

**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS  
DEL INSTITUTO POLITÉCNICO NACIONAL**

UNIDAD ZACATENCO

Laboratorio UMI-LAFMIA 3175 CNRS

**Algoritmo de navegación basado en GPS-RTK para  
aplicación en agricultura de precisión**

**T E S I S**

Que presenta

**Ing. Edgar López Díaz**

Para obtener el grado de

**Maestro en Ciencias**

en

**Sistemas Autónomos de Navegación Aérea y Submarina**

Directores de Tesis:

Dr. Sergio Rosario Salazar Cruz

Dr. Eduardo Steed Espinoza Quesada

Ciudad de México

Diciembre, 2016.



# Agradecimientos

---

Porque todos podemos caminar solos, pero pocos tenemos la opción de hacerlo acompañados. Por eso me permito agradecer a:

**Mis padres:** quienes sin importar los problemas siempre han hecho lo posible por apoyarme.

**A mi hermana y su nueva familia:** quienes cuando así lo necesite me brindaron su apoyo y un segundo hogar.

**Familiares:** que sin pedir nada me ayudaron a comenzar esta experiencia.

**Compañeros y ahora amigos:** quienes con el transcurso de los días fueron quienes volvieron más amena esta experiencia.

**CONACYT:** que de no haber sido por los recursos brindados tal vez otra cosa sería.

**Mis asesores:** quienes me dieron la oportunidad de incursionar en este, el ámbito de la investigación así como fomentar la idea de continuar.

A aquellas personas que sin saber llegaron en el momento justo.



# Resumen

---

Siguiendo con la vertiente del rubro tecnológico es que la sociedad en si ha tendido a implementar el uso de las tecnologías en su día con día. Ya sea por mejorar su desarrollo, calidad de vida, o mero entretenimiento. Cambiando la forma en que se realizaban las tareas. Tomando como ejemplo el sector agrario, el cual a tomado una postura a favor de incorporar nuevas herramientas abriendo paso a la llamada Agricultura de Precisión.

Tomando en cuenta este rubro, y tras lo ya mencionado es que este trabajo consta en incursionar en el desarrollo de herramientas en favor de la Agricultura de Precisión con el desarrollo de un Algoritmo de Navegación basado en GPS de tipo RTK. El cual se planea servirá como apoyo a la ejecución autónoma de ciertas tareas.

El trabajo fue desarrollado en siete bloques: 1) Inmersión en el tema, donde tras cierta investigación se informo de los avances que a este medio le rodean. 2) Creación. Que mediante la selección de una plataforma y su correcta instrumentación dio origen al que seria el sistema con que se realizarían los objetivos planteados. 3) Análisis del funcionamiento por parte del que sera el sensor principal del sistema. 4) Comprensión del funcionamiento por parte de la plataforma tomando como referencia el desarrollo de una expresión matemática. 5) Elaboración del que será el encargado de la toma de desiciones por parte del sistema, el Sistema de navegación. 6) Pruebas que aunque a manera teórica buscan comprobar el comportamiento y desempeño del sistema. 7) Aplicación sometiendo el Sistema obtenido a pruebas que demuestren su correcto funcionamiento bajo las tareas requeridas.

Presentado así a grandes rasgos un resumen del que será el contenido de este trabajo.



# Abstract

---

Following with the slope of the technological field is that society itself has tended to implement the use of technologies in their day by day. Whether to improve their development, quality of life, or mere entertainment. Changing the way the tasks were done. Taking as an example the agricultural sector, which has taken a stance in favor of incorporating new tools opening the way to the so-called Precision Agriculture. Taking into account this item, and after what is already mentioned, this work consists of entering into the development of tools in favor of Precision Agriculture with the development of a GPS-based Navigation Algorithm of the RTK type. Which is planned will serve as support for the autonomous execution of certain tasks. The work was developed in seven blocks: 1) Immersion in the subject, where after some investigation he was informed of the advances that this environment surround him. 2) Creation. That through the selection of a platform and its correct instrumentation gave rise to what would be the system with which the objectives would be achieved. 3) Analysis of the operation by which will be the main sensor of the system. 4) Understanding of the operation by the platform taking as a reference the development of a mathematical expression. 5) Elaboration of which will be in charge of making decisions by the system, the navigation system. 6) Tests that although in a theoretical way seek to verify the behavior and performance of the system. 7) Application by submitting the System obtained to tests that demonstrate its correct operation under the required tasks.

Presented thus broadly a summary of what will be the content of this work.



# Índice general

---

<b>Agradecimientos</b>	<b>III</b>
<b>Resumen</b>	<b>V</b>
<b>Índice de figuras</b>	<b>XI</b>
<b>1. Introducción</b>	<b>3</b>
1.1. Agricultura de precisión . . . . .	5
1.2. Robots móviles con ruedas . . . . .	6
1.2.1. Configuración . . . . .	7
1.2.1.1. Diferencial . . . . .	7
1.2.1.2. Triciclo . . . . .	7
1.2.1.3. Ackerman . . . . .	8
1.2.1.4. Omnidireccional . . . . .	8
1.2.1.5. Skid Steer . . . . .	9
1.2.1.6. Oruga . . . . .	9
1.3. Vehículos Autónomos . . . . .	10
1.4. Sistemas de Navegación como parte de la Agricultura de Precisión . . . . .	11
1.5. Organización del trabajo . . . . .	17
<b>2. Instrumentación de plataforma</b>	<b>19</b>
2.1. Estación base (GCS) . . . . .	19
2.1.1. Elementos que componen GCS . . . . .	20
2.2. Vehículo (Rover) . . . . .	22
2.2.1. Elementos que componen al vehículo (Rover) . . . . .	22
2.2.2. Descripción del vehículo . . . . .	24

