)$$

### 5.6.2. Estimación del Ángulo de Orientación

El ángulo que interesa estimar es el ángulo de rotación en el eje Z, ya que la rotación en este eje es la que corresponde al ángulo de orientación que guía al vehículo móvil. Para esto se plantea la siguiente ecuación continua del proceso:

$$\theta_z = [0]\theta_z + [1]\omega_z \quad (5.23)$$

Lo cual corresponde a la ecuación del ángulo de rotación en el eje Z y que es directamente observable por lo que se le puede aplicar cualquier observador. Esta ecuación puede ser llevada a la forma continua de estimación para el ángulo de rotación utilizando el filtro de Kalman, obteniendo:

$$\hat{\theta} = [0]\hat{\theta} + [1]\omega_z + L(\theta - \hat{\theta}) \quad (5.24)$$

Donde el valor de  $L$  es la ganancia de corrección que se ha de calcular.

Para realizar la estimación se utilizará las mediciones del giroscopio y las del magnetómetro, obteniendo la velocidad angular del giroscopio y la orientación con respecto al norte magnético del magnetómetro, consiguiendo  $\omega_z$  del giroscopio, mientras que  $\theta$  se debe calcular utilizando la fórmula de la ecuación (5.1).

Para validar la ecuación del filtro de Kalman, se realiza una prueba en Matlab simulando una señal sinusoidal y su derivada que sería la velocidad angular, ambas con ruido, para validar el modelo en la ecuación (5.23) y para corroborar que se tiene una mejor estimación del ángulo de orientación, obteniendo el resultado en la figura 5.22:

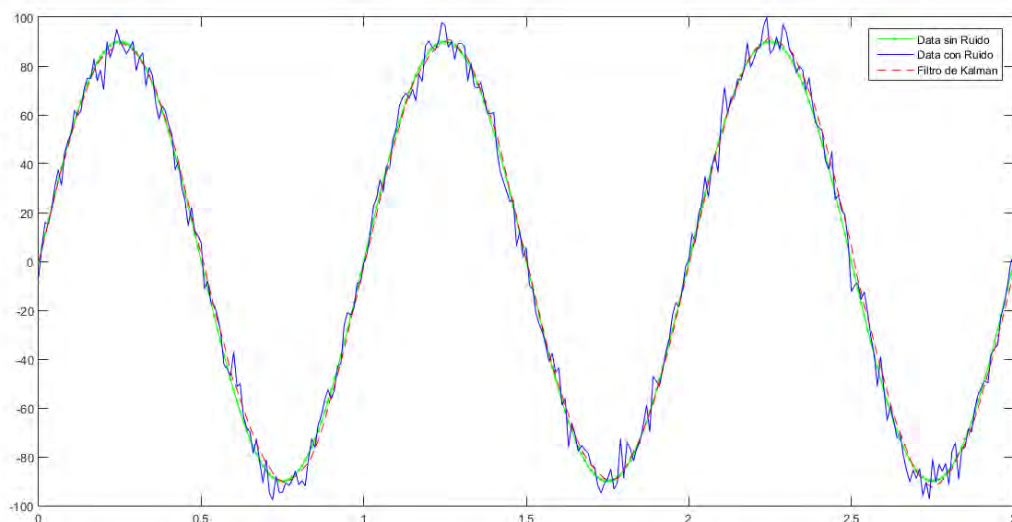


Figura 5.22: Respuesta del Filtro de Kalman para una Señal Sinusoidal

Se observa que el filtro de Kalman logra estimar correctamente la señal y a su vez logra minimizar el ruido generado en la señal medible, lo que conlleva a que está bien diseñado el filtro.

Ahora se realiza una prueba aplicando la fusión de sensores con las mediciones del giroscopio y magnetómetro realizando un movimiento circular, obteniéndose así las mediciones de los sensores en las figuras 5.23 y 5.24.