2.2.2.1.	Adquisición de datos . . . . .	25
2.2.2.2.	Procesamiento y sistema de navegación . . . . .	26
2.2.2.3.	Acoplamiento . . . . .	27
2.3.	Descripción de componentes . . . . .	29
2.3.1.	Odroid-C2 . . . . .	29
2.3.2.	PIXHAWK . . . . .	29
2.3.2.1.	Magnetometro (brújula digital) . . . . .	30
2.3.3.	Piksi-RTK . . . . .	31
2.3.4.	Sabertooth 2x25 V2 . . . . .	31
2.3.5.	RPLidar 360 . . . . .	32
2.3.6.	Telemetria Sik . . . . .	33
2.3.7.	Receptor SPECKTRUM AR8000 . . . . .	33
2.3.8.	Traco Power TSR 1-2450 . . . . .	34
2.3.9.	Bateria HeliBoss . . . . .	35
2.3.10.	POWERBANK . . . . .	35
2.3.11.	Lenguaje de programación . . . . .	36
2.3.12.	Protocolos de comunicación . . . . .	37
2.3.12.1.	Secure Shell (SSH) . . . . .	37
2.3.12.2.	Comunicación TCP/IP por SOCKET . . . . .	37
2.3.13.	Piksi Console . . . . .	39
2.3.14.	MissionPlanner . . . . .	39
2.3.15.	MAVProxy . . . . .	40
<b>3.</b>	<b>GPS-RTK</b>	<b>43</b>
3.1.	Concepto general de GPS . . . . .	43
3.1.1.	Bloques de una estructura GPS . . . . .	43
3.1.2.	Estructura de la señal de satélite . . . . .	44
3.1.3.	Rango de satélite basado en la medición de código . . . . .	45
3.1.4.	Rango de satélite basado en medición de fase portadora . . . . .	45
3.1.5.	Sistemas de coordenadas GPS . . . . .	46
3.1.6.	Fuentes de error en GPS . . . . .	46
3.2.	GNSS diferencial y RTK . . . . .	47
3.2.1.	IAR (Integer Ambiguity Resolution) . . . . .	48

3.2.1.1.	Medición de fase . . . . .	48
3.2.1.2.	Diferencia de fase portadora . . . . .	48
3.2.1.3.	Solución de Ambigüedad . . . . .	48
3.2.1.4.	Solución Float y FIX . . . . .	49
3.2.1.5.	Calcular solución de baseline . . . . .	50
3.3.	RTK como parte del Sistema de Navegación . . . . .	50
<b>4.</b>	<b>Modelado</b>	<b>53</b>
4.1.	Modelado del vehículo . . . . .	53
4.1.1.	Modelo cinemático . . . . .	53
<b>5.</b>	<b>Sistema de navegación</b>	<b>61</b>
5.1.	Cálculo de trayectoria de referencia . . . . .	63
5.1.1.	Selección de la trayectoria . . . . .	63
5.1.2.	Generación de la trayectoria . . . . .	63
5.2.	Sistema de control . . . . .	67
5.2.1.	Definición de error . . . . .	67
5.2.2.	Control Proporcional Integral Derivativo (PID) . . . . .	69
5.3.	Localización . . . . .	70
<b>6.</b>	<b>Simulación</b>	<b>73</b>
6.0.1.	Resultados experimentales . . . . .	73
6.1.	Simulación 3D bajo entorno Blender . . . . .	82
6.2.	Conclusión . . . . .	82
<b>7.</b>	<b>Resultados experimentales</b>	<b>84</b>
7.1.	Problema de regulación . . . . .	84
7.2.	Patron de cultivo . . . . .	88
7.3.	Pruebas extra . . . . .	91
<b>8.</b>	<b>Conclusiones</b>	<b>96</b>
	<b>Bibliografía</b>	<b>98</b>

# Índice de figuras

---

1.1. Esta ilustración muestra ejemplos de las diferentes tecnologías de la agricultura de precisión que están siendo usadas para la producción de cultivo en sitios específicos [18]. . . . .	6
1.2. Vehículo con configuración diferencial visto en plano XY. . . . .	7
1.3. Vehículo con configuración de Triciclo visto en plano XY . . . . .	8
1.4. Vehículo con configuración Ackerman visto en plano XY . . . . .	8
1.5. Vehículo con configuración Omnidireccional visto en plano XY . . . . .	9
1.6. Vehículo con configuración Skid Steer visto en plano XY, cave mencionar el conjunto de ruedas suele conformarse de un numero par partiendo de cuatro en adelante. . . . .	9
1.7. Vehículo con sistema de Orugas visto en plano XY . . . . .	10
1.8. Se pueden ver algunas muestras de lo que hoy en día la investigación esta desarrollando con el objetivo de la agricultura de precisión. (a)CT801 (YANMAR). Diseñado en Japón para su uso en los campos de arroz [34][35]. (b) El tractor guiado automáticamente y el sistema de registro automático para superficies rugosas sobre la superficie de grava [9]. (c) Orpheus-X3 mobile robot. Forma parte de un sistema heterogéneo de reconocimiento [15]. (d) Un Drone volando para la exploración aérea de varias parcelas de trigo. [18] . . . . .	12
1.9. Robocrop, tractor robotizado de la empresa Tillett and Hague Technology Ltd., para eliminar la maleza de forma mecánica sin dañar el cultivo útil [18]. . . . .	13