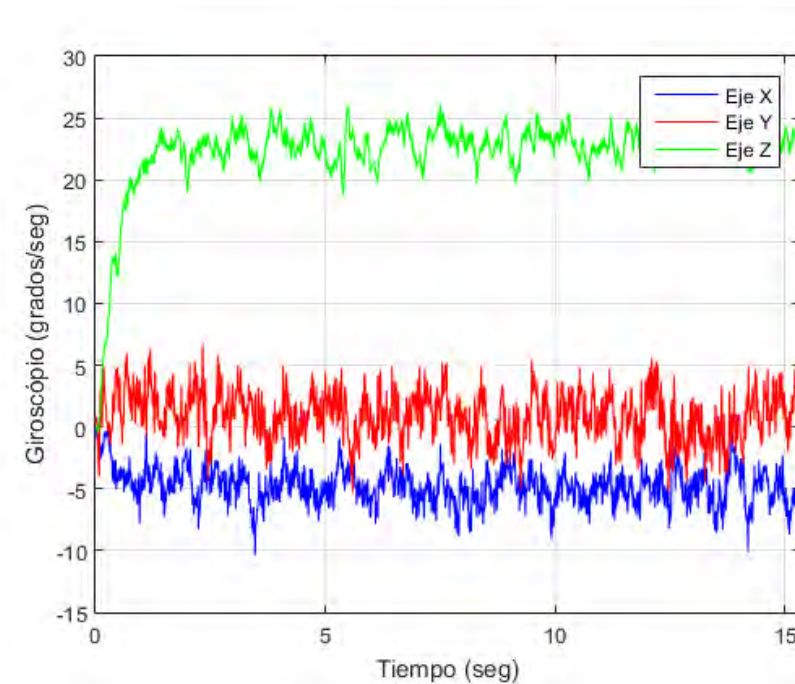


Figura 5.23: Señales de las Mediciones del Giroscopio

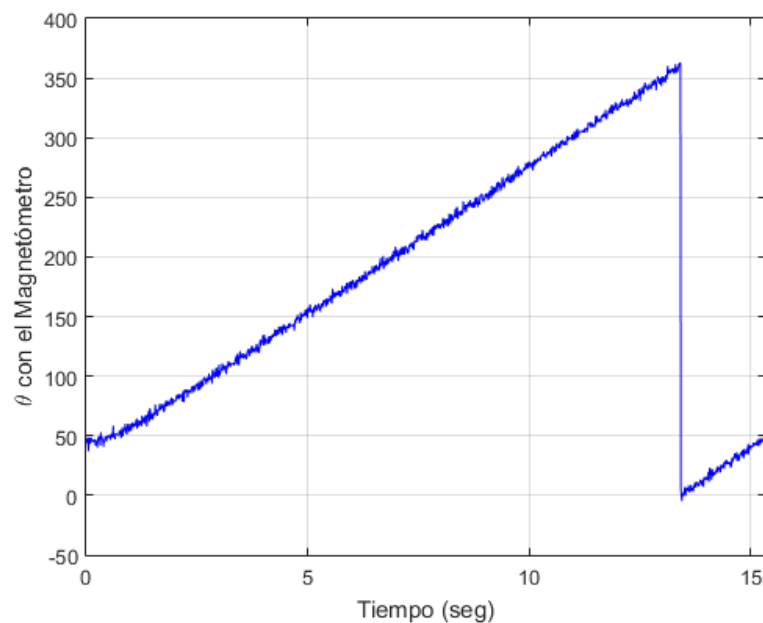


Figura 5.24: Señal del Ángulo de Orientación Calculado con el Magnetómetro

Se puede observar en la figura 5.23 las mediciones obtenidas del giroscopio, notando que se genera una variación en la velocidad angular en el eje Z, obteniendo así la señal  $\omega_z$ ; en la figura 5.24 se observa el ángulo de orientación calculado utilizando el magnetómetro, apreciándose que el sensor partió de una orientación de casi 50° y realiza una vuelta completa, obteniéndose así la señal  $\theta$ .

Con estas 2 mediciones se aplica el filtro de Kalman para corregir el error del ángulo de orientación y minimizar el ruido, obteniendo así la respuesta en la figura 5.25.

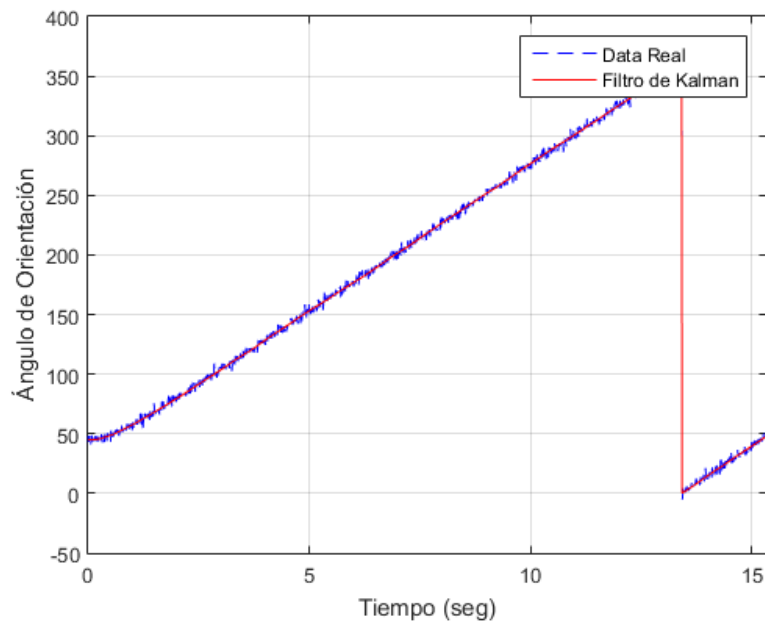


Figura 5.25: Respuesta del Filtro de Kalman

### 5.6.3. Estimación de la Posición

La posición se estima partiendo de la connotación:

$$\frac{d}{dt}x = x \quad (5.25)$$

Llevando la ecuación 5.22 a la forma:

$$\begin{aligned} x &= Ax + Ww \\ y &= Cx \end{aligned} \quad (5.26)$$

Se tiene:



$$\begin{bmatrix} \dot{x} \\ x \end{bmatrix} = \begin{bmatrix} v & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ x \end{bmatrix} + \begin{bmatrix} v \\ 1 \end{bmatrix} x \quad (5.27)$$

$$y = [1 \quad 0] \begin{bmatrix} x \\ x \end{bmatrix}$$

Donde el vector de estado es  $x = [x \quad \dot{x}]^T$  y la salida que interesa es la posición referida por la primera variable del vector de estado  $x$ . Para utilizar la aceleración en mejorar la estimación del vector de estado, esta se considerará como la “perturbación” de tal manera que pueda ser usada como otra entrada al sistema.

Primero se corrobora que el sistema sea observable. Para esto se revisa que el rango de la matriz de observabilidad sea igual a número de estados del sistema, siendo la matriz de observabilidad:

$$Ob = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Siendo el rango de la matriz de observabilidad:

$$\text{rank}(Ob) = 2$$

Siendo igual al número de estados del sistema, lo que permite que se pueda estimar los valores del vector de estado utilizando el filtro de Kalman.

El sistema en espacio estado representado en la ecuación (5.27) proporciona la estimación de posición y velocidad absoluta del vehículo móvil por lo que, para estimar la posición en cada eje, se debe descomponer la posición estimada en ambos ejes de coordenadas y realizar el filtro de Kalman para cada eje.

Utilizando el mismo caso de la trayectoria circular descrita en el caso de la estimación del ángulo de orientación, para este caso utilizaremos datos obtenidos del mismo módulo MPU92/65 donde se tienen los datos del acelerómetro en la figura 5.26 y los datos del encoder en la figura 5.27.

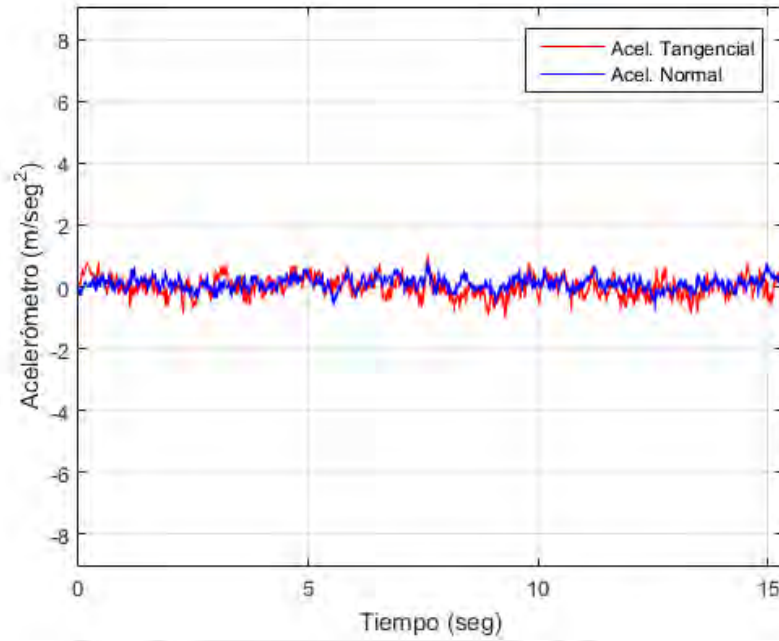


Figura 5.26: Mediciones del Acelerómetro

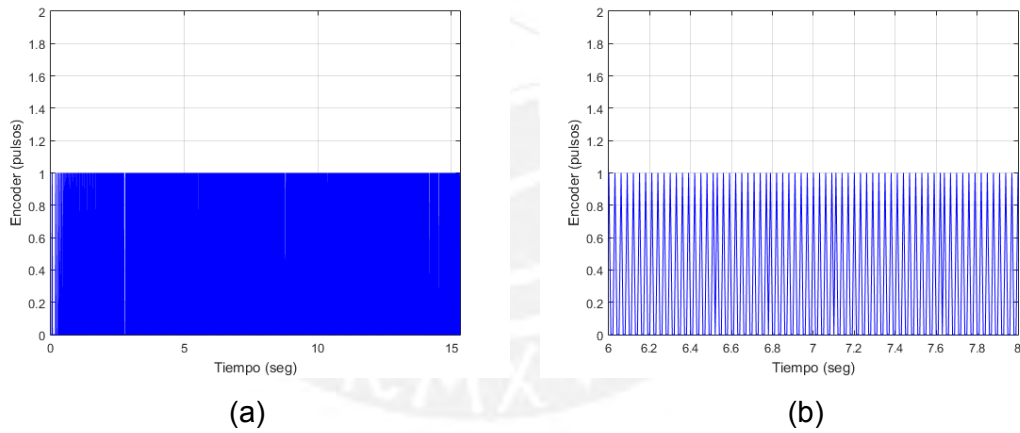


Figura 5.27: Medición del Encoder. (a) Señal Completa, (b) Zoom en un Tramo

Se observa como son las señales tomadas de las mediciones de los sensores, siendo para el caso del acelerómetro, pequeñas variaciones en la aceleración, esto debido a que la prueba se ha realizado con una velocidad constante del vehículo, pero que de igual manera existen aceleración tangenciales y normales debido a la rotación circular. Mientras que en el caso de las mediciones del encoder, esta se visualiza como pulsos, ya que representa la cantidad de pulsos por muestra

realizada, obtenido a partir de esta medición, la variación de distancia recorrida por el robot móvil que, en conjunto con las mediciones del acelerómetro, se puede descomponer las mediciones en los ejes  $x$ ,  $y$  y así saber cuánto ha sido su avance en cada eje.

Utilizando el filtro de Kalman para estimar el trayecto realizado por el robot móvil, mediante la fusión de los sensores acelerómetro y los encoders, se obtiene la trayectoria recorrida como se muestra en la figura 5.28.

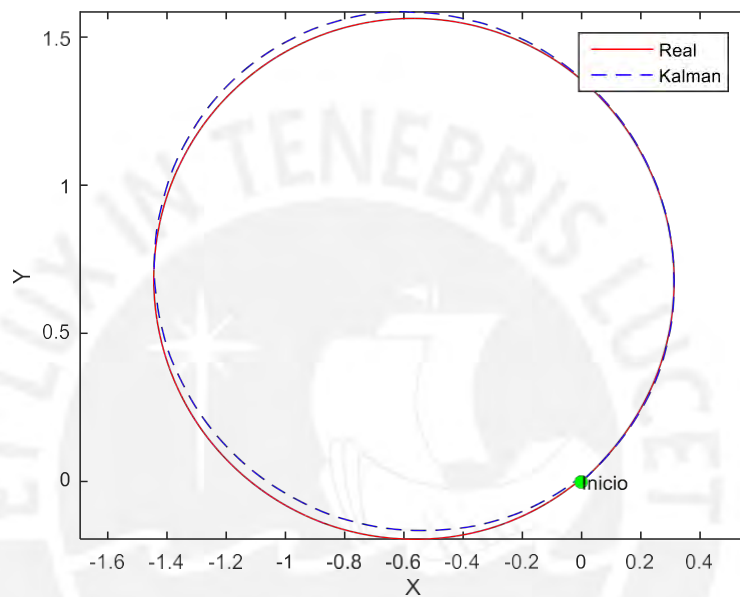


Figura 5.28: Trayectoria Circular del Robot Utilizando Filtro de Kalman

## 5.7. Conclusiones Preliminares

- Se consiguió definir los dispositivos a utilizar que permitieran estimar la posición y orientación del robot móvil dentro de un área y lograr la navegación del vehículo.
- Se mostró el efecto que tiene el error de bias en los sensores inerciales, que es una característica típica de estos y que se deben de considerar al momento de realizar las mediciones.

- Se mostró como se deben realizar las calibraciones de los sensores acelerómetro, giroscopio y magnetómetro, para obtener las mediciones adecuadas.
- Se mostró la importancia de la fusión de sensores utilizando el filtro de Kalman, demostrando que se consigue una mejor estimación de los parámetros del robot móvil usando más de un solo sensor y que se minimiza el ruido de medición.
- Se obtuvieron buenos resultados en las pruebas de mediciones en la posición y orientación del robot móvil, observándose que, pese a que existe un margen de error, se obtiene un resultado aceptable y útil para realizar la implementación.



## **CAPÍTULO 6. IMPLEMENTACIÓN DEL SISTEMA DE CONTROL Y NAVEGACIÓN**

### **6.1. Introducción**

A continuación, se presenta la implementación del sistema de control y navegación en un robot móvil real, lo que permite corroborar la teoría y simulaciones realizadas en los capítulos anteriores aplicados a un prototipo que consta del computador principal, los sensores inerciales, el magnetómetro y los encoders, conectados de manera correcta a las etapas a controlar que son el motor de las ruedas de dirección y a los motores de las ruedas de tracción.

Se pasará a explicar inicialmente como está compuesto el hardware del robot móvil, describiendo los dispositivos, conexiones y su importancia dentro de cada etapa del sistema. También se explicará cómo se ha realizado el programa en C++, resaltando las partes importantes del algoritmo y como se logró tener acceso y control de los periféricos. Finalmente, se mostrarán resultados de pruebas realizadas con el robot móvil para validar la respuesta del sistema, mostrando graficas de la trayectoria del robot en base a los datos almacenados en cada instante de tiempo.

### **6.2. Estructura y Hardware del Robot Móvil**

El cerebro del sistema está compuesto por el Raspberry Pi 3, el cual se muestra en la figura 6.1, que es un mini computador siendo su sistema operativo principal Raspbian pero que permite otros sistemas operativos que son variaciones en base a Linux o también una versión de Windows 10. El raspberry es open-source, por lo que se puede tener acceso directo a todo el hardware de la tarjeta, permitiendo programar sus partes y así poder manipular el acceso a periféricos.