1.10. Sistema de monitoreo y de siembra variable montado en una sembradora de la empresa BHV, dinámica presentada en Chile, (derecha) y sistema incorporado en plataforma digital (izquierda). [18] . . . . .	14
1.11. Instrumentación convencional para vehículos dedicados a la agricultura de precisión. (a) Sensores necesarios para el control del rendimiento [9]. (b) Esquema del circuito del sistema de la sembradora IOM Agrometal - Verion inteligente [17]. . . . .	15
1.12. Sistema autopiloto montado en tractor y siembra realizada con autopiloto [18]. . . . .	15
1.13. Sistema de auto guiado para cosechadoras: SmartSteer [18]. . . . .	16
1.14. Esquema de funcionamiento de Connected Farm [18] . . . . .	16
2.1. Diagrama de interacción de componentes en GCS. . . . .	20
2.2. Ventana de Piksi Console al monitorear posición relativa de GPS Rover al este estar en desplazamiento y encontrarse bajo estado RTK-FIX. . . .	21
2.3. Ventana de Mission Planner al realizar monitoreo de posición global del vehículo al este estar en desplazamiento. Prueba realizada en las instalaciones de UNR. . . . .	21
2.4. Modulo de control agregado para realizar ajuste de ganancias y asignar referencias de prueba, así como mantener un monitoreo en tiempo real de la dinámica del vehículo. Solo accesible mediante el Servidor SSH. . . .	22
2.5. Esquema de elementos que conforman al vehículo (Rover). El cuadro que conforma el esquema muestra la forma en que se comunica cada dispositivo con los demás. . . . .	23
2.6. Diagrama de interacción de componentes en el vehículo. . . . .	24
2.7. Visualización de entorno detectado por vehículo mediante lidar. Modulo creado para trabajar en combinación con el GPS y compass. . . . .	25
2.8. Diagrama de procesamiento realizado a bordo del vehículo. . . . .	27
2.9. Foto de sistema RTK. . . . .	28
2.10. Odroid-C2 . . . . .	29
2.11. Pixhawk . . . . .	30
2.12. Dispositivo Piksi de Swift NAV . . . . .	31
2.13. Driver Sabertooth 2x25 V2 . . . . .	32

2.14. RPLidar 360 . . . . .	33
2.15. Telemetria Sik . . . . .	34
2.16. Receptor SPECKTRUM AR8000 . . . . .	34
2.17. Traco Power TSR 1-2450 . . . . .	35
2.18. Bateria HeliBoss (4S, 6200mA, 25C) . . . . .	35
2.19. POWERBANK (4000mAh) . . . . .	36
2.20. Proceso de comunicación a través de Sockets . . . . .	38
2.21. Consola de Piksi (se muestra la ventana de Baseline una vez se a obtenido RTK aclarando que la presente se encuentra en modo simulación) . . . . .	39
2.22. Mission Planner (ventana de inicio) . . . . .	40
3.1. Estructura de la Señal de Satélite. Relación entre Código y señal portadora .	44
3.2. Medición de código, desfase presente entre señal del satélite y la generada por el receptor. . . . .	45
3.3. Diferencia de fase portadora entre receptor Rover y receptor GCS . . . . .	49
3.4. Como se muestra en la imagen, al ser relativa la distancia entre estos dos puntos, la posición de uno con respecto del otro dependerá del receptor del que se este tomando la medición. Puesto que aquí se muestra como el Rover se encuentra al Este de la Base y de igual forma la Base se encuentra al Oeste del Rover. . . . .	51
3.5. En ambas imágenes se muestra el monitoreo de los receptores Piksi, to- mando como diferencia que la imagen (a) muestra un estado Float mien- tras que la (b) un estado Fix. Así mismo se muestra como en la imagen (a) previo a pasar a un estado Fix a diferencia de (b) en la sección IAR refleja un numero elevado indicando los flotantes por resolver. . . . .	52
4.1. Esquema de movimientos según efecto de las velocidades en las ruedas. .	54
4.2. Diagramas de fuerzas que intervienen en el vehículo. . . . .	55
5.1. Diagrama que muestra el algoritmo designado a seguir para la navegación del vehículo. . . . .	62
5.2. Trayectoria de siembra por surcos. La orientación del surco dependerá de la caída del terreno y el arco de insolación del sol. . . . .	63
5.3. Parámetros a considerar para el cálculo del polinomio a aplicar a la tarea de siembra. . . . .	64