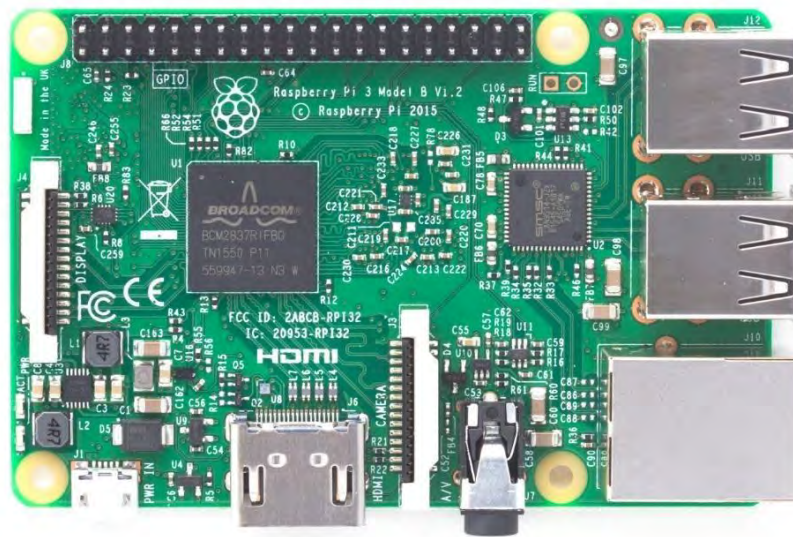


Figura 6.1: Raspberry Pi 3

Dentro de los componentes hardware que tiene el raspberry, las partes que se utilizaran para esta aplicación, se encuentran:

- Puerto Ethernet
- Chip Wi-Fi integrado
- Pines de entradas y salidas digitales
- Pines con voltaje de salida de 5V y 3V

Para realizar la interacción con el raspberry y poder programarlo existen 2 formas: mediante el puerto ethernet o mediante Wi-Fi. La conexión por el puerto ethernet es básica ya que el raspberry cuenta con una dirección IP estática por la cual podemos comunicarnos, así como también se puede realizar mediante Wi-Fi conectando el raspberry y la PC a un modem inalámbrico y esta opción permite interactuar con el raspberry a distancias y libremente.

Para realizar la conexión con los sensores y los motores, se requiere utilizar los pines digitales, dentro de los cuales, hay ciertos pines con funciones especiales que permitirán obtener los datos de manera correcta de los dispositivos externos dependiendo del protocolo de comunicación de cada uno.

Para el sistema de navegación, el raspberry obtendrá los datos del acelerómetro, giroscopio y magnetómetro mediante el protocolo de comunicación I2C, mediante el

cual también se realiza su configuración para la calibración; también se toman las señales emitidas por los 2 encoders para sensor la velocidad, utilizando el módulo FC-03 (figura 6.2).

Estos datos son procesados por el raspberry para luego poder emitir las correctas señales de control al robot utilizando un servomotor (figura 6.3), para el caso de las ruedas de dirección, que se conecta mediante una señal PWM, y utilizando un módulo driver L298 (figura 6.4), para la conexión con el motor de las ruedas de tracción, que se conecta también mediante una señal PWM y 2 pines digitales para realizar el cambio del sentido de giro.



Figura 6.2: Módulo Encoder FC-03



Figura 6.3: Servo SG-5010



Figura 6.4: Módulo Driver L298

En la figura 6.5, se muestra el prototipo diseñado para realizar las pruebas reales, con todos los dispositivos, previamente mencionados, instalados.

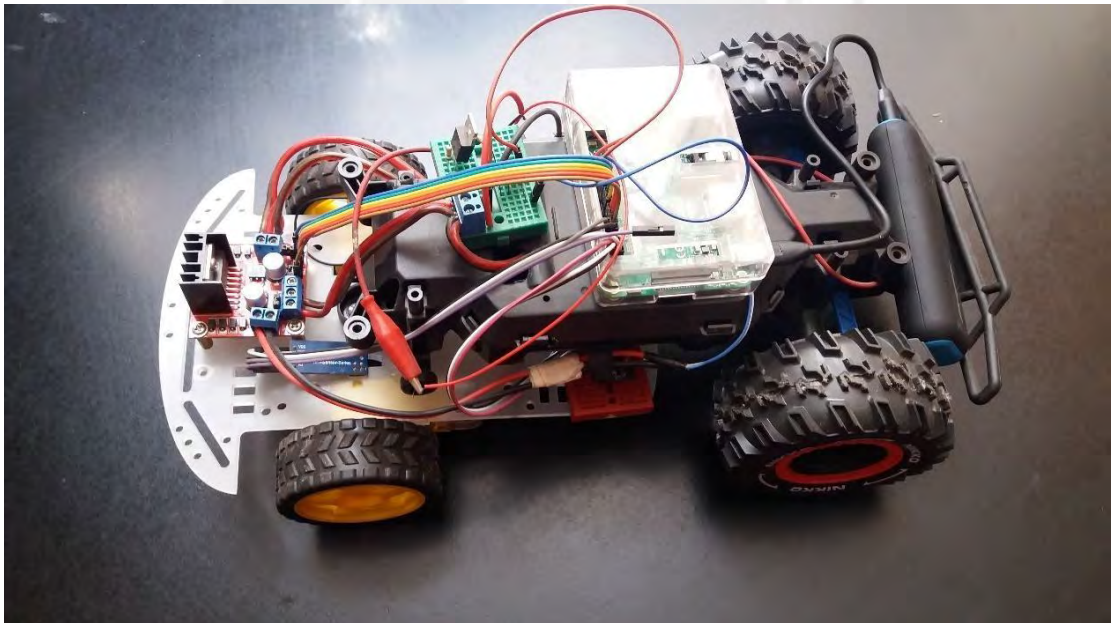


Figura 6.5: Prototipo de un Robot Móvil Tipo Ackerman

### 6.3. Implementación del Código en C++

Lo principal, dentro de la programación del código, es el poder tener control de los pines de entradas y salidas, los cuales permitirán acceder a los sensores y los



motores, para lo cual, existe una librería diseñada para el raspberry, que permite configurar y manejar los pines, la cual se llama *WiringPi* (Henderson,2013). Esta librería ha sido diseñada en C++ y mediante comandos de configuración sencillos, se puede tener gran acceso de configuración a los pines, permitiendo configurarlos como funciones adicionales que tiene cada pin, ya sea el caso de protocolo serial, I2C, PWM, etc.

Para el control de los sensores y de los motores se requieren los siguientes pines:

- ServoMotor: 1 pin en configuración PWM
- Driver Motor DC: 1 pin en configuración PWM y 2 pines de salida digital
- MPU92/65: 2 pines en configuración I2C
- Encoder: 1 pin de entrada digital.

Luego de haber configurado los pines de manera correspondiente, la configuración de los sensores inerciales y del magnetómetro se trabaja en base a los registros del chip integrado de acuerdo a su hoja de datos, mientras que para las señales PWM para el control de los motores basta con ciertos comandos para el envío de la señal.