5.4. Estructura de algoritmo para crear trayectoria representada mediante diagrama de flujo. . . . .	66
5.5. Trayectoria de prueba generada mediante polinomio donde $dL=3m$ , $dl=0.5m$ y $rg=0.25m$ . . . . .	66
5.6. Diagrama de referencias. Se muestra al vehículo en relación con el punto de partida y el punto final ( $Wp_A$ , $Wp_B$ ) respectivamente, denotando así las diferencias de posición y orientación. . . . .	68
5.7. Diagrama a bloques de un robot en el cual interviene un controlador de tipo Proporcional Integral Derivativo. . . . .	70
5.8. Localización global del vehículo con respecto a su posición relativa entre Rover y Base, así como la absoluta de la Base. . . . .	71
6.1. Trayectoria de prueba 1: Recta, se parte de condiciones iniciales iguales con cero. Demandando un desplazamiento constante por parte de X mientras la variable Y se mantiene fija con cero. . . . .	74
6.2. Respuesta de los estados $[X Y \varphi]$ vista de manera separada al realizar la trayectoria de prueba 1. Donde se muestra como la línea que representa al vehículo (Plataforma) trata de seguir la señal deseada con un pequeño desfase por parte de la respuesta en X. Mientras que para Y y $\varphi$ aunque no se muestra del todo uniforme, mantiene una oscilación acotada al rededor de cero. . . . .	75
6.3. Trayectoria de prueba 2: Rampa, se parte de condiciones iniciales iguales con cero. Demandando un desplazamiento constante por parte de $[X,Y]$ . . . . .	76
6.4. Respuesta de los estados $[X Y \varphi]$ vista de manera separada al realizar la trayectoria de prueba 2. Donde se muestra como la línea que representa al vehículo (Plataforma) trata de seguir la señal deseada con un pequeño desfase por parte de la respuesta en $[X, Y]$ . Mientras que para $\varphi$ aunque no se muestra del todo uniforme, mantiene una oscilación acotada al rededor de 45. . . . .	77
6.5. Trayectoria de prueba 3: Seno, se parte de condiciones iniciales iguales con cero. Demandando un desplazamiento constante por parte de X, mientras que para Y se pide seguir una función seno. . . . .	78

6.6.	Respuesta de los estados $[X \ Y \ \varphi]$ vista de manera separada al realizar la trayectoria de prueba 3 Donde se muestra como la linea que representa al vehículo (Plataforma) trata de seguir la señal deseada con un pequeño desfase por parte de la respuesta en $[X, Y]$ . Mientras que $\varphi$ aunque no se muestra del todo uniforme, mantiene una oscilación acotada al rededor de la orientación deseada provocada por la trayectoria. . . . .	79
6.7.	Trayectoria de prueba 4: Hélice u Ocho, se parte de condiciones iniciales iguales con cero. Demandando un desplazamiento senoidal para X, mientras que para Y se pide seguir una función triangular. . . . .	80
6.8.	Respuesta de los estados $[X \ Y \ \varphi]$ vista de manera separada al realizar la trayectoria de prueba 4 Donde se muestra como la linea que representa al vehículo (Plataforma) trata de seguir la señal deseada con un pequeño desfase por parte de la respuesta en $[X, Y]$ . Mientras que $\varphi$ aunque no se muestra del todo uniforme, mantiene una oscilación acotada al rededor de la orientación deseada provocada por la trayectoria. . . . .	81
6.9.	Vista del entorno de simulación 3D desarrollado bajo Blender. . . . .	83
7.1.	Foto de sistema tomada en área de prueba. . . . .	85
7.2.	Desplazamiento realizado por el vehículo al dirigirse a un punto deseado denotado por las coordenadas (4,4) teniendo como punto de partida la ubicación denotada por las coordenadas(0.6,0). . . . .	86
7.3.	Respuesta de las variables de estado (X,Y, Yaw) así como de las señales de control $(U_1, U_2)$ . . . . .	87
7.4.	Evolución del error durante el desplazamiento del vehículo, mostrando como este tiende a cero, quedando con un valor de estado estacionario de .022m. . . . .	88
7.5.	Desplazamiento del vehículo siguiendo el algoritmo asignado para realizar la tarea de cultivo de manera autónoma. . . . .	89
7.6.	Error de posición a lo largo de la trayectoria, donde el valor máximo, mínimo y promedio fueron (0.13664m, $31794e^{-4}$ m, 0.0769m) respectivamente. . . . .	89
7.7.	Respuesta de las variables de estado (X,Y, Yaw) así como de las señales de control $(U_1, U_2)$ . . . . .	90

7.8. Trayectoria uno (recta). Desplazamiento bajo evolución de una variable con condiciones iniciales igual con cero. . . . .	91
7.9. Error de seguimiento para la trayectoria uno, donde el valor máximo, mínimo y promedio fueron (0.1071m, 0.0202m, 0.0533m) respectivamente. . . . .	92
7.10. Trayectoria dos (Pendiente). Desplazamiento bajo evolución de dos variables con condiciones iniciales igual con cero. . . . .	92
7.11. Error de seguimiento para la trayectoria dos, donde el valor máximo, mínimo y promedio fueron (0.0834m, -0.0117m, 0.0537m) respectivamente. . . . .	93
7.12. Trayectoria tres (seno). Desplazamiento bajo evolución de dos variables con condiciones iniciales igual con cero. . . . .	93
7.13. Error de seguimiento para la trayectoria tres, donde el valor máximo, mínimo y promedio fueron (0.0957m, -0.1083m, 0.0181m) respectivamente. . . . .	94
7.14. Trayectoria cuatro (hélice). Desplazamiento bajo evolución de dos variables con condiciones iniciales igual con cero. . . . .	94
7.15. Error de seguimiento para la trayectoria cuatro, donde el valor máximo, mínimo y promedio fueron (0.1414m, $1.4527e^{-4}$ m, 0.0371m) respectivamente. . . . .	95