El proceso que corresponde al algoritmo completo programado en el raspberry, el cual está compuesto por los algoritmos en los anexos B.2 y B.3, se basa en 3 etapas generales:

- Cálculo de Mínima Trayectoria
- Generación del Vector de Puntos
- Control y Navegación

La primera etapa, se basa en recibir los datos de la posición final del robot móvil  $x_f, y_f, \theta_f$ , los cuales serán ingresados desde una PC o dispositivo móvil, con estos datos se ejecutará un sub-programa que se encarga de realizar el cálculo de los puntos  $P_1(x, y)$ ,  $P_2(x, y)$  y del ángulo  $\phi$ , que son los datos correspondientes a una trayectoria óptima como se indicó en el capítulo 3 mediante el análisis de generación de trayectoria basada en arcos de círculos y líneas rectas. Luego de haber calculado dichos datos, estos son almacenados en el archivo *minPath.txt* para ser cargados por el algoritmo de la siguiente etapa.

La segunda etapa, se basa en generar los vectores de puntos correspondientes a las salidas flat y sus respectivas derivadas realizando el proceso indicado en la figura 3.19. Este algoritmo carga el archivo *minpath.txt* obteniendo los datos

necesarios de la trayectoria óptima, los cuales son ingresados a la fórmula de las funciones switching diseñadas en la ecuación (3.34) para la velocidad y en la ecuación (3.35) para el ángulo de dirección, para luego ir estimando los valores de los estados del sistema y almacenar los correspondientes a las salidas flat y utilizar las ecuaciones planteadas en (3.36) y (3.37). al final, se almacena en el archivo *optimalPath.txt*, que va a contener los vectores de las salidas flat y sus respectivas derivadas.

La tercera etapa, se basa en realizar el lazo de control y navegación en tiempo real con el robot móvil. Se utilizan las trayectorias óptimas deseadas haciéndolas ingresar a la ley de control del sistema realimentado y así obtener las señales de control del sistema extendido  $\eta_v$  y  $u$ , para luego realizar la conversión a las variables de control del robot móvil  $v$  y  $\delta$ . Luego, se realizan las lecturas de los sensores, para el sistema de navegación, ingresando las mediciones al filtro de Kalman, como se indicó en el capítulo 5, y de esta manera obtener la posición y orientación real del robot móvil en su entorno. Finalmente, estos datos son utilizados nuevamente en la ley de control y así consecutivamente hasta finalizar los vectores de puntos de las trayectorias deseadas.

Para validar la trayectoria real que ha realizado el robot móvil, los datos de navegación real obtenidos son almacenados en un archivo *realPath.txt* para así poder visualizar la trayectoria real y poder analizar su respuesta.

#### 6.4. Resultados del Sistema de Control y Navegación

Se han realizado 4 pruebas con el robot móvil, en las cuales se ha almacenado la trayectoria deseada generada y la posición del robot móvil en cada instante de tiempo y así ser mostradas en las figuras 6.6, 6.7, 6.8 y 6.9, donde se aprecia que el controlador funciona correctamente llevando al robot a describir trayectorias óptimas, según la trayectoria deseada generada, para llegar a su objetivo.

- Primera trayectoria con condiciones iniciales:  $x_i = 0$ ,  $y_i = 0$ ,  $\theta_i = -140^\circ$  y condiciones finales:  $x_f = 1$ ,  $y_f = 3$ ,  $\theta_f = 170^\circ$

Trayectoria Real del Robot Móvil Tipo Ackerman

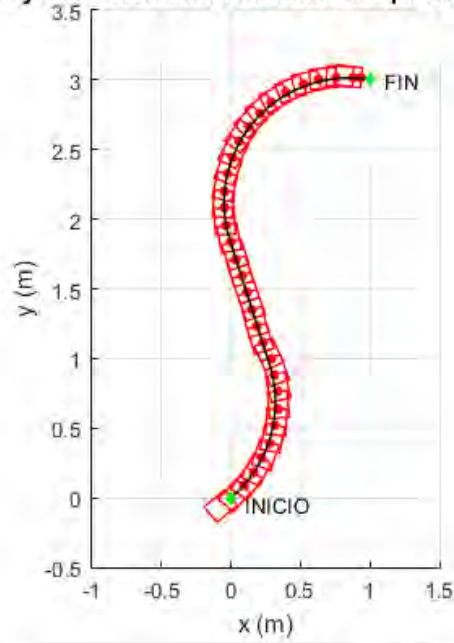


Figura 6.6: Trayectoria Real 1

- Segunda trayectoria con condiciones iniciales:  $x_i = 0, y_i = 0, \theta_i = 90^\circ$  y condiciones finales:  $x_f = 4, y_f = 0, \theta_f = 45^\circ$

Trayectoria Real del Robot Móvil Tipo Ackerman

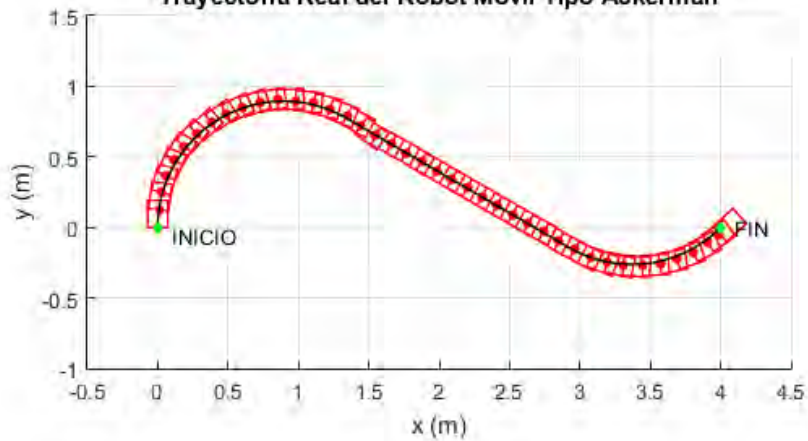


Figura 6.7: Trayectoria Real 2

- Tercera trayectoria con condiciones iniciales:  $x_i = 0, y_i = 0, \theta_i = -135^\circ$  y condiciones finales:  $x_f = 2, y_f = 2, \theta_f = 40^\circ$