# Lista de Acrónimos

---

**ADC** Analogic Digital Converter.

**AP** Agricultura de Precision.

**APC** Antena Phase Center.

**C/A** Coarse Acquisition.

**DARPA** Defense Advanced Research Projects Agency.

**DGNSS** Diferencial GNSS.

**DoD** Department of Defense .

**ECEF** Earth Centered Earth Fixed.

**GCS** Ground Control Station.

**GNSS** Global Navigation Satellite System.

**GPIO** General-Purpose Input/Output.

**GPS** Global Positioning System.

**GUI** Graphic User Interface.

**IAR** Integer Ambiguity Resolution.

**IMU** Inertial Measurement Unit.

**IP** Internet Protocol.

**NAVSTAR** Navigation System using Timing And Ranging.

**NED** North East Down.

**OS** Operating System.

**OTG** One-to-Go.

**PID** Proportional Integer Derivative.

**PRN** Pseudo Random.

**PWM** Pulse-Width Modulation.

**RTK** Real-Time Kinematics.

**SBP** Swift Navigation Binary Protocol.

**SSH** Secure Shell .

**TCP** Transmisión Control Protocolo.

**TICs** Tecnologías de la Información y Comunicaciones.

**UAV** Unmanned Aerial Vehicle.

# Introducción

---

## Problema a resolver

Hoy en día la tecnología ha aumentado la productividad agrícola. Razón por la cual la agricultura moderna esta obligando cada vez más a utilizar practicas que lleven a los agricultores a mejorar sus costos productivos, evitar perdidas y garantizar una producción estable, por lo cual se ha comenzado a utilizar métodos modernos para mejorar su producción, tales como la agricultura de precisión. Siendo esta tecnología diseñada para facilitar el trabajo a quienes laboran en el campo de diversas maneras siendo algunas el aporte de datos que revelan donde se encuentran las fallas que originan bajos rendimientos de los cultivos, la automatización de tareas que requieren precisión, el ahorro de insumos, la posibilidad de trabajar a distancia sin importar el horario, clima, entre otro conjunto de factores que durante siglos han complicado el trabajo agropecuario. Sin embargo, la baja disponibilidad, falta de información y los elevados costos resultan en un constante problema para los sectores minoristas, limitando el uso a grandes productores o prestadores de servicios con recursos financieros para solventar una inversión de este tipo. Motivado por las observaciones previas, en el presente proyecto se pretenden utilizar sistemas de bajo costo para la construcción e implementación de un autopiloto para vehículos terrestres no tripulados en aplicaciones de agricultura de precisión a fin de mejorar eficiencia y sobre todo, disminuir los costos de producción.

## Objetivos

Desarrollar un vehículo terrestre el cual será capaz de realizar trayectorias asignadas con precisión de centímetros. Dicho vehículo se utilizará como base para uno más grande el cual podrá ocuparse en terrenos agrícolas: para esparcir fertilizante, o fungicidas en áreas específicas detectadas, o ya sea el control de un tractor con fin de cultivo.

Teniendo como objetivos específicos:

- Desarrollar algoritmos de navegación y comunicación que permitan realizar misiones de manera autónoma, compartiendo información con una estación de monitoreo (GCS).
- Implementar los algoritmos desarrollados mediante software y hardware basado en código libre ejecutado en autopilotos y sistemas embebidos.
- Integrar sensores, actuadores, sistemas embebidos y plataforma para desarrollar vehículo autónomo y estación fija para realizar aplicaciones de agricultura de precisión.
- Realizar pruebas en el exterior para validar en tiempo real los algoritmos desarrollados.

## Hipótesis

Se pretende desarrollar algoritmos que permitan a un vehículo terrestre el desplazamiento en áreas de cultivo mediante sensores (GPS, Lidar) los cuales le permitan realizar trayectorias con precisión de centímetros, con el fin de resolver una tarea para el cultivo (inspección, siembra, o esparcir fungicidas) todo esto bajo la filosofía Open Source.

## Justificación

En la actualidad ya es conocida la eficiencia de los vehículos autónomos en diversos ámbitos, siendo ahora presente el caso de la agricultura de precisión, la cual consiste en la aplicación de técnicas geoespaciales y sensores (sistemas de información geográfica, sensor remoto, GPSs, Lidar, cámaras, entre otros) permitiendo la detección de anomalías

y posteriormente lidiar con ellas mediante la implementación de estrategias alternativas. Teniendo, los sistemas globales de navegación por satélite como los mas populares. Los cuales han impulsado la navegación y medición de áreas y distancias, especialmente en el medio rural. Así mismo estos han sido aplicados a pilotos automáticos, utilizados para guiar vehículos sin la asistencia de un ser humano. Hoy en día esta clase de equipos es utilizado implementando el uso de sistemas de posicionamiento RTK (Real Time Kinematics) los cuales permiten dirigir ya sea un tractor con un alto grado de precisión, evitando así errores en las líneas de siembra y realizar tareas aun mas precisas. Tomando en cuenta este enfoque, se tiene como resultado una gran ventaja por parte de la agricultura de precisión en comparación con la tradicional debido a la posibilidad de utilizar los insumos de manera cada vez más eficiente con dosis adecuadas y de acuerdo a la real necesidad del cultivo. Es por este motivo que en el presente proyecto se pretenden abordar los retos tecnológicos necesarios para realizar agricultura de precisión con la intención de aprovechar al máximo los beneficios potenciales de los vehículos no tripulados.

## **1.1. Agricultura de precisión**

El uso de sistemas no tripulados hoy en día se encuentra enfocado principalmente a aplicaciones de fotografía aérea, vigilancia y cartografía. Sin embargo, se prevé que este tipo de vehículos pueda ser utilizado para una variedad más grande de aplicaciones, dentro de las cuales se encuentran la entrega de paquetes, actividades policíacas y militares en áreas urbanas y aplicaciones relacionadas con la agricultura. Esta última actividad es en la que se considera que tendrá más impacto la utilización de UAS, debido a que no existen riesgos potenciales para la población en general, así como para la infraestructura.

Agricultura de precisión ha sido el termino dado que describe al aumento en cuanto a la eficiencia en la administración de la agricultura el cual consiste en automatizar la gestión de Cultivos usando las TICs [9][20].

Partiendo de que la agricultura de precisión fue concebida desde EE.UU. como un círculo que se retroalimentaba año con año, donde el principal objetivo era concluido con la dosis variable de insumos en el momento adecuado y el lugar exacto. Siendo incorporado el uso de tecnología de información para así adecuar el uso de suelos y cultivos [8][34]. Contando dentro del sistema con herramientas clave, tales como lo es el uso de sistemas de posicionamiento global (GPS) [9] [21][15][35]y la electrónica, los cuales dan

vialidad a la recopilación de datos en tiempo real sobre lo que sucede y ha sucedido en un cultivo. La implementación de la tecnología en la agricultura de precisión han dado paso a cubrir las necesidades de la agricultura moderna: optimización en cuanto al correcto manejo de las tierras (Figura 1.1). Teniendo como principal ventaja que el análisis de resultados de las pruebas puede ser realizado en distintos sectores dentro del mismo lote, y de esta manera ajustar el manejo diferencial dentro del mismo. De este modo, podrán analizarse, el tipo y la dosis de fertilizante a aplicar, la densidad de semilla, la fecha de siembra, el esparcimiento entre hileras, etc. El uso de las tecnologías de la agricultura de precisión puede ayudar a mejorar los márgenes, a través de un aumento del valor del rendimiento (cantidad o calidad), de una reducción en la cantidad de insumos, o de ambos simultáneamente.

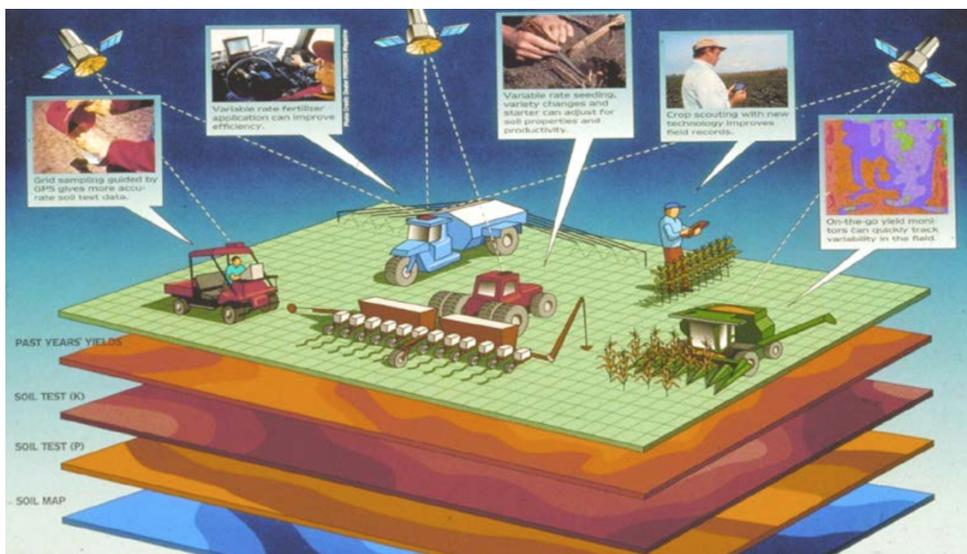


Figura 1.1: Esta ilustración muestra ejemplos de las diferentes tecnologías de la agricultura de precisión que están siendo usadas para la producción de cultivo en sitios específicos [18].

## 1.2. Robots móviles con ruedas

Siendo la plataforma de prueba un vehículo terrestre es necesario considerar los diferentes tipos disponibles.

Los robot móviles con ruedas son una solución rápida y eficiente para superficies duras y despejadas. Permiten regular y alcanzar altas velocidades. El problema de estos robot aparece en terrenos blandos o en el arranque donde se puede producir el deslizamiento de las ruedas. También aparecen vibraciones cuando transitan por terrenos irregulares.

## 1.2.1. Configuración

Dentro de este tipo de vehículos existen diferentes configuraciones dependiendo de su maniobrabilidad o configuración de ruedas.