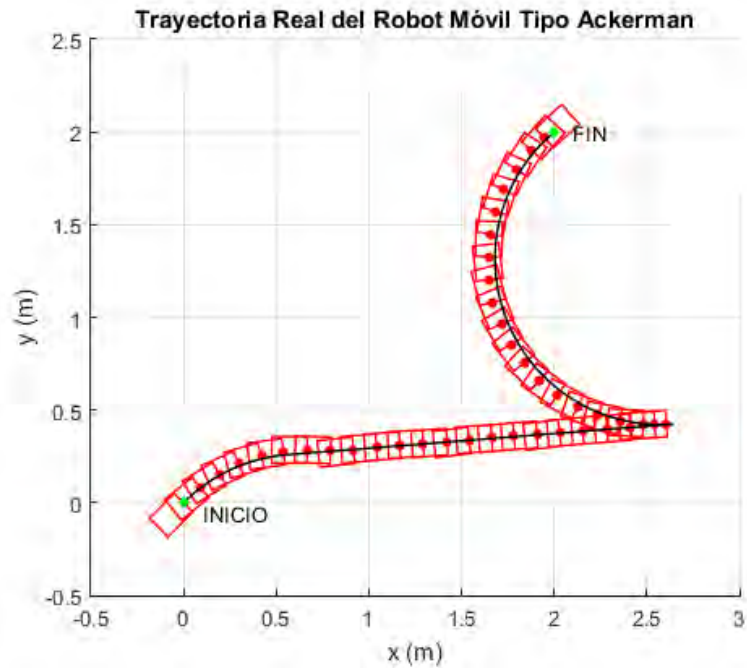


Figura 6.8: Trayectoria Real 3

- Cuarta trayectoria con condiciones iniciales:  $x_i = -3, y_i = -2, \theta_i = 20^\circ$  y condiciones finales:  $x_f = 0, y_f = 0, \theta_f = -160^\circ$

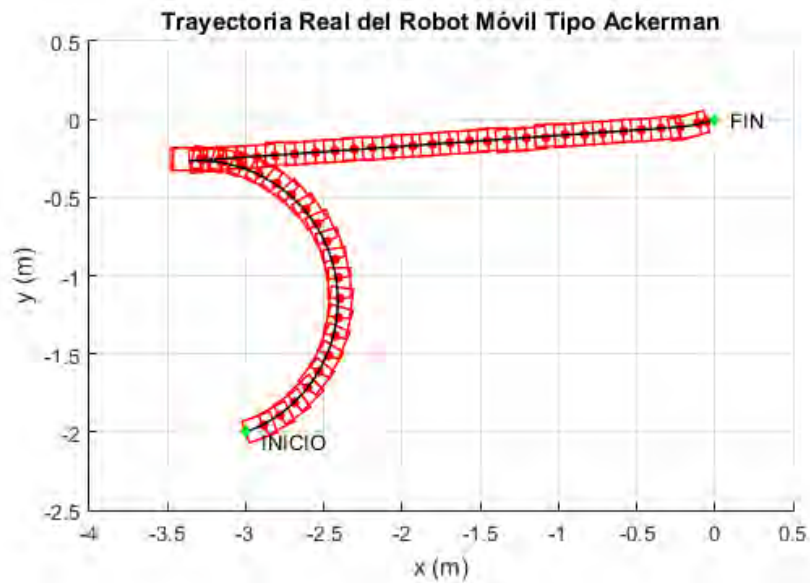


Figura 6.9: Trayectoria Real 4

En las figuras se muestra la trayectoria deseada, dibujada de color negro, y la trayectoria realizada por el robot móvil real, dibujada de color rojo. A continuación, en la tabla 6-1 se presentan los errores entre los valores finales reales y deseados de la posición y orientación

	Error X	Error Y	Error $\theta$
Trayectoria 1	0.0632 m	0.0282 m	4.34°
Trayectoria 2	0.0046 m	0.0442 m	3.33°
Trayectoria 3	0.0453 m	0.0218 m	1.58°
Trayectoria 4	0.0177 m	0.0089 m	1.96°

Tabla 6-1: Errores entre los Valores Finales Reales y los Deseados

## 6.5. Conclusiones Preliminares

- Ha sido posible implementar el sistema de control y navegación, utilizando un robot móvil real equipándolo de los periféricos y sensores necesarios para realizar la demostración.
- Se ha observado que el controlador funciona correctamente, permitiendo dirigir al robot móvil por la trayectoria deseada.
- Se demostró que el robot móvil puede seguir una trayectoria optima basándose en la teoría del diseño de trayectoria basada en líneas rectas y arcos de círculos.
- Se observo que, pese a existir pequeños errores en los valores finales de posición y orientación del robot móvil, el sistema responde de una manera correcta y aceptable.
- En algunas trayectorias se observa la diferencia entre la posición real del robot móvil y la trayectoria a seguir, lo cual se debe a pequeños errores entre las mediciones de los sensores, el cálculo de las coordenadas exactas o por la estructura del robot, pero que no perjudican en general al buen control de este.

## CONCLUSIONES

- Se logró diseñar un sistema de control y navegación que permite trabajar con velocidad variable y, lo más importante, que en una misma trayectoria pueda generar cambios de velocidades para poder trazar la mínima trayectoria posible a su destino.
- Se realizaron pruebas con los métodos de trayectoria planteada, donde se observó que el planeamiento de trayectoria basado en polinomios no es suficiente para que el robot móvil pueda seguir dicha trayectoria, puesto que no se podía saturar el ángulo del timón, pero se pudo solucionar integrando la trayectoria segmentada que a su vez proporciona la ruta más óptima.
- Cada método de planeamiento de trayectoria, tanto polinomial como segmentada, tienen sus ventajas, ya que la primera proporciona velocidad variable y generar velocidades en ambos sentidos, mientras que la segmentada permite trazar rutas óptimas.
- El modelo matemático para el robot móvil tipo Ackerman funciona muy bien cumpliendo con las restricciones dadas, más aún el giro del timón que es muy importante a revisar que se cumplan sus límites.
- La técnica de control planteada de linealización por realimentación utilizando salidas flat permitió un buen control del robot móvil, permitiendo dar la trayectoria deseada al sistema y generando las señales de control correcta para el buen desempeño del sistema.
- Se diseñó un sistema de navegación que permite estimar la posición y orientación del sistema dentro de su entorno de trabajo, utilizando sensores no muy complejos, pero de buena precisión, pero que aun así siempre habrá ruido en las señales.
- Utilizando el filtro de Kalman se pudo estimar correctamente los parámetros mediante la fusión de sensores que permiten medir los cambios que realiza el robot en su trayectoria.

- Se logra minimizar el ruido en las mediciones y estimar un buen valor de las variables del robot, aun así, siempre hay márgenes de error a considerar debido a múltiples perturbaciones que ocurre en el entorno que se mueve el robot móvil.
- Se sugiere realizar el análisis de trayectoria óptima, partiendo de la teoría de control óptimo donde se define una función de costo, demostrando que las funciones switching obtenidas minimizan la función de costo planteada para tiempo mínimo.
- Se sugiere aplicar el criterio de arcos de circunferencias y segmentos de rectas a otros robots móviles como los aéreos o marítimos, considerando de igual manera los ángulos de giros que puedan realizar y su área de movimiento, ya sea en un plano 2D o 3D.
- Se sugiere aplicar la técnica de trayectoria mínima con segmentos de rectas y arcos de circunferencia para evasión de obstáculos, de tal manera que, conociendo la ubicación de los obstáculos se utilice dicha ubicación como un punto objetivo y así generar una trayectoria mínima posible entre la ubicación del móvil y un obstáculo de tal manera que lo pueda evadir.