### 1.2.1.1. Diferencial

Resulta en una configuración simple y fácil de implementar. Parte de una estructura formada por dos ruedas colocadas en el eje perpendicular a la dirección del robot como se muestra en la Figura 1.2. Al ser cada rueda controlada por un motor independiente el giro del robot es determinado por la diferencia de velocidad de las ruedas [7][2][24]. Así para girar a la izquierda, hay que darle una velocidad mayor a la rueda derecha, mientras que al girar a la derecha hay que dar mayor velocidad a la rueda izquierda. Así mismo es capaz de girar sin tener que desplazarse. Uno de los problemas que tiene la configuración diferencial es mantener el equilibrio del robot, ya que consta de dos ruedas, por lo que es usual que se agregan ruedas locas, permitiendo así mantener el equilibrio y mejorar la orientación para seguir caminos. Otro problema es el de apoyo en terrenos irregulares, ya que pueden producir la pérdida de tracción.

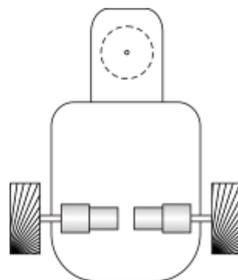


Figura 1.2: Vehículo con configuración diferencial visto en plano XY.

### 1.2.1.2. Triciclo

Formado por una estructura de 3 ruedas. Donde las dos ruedas traseras son consideradas pasivas al no estar acopladas a algún motor, sirviendo únicamente como soporte véase Figura 1.3. Mientras que la frontal es la que se encarga de la dirección y tracción del vehículo. Teniendo como desventaja el no poder girar sin desplazarse y suele tener el centro de masa alejado de la rueda motriz. Generando así pérdida de tracción, por ejemplo tratando de trepar una subida.

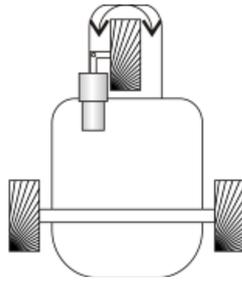


Figura 1.3: Vehículo con configuración de Triciclo visto en plano XY

### 1.2.1.3. Ackerman

Su configuración consta de dos ruedas traseras que se encargan de la tracción y un par frontal que se encarga de la dirección [21] véase Figura1.4. Esta configuración está creada para evitar el derrape de las ruedas, lo que se consigue haciendo que la rueda delantera interior posea un ángulo ligeramente mayor que el ángulo de la rueda exterior cuando el sistema se encuentra girando . La configuración Ackerman, constituye un buen sistema de tracción, aplicable inclusive para terrenos inclinados. Teniendo como inconveniente al igual que el triciclo el no poder girar si no es desplazándose.

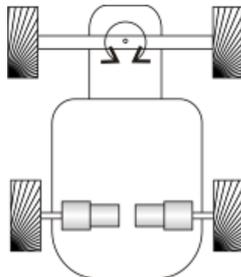


Figura 1.4: Vehículo con configuración Ackerman visto en plano XY

### 1.2.1.4. Omnidireccional

Este tipo de configuración esta provista con ruedas omnidireccionales, lo que hace que los cálculos de odometría sean mas complicados; pero al mismo tiempo dando al robot la posibilidad moverse en cualquier dirección [11] véase Figura1.5. La rueda omnidireccional, se define como una rueda estándar, a la cual se le ha dotado de una corona de rodillos. Los ejes de giro de los rodillos, son perpendiculares a la dirección normal de avance. De este modo, al aplicarle una fuerza lateral, los rodillos giran sobre si mismo y permitiendo que la componente de la velocidad en el eje x, no sea nulo .

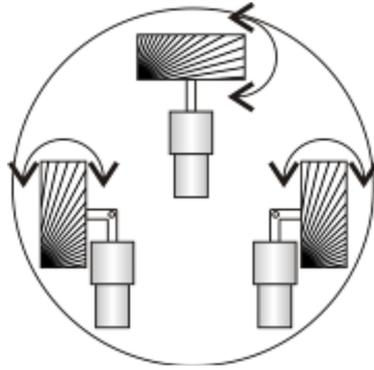


Figura 1.5: Vehículo con configuración Omnidireccional visto en plano XY

### 1.2.1.5. Skid Steer

Esta es una extensión de la configuración diferencial ya que dispone de varias ruedas en cada lado del robot las cuales actúan simultáneamente [28][33][15][29], véase Figura 1.6. La navegación queda definida por el control de la velocidad dada por el grupo de ruedas derechas e izquierdas. Tiene la ventaja de poder girar sobre su centro y poder cargar peso sobre sus ruedas lo que lo hace extremadamente robusto en terrenos irregulares. Estas dos ventajas lo convierten en el modelo elegido para los minicargadores y robots de investigación minera.

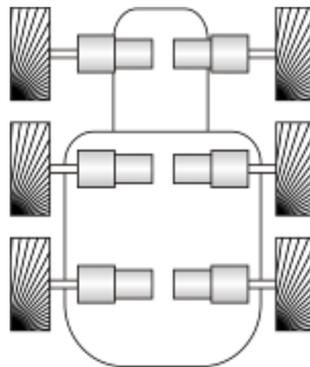


Figura 1.6: Vehículo con configuración Skid Steer visto en plano XY, cave mencionar el conjunto de ruedas suele conformarse de un numero par partiendo de cuatro en adelante.

### 1.2.1.6. Oruga

Funciona de forma análoga al Skid Steer pero se impulsa y dirección a través de cintas de deslizamiento, véase Figura 1.7. Es muy eficiente en terrenos irregulares ya que se asegura la tracción a pesar de las imperfecciones del terreno.