## BIBLIOGRAFÍA

- Biagiotti, L., & Melchiorri, C. (2008). *Trajectory Planning for Automatic Machines and Robots*. Springer Science & Business Media.
- Enciso Salas, L. M. (2015). Diseño de un Sistema de Navegación Autónomo para Robots Móviles Usando Fusión de Sensores y Controladores Neuro Difusos.
- Moran, A., Rázuri, J. G., Sundgren, D., & Rahmani, R. (2013, November). Autonomous Motion of Mobile Robot Using Fuzzy-Neural Networks. In *Artificial Intelligence (MICAI), 2013 12th Mexican International Conference on* (pp. 80-84). IEEE.
- Mohanty, P. K., Parhi, D. R., Jha, A. K., & Pandey, A. (2013, February). Path Planning of an Autonomous Mobile Robot Using Adaptive Network Based Fuzzy Controller. In *Advance Computing Conference (IACC), 2013 IEEE 3rd International* (pp. 651-656). IEEE.
- Fan, K., Yang, Q., Li, P., & Yan, W. (2012, September). On the Design of AGV Obstacle Avoidance System Based on Fuzzy-PID Dual-Mode Controller. In *Control, Systems & Industrial Informatics (ICCSII), 2012 IEEE Conference on* (pp. 54-58). IEEE.
- Tsalatsanis, A., Valavanis, K. P., Kandel, A., & Yalcin, A. (2007, June). Multiple Sensor Based UGV Localization Using Fuzzy Extended Kalman Filtering. In *Control & Automation, 2007. MED'07. Mediterranean Conference on* (pp. 1-8). IEEE.
- Tang, C. P. (2009, December). Differential Flatness-Based Kinematic and Dynamic Control of a Differentially Driven Wheeled Mobile Robot. In *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on* (pp. 2267-2272). IEEE.
- Rigatos, G. (2015). *Nonlinear Control and Filtering Using Differential Flatness Approaches: Applications to Electromechanical Systems* (Vol. 25). Springer.
- Tang, C. P., Miller, P. T., Krovi, V. N., Ryu, J. C., & Agrawal, S. K. (2011). Differential-Flatness-Based Planning and Control of a Wheeled Mobile Manipulator - Theory and Experiment. *IEEE/ASME Transactions on Mechatronics*, 16(4), 768-773.
- Khan, Z. H., Ali, A. S., & Riaz, Z. (Eds.). (2014). *Computational Intelligence for Decision Support in Cyber-Physical Systems* (Vol. 540). Springer.
- Song, Z., Yi, J., Zhao, D., & Li, X. (2004, August). Control of Nonholonomic Chained Systems Application to Time-Varying Exponential Stabilization of Mobile Robots. In *Intelligent Mechatronics and Automation, 2004. Proceedings. 2004 International Conference on* (pp. 964-969). IEEE.
- Mekki, H., & Letaief, M. (2013, December). Path Planning for 3D Visual Servoing: For a Wheeled Mobile Robot. In *Individual and Collective Behaviors in Robotics (ICBR), 2013 International Conference on* (pp. 86-91). IEEE.



- Alger, B., Votion, J., & Qian, C. (2014, June). Sampled-Data Output Feedback Control of Uncertain Nonholonomic Systems in Chained Forms with Applications to Mobile Robots. In *American Control Conference (ACC), 2014* (pp. 4659-4664). IEEE.
- Li, Z., Xiao, H., Yang, C., & Zhao, Y. (2015). Model Predictive Control of Nonholonomic Chained Systems Using General Projection Neural Networks Optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(10), 1313-1321.
- Song, X., Gao, H., Gao, H., & Ding, L. (2016, July). Neural Adaptive Tracking Control Via Chained Transform for Wheeled Mobile Robot. In *Control Conference (CCC), 2016 35th Chinese* (pp. 3029-3034). IEEE.
- De Luca, A., Oriolo, G., & Samson, C. (1998). Feedback Control of a Nonholonomic Car-Like Robot. *Robot Motion Planning and Control*, 171-253.
- Dierks, T., & Jagannathan, S. (2007, October). Control of Nonholonomic Mobile Robot Formations: Backstepping Kinematics into Dynamics. In *Control Applications, 2007. CCA 2007. IEEE International Conference on* (pp. 94-99). IEEE.
- Lu, J., Sekhavat, S., Xie, M., & Laugier, C. (2000, December). Sliding Mode Control for Nonholonomic Mobile Robot. In *Proceedings of the International Conference on Control, Automation, Robotics and Vision, ICARCV*.
- Moran, A., & Hayase, H. (1995, July). Shortest-Trajectory Control of Autonomous Mobile Robots Using Nonlinear Observers. In *SICE'95. Proceedings of the 34th SICE Annual Conference. International Session Papers* (pp. 1415-1418). IEEE.
- Moret, E. N. (2003). Dynamic Modeling and Control of a Car-Like Robot.
- Van Nieuwstadt, M. J. (1997). *Trajectory Generation for Nonlinear Control Systems* (Doctoral dissertation, California Institute of Technology).
- De Luca, A., Oriolo, G., & Vendittelli, M. (2001). Control of Wheeled Mobile Robots: An Experimental Overview. In *Ramsete* (pp. 181-226). Springer Berlin Heidelberg.
- Corke, P. (2011). *Robotics, Vision and Control: Fundamental Algorithms in MATLAB* (Vol. 73). Springer.
- Tzafestas, S. G. (2013). *Introduction to Mobile Robot Control*. Elsevier.
- Kleeman, L. (1996, January). Understanding and Applying Kalman Filtering. In *Proceedings of the Second Workshop on Perceptive Systems, Curtin University of Technology, Perth Western Australia (25-26 January 1996)*.
- Bishop, G., & Welch, G. (2001). An Introduction to the Kalman Filter. *Proc of SIGGRAPH, Course*, 8(27599-23175), 41.
- Grover, R., & Hwang, P. Y. (1997). Introduction to Random Signals and Applied Kalman Filtering. *Wiley, New York*.