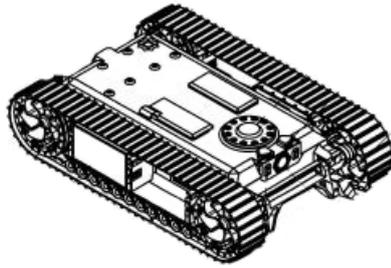


Figura 1.7: Vehículo con sistema de Orugas visto en plano XY

### 1.3. Vehículos Autónomos

Esta habilidad ha sido considerada como vital para los robots que nos interesan, ya que de poco nos sirve un robot que no sea capaz de desplazarse en su entorno.

Un vehículo autónomo es capaz de resolver las tareas humanas de manejo y control [11][14][29]. Siendo capaz de percibir el medio que le rodea y navegar en consecuencia [28][37]. Siendo el operador quien elija el destino, y dejando al vehículo se encargue de toda operación mecánica del vehículo.

Algunas de las técnicas empleadas para la percepción del entorno suelen implementar el uso de láser, radar [38], lidar, GPS [9][21][13][32 ]y visión artificial [22][12]. Siendo los sistemas de control los encargados de interpretar la información y en base a ello identificar la ruta, así como obstáculo e interpretación del entorno [33][25]. Los vehículos autónomos son capaces de recorrer trayectorias previamente programadas.

La presentación más antigua de un vehículo autónomo que se conoce fue por Norman Bel Geddes en la feria de muestras Futurama patrocinada por General Motors para la Exposición Universal de 1939, que consistía en un vehículo eléctrico que era controlado por un circuito eléctrico embebido en el pavimento de la carretera.

En 1980 Defense Advanced Research Projects Agency (DARPA) consiguió el primer vehículo que funcionaba mediante un radar láser y visión computarizada.

Para 1987, los laboratorios HRL demostraron que se podía construir un vehículo que podía diseñar su propia ruta una vez que se salía del mapa.

Los principales retos que conciernen a este tema resultan:

- Localización.
- Mapeo.
- Planificador de rutas

- Seguimiento de rutas

## **1.4. Sistemas de Navegación como parte de la Agricultura de Precisión**

Siendo en los años 70 que el DoD comenzó el lanzamiento de satélites geoestacionarios alrededor de la Tierra, teniendo como principal tarea la de facilitarles el localizar objetivos militares de forma precisa y rápida. Nombrando a este proyecto como Sistema de Posicionamiento Global, mejor conocido como GPS[18][37]. La principal meta del GPS cuando se creó era el de dirigir proyectiles teniendo como objetivos países enemigos. Sin embargo, hoy en día suele ser aplicado a diversos campos como lo son la geodesia, geofísica, geodinámica, astronomía, meteorología, cartografía o topografía, por nombrar algunos. Así mismo también se utiliza en la navegación marina, aérea o terrestre. Dando origen a nuevas materias de investigación que dependen totalmente del GPS, como lo es la Agricultura de Precisión como se muestra en la Figura 1.8.

Trayendo a flote el tema de Robotización de la agricultura. Debido a la alza en cuanto exigencia de producción, diversidad y calidad en los productos, así como la presentación de estos, es que el sector alimentario ha comenzado con la incorporación de tecnologías avanzadas (Figura 1.8). Siendo los objetivos planteados bajo la robotización de las tareas agrícolas[18] (Figura 1.9):

- Permitir la sustitución de operarios en tareas peligrosas para la salud.
- Abordar la realización de tareas repetitivas y tediosas, para el hombre.
- Realizar tareas en horas nocturnas, lo cual permite el ahorro de tiempo, por ejemplo, en la recolección.
- Mejorar la precisión en algunas de las tareas agrícolas, como las relacionadas con la biotecnología, y en concreto la multiplicación de plantas a partir de tejido vegetal.
- Optimizar la calidad de algunas de las tareas como la uniformidad en la realización de huecos para el trasplante.
- Lograr la disminución de riesgos ambientales.



(a)



(b)



(c)



(d)

Figura 1.8: Se pueden ver algunas muestras de lo que hoy en día la investigación esta desarrollando con el objetivo de la agricultura de precisión. (a)CT801 (YANMAR). Diseñado en Japón para su uso en los campos de arroz [34][35]. (b) El tractor guiado automáticamente y el sistema de registro automático para superficies rugosas sobre la superficie de grava [9]. (c) Orpheus-X3 mobile robot. Forma parte de un sistema heterogéneo de reconocimiento [15]. (d) Un Drone volando para la exploración aérea de varias parcelas de trigo. [18]

- Reducir costes, ya que se disminuye la cantidad de combustible y de productos utilizados en algunas tareas.
- Elevar la calidad de los productos como por ejemplo, la utilización de menos pesticidas.



Figura 1.9: Robocrop, tractor robotizado de la empresa Tillett and Hague Technology Ltd., para eliminar la maleza de forma mecánica sin dañar el cultivo útil [18].

Llegando así al Guiado semiautomático y al Sistema autopiloto de guía.

**Guiado semiautomático** Consiste en la incorporación a la maquinaria que realiza las operaciones en el campo, de un sistema de posicionamiento DGPS que posibilita el conocimiento en cada momento de su posición. Con la ayuda de un panel gráfico, es posible trazar líneas rectas perfectas para trabajar la tierra [18] siendo un ejemplo el que se muestra en la Figura 1.10.

**Sistema autopiloto de guía** Un piloto automático es un sistema