- Vigouroux Cavolina, D. (2010). Implementación de Unidad de Mediciones Inerciales (IMU) para Robótica Utilizando Filtro de Kalman. *Sartenejas, Venezuela*.
- Caruso, M. J. (2000). Applications of Magnetic Sensors for Low Cost Compass Systems. In *Position Location and Navigation Symposium, IEEE 2000* (pp. 177-184). IEEE.
- Caruso, M. J. (1997). *Applications of Magnetoresistive Sensors in Navigation Systems* (No. 970602). SAE Technical Paper.
- Gulmammadov, F. (2009, June). Analysis, Modeling and Compensation of Bias Drift in MEMS Inertial Sensors. In *Recent Advances in Space Technologies, 2009. RAST'09. 4th International Conference on* (pp. 591-596). IEEE.
- Jekeli, C. (2001). *Inertial Navigation Systems with Geodetic Applications*. Walter de Gruyter.
- Bekir, E. (2007). *Introduction to Modern Navigation Systems*.
- Woodman, O., & Harle, R. (2008, September). Pedestrian Localization for Indoor Environments. In *Proceedings of the 10th International Conference on Ubiquitous Computing* (pp. 114-123). ACM.
- Woodman, O. J. (2007). *An Introduction to Inertial Navigation* (No. UCAM-CL-TR-696). University of Cambridge, Computer Laboratory.
- Levine, J. (2009). *Analysis and Control of Nonlinear Systems: A Flatness-Based Approach*. Springer Science & Business Media.
- Rigatos, G. (2015). *Nonlinear Control and Filtering Using Differential Flatness Approaches: Applications to Electromechanical Systems* (Vol. 25). Springer.
- Guo, Y., & Tang, T. (2008, May). Optimal Trajectory Generation for Nonholonomic Robots in Dynamic Environments. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on* (pp. 2552-2557). IEEE.
- Gasparetto, A., Boscariol, P., Lanzutti, A., & Vidoni, R. (2012). Trajectory Planning in Robotics. *Mathematics in Computer Science*, 6(3), 269-279.
- Zhao, R. (2015). *Trajectory Planning and Control for Robot Manipulations*, (Doctoral dissertation, Université de Toulouse, Université Toulouse III-Paul Sabatier).
- Murray, R. M., & Sastry, S. S. (1993). Nonholonomic Motion Planning: Steering Using Sinusoids. *IEEE Transactions on Automatic Control*, 38(5), 700-716.
- Guo, Y., Long, Y., & Sheng, W. (2007, April). Global Trajectory Generation for Nonholonomic Robots in Dynamic Environments. In *Robotics and Automation, 2007 IEEE International Conference on* (pp. 1324-1329). IEEE.
- Papadopoulos, E., Poulakakis, I., & Papadimitriou, I. (2002). On Path Planning and Obstacle Avoidance for Nonholonomic Platforms with Manipulators: A Polynomial Approach. *The International Journal of Robotics Research*, 21(4), 367-383.

- Chen, Q. (2007). *Studies in Autonomous Ground Vehicle Control Systems: Structure and Algorithms* (Doctoral dissertation, The Ohio State University).
- Oriolo, G., De Luca, A., & Vendittelli, M. (2002). WMR Control Via Dynamic Feedback Linearization: Design, Implementation, and Experimental Validation. *IEEE Transactions on Control Systems Technology*, 10(6), 835-852.
- Beleno, R. H., Vitor, G. B., Ferreira, J. V., & Meirelles, P. S. (2012, September). Planning and Tracking Trajectories of Land Vehicles Based on the Direction Control in a Real Environment. In *First International Conference on Advanced Mechatronics, Design, and Manufacturing Technology-AMDM 2012* (Vol. 1, pp. 1-6).
- Martin, P., Murray, R. M., & Rouchon, P. (2009). Flatness Based Design. *CONTROL SYSTEMS, ROBOTICS AND AUTOMATION—Volume XIII: Nonlinear, Distributed, and Time Delay Systems-II*, 65.
- Jarzebowska, E. (2012). *Model-Based Tracking Control of Nonlinear Systems*. CRC Press.
- Dong, W., & Guo, Y. (2005, May). New Trajectory Generation Methods for Nonholonomic Mobile Robots. In *Collaborative Technologies and Systems, 2005. Proceedings of the 2005 International Symposium on* (pp. 353-358). IEEE.
- Bessas, A., & Benalia, A. Differential Flatness-Based Planning and Trajectory Tracking of Wheeled Mobile Robots.
- Tick, D., Satici, A. C., Shen, J., & Gans, N. (2013). Tracking Control of Mobile Robots Localized Via Chained Fusion of Discrete and Continuous Epipolar Geometry, IMU and Odometry. *IEEE transactions on cybernetics*, 43(4), 1237-1250.
- Lugo-Cárdenas, I., Flores, G., Salazar, S., & Lozano, R. (2014, May). Dubins Path Generation for a Fixed Wing UAV. In *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on* (pp. 339-346). IEEE.
- Gloderer, M., & Hertle, A. (2010). Spline-Based Trajectory Optimization for Autonomous Vehicles with Ackerman drive.
- Sprunk, C. (2008). Planning Motion Trajectories for Mobile Robots Using Splines.
- Furtuna, A. A., & Balkcom, D. J. (2010). Generalizing Dubins curves: Minimum-Time Sequences of Body-Fixed Rotations and Translations in the Plane. *The International Journal of Robotics Research*, 29(6), 703-726.
- Gao, X. Z., Hou, Z. X., Zhu, X. F., Zhang, J. T., & Chen, X. Q. (2013). The Shortest Path Planning for Manoeuvres of UAV. *Acta Polytechnica Hungarica*, 10(1), 221-39.
- Fraichard, T., & Scheuer, A. (2004). From Reeds and Shepp's to Continuous-Curvature Paths. *IEEE Transactions on Robotics*, 20(6), 1025-1035.

- Sussmann, H. J., & Tang, G. (1991). Shortest Paths for the Reeds-Shepp Car: A Worked Out Example of the Use of Geometric Techniques in Nonlinear Optimal Control. *Rutgers Center for Systems and Control Technical Report, 10*, 1-71.
- Rigatos, G. (2011). *Modelling and Control for Intelligent Industrial Systems: Adaptive Algorithms in Robotics and Industrial Engineering* (Vol. 7). Springer Science & Business Media.
- Wang, Y., Chao, T., Wang, S., & Yang, M. (2016, July). Trajectory Tracking Control Based on Differential Flatness. In *Control Conference (CCC), 2016 35th Chinese* (pp. 10828-10832). IEEE.
- Bryson, A. E. (1975). *Applied Optimal Control: Optimization, Estimation and Control*. CRC Press.
- Grisales Ramírez, E. (2013). *Control de un Robot Móvil en Entornos Domésticos* (Doctoral dissertation, Universidad Nacional de Colombia-Sede Manizales).
- Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. (2011). *Introduction to autonomous mobile robots*. MIT press.
- Lewis, L. P. R. (2006). *Rapid Motion Planning and Autonomous Obstacle Avoidance for Unmanned Vehicles* (Doctoral dissertation, Monterey, California. Naval Postgraduate School).
- Hurni, M. A., Sekhavat, P., & Ross, I. M. (2009, April). Issues on UGV Optimal Motion Planning and Obstacle Avoidance. In *Proceedings of the AIAA Unmanned... Unlimited Conference and Exhibit*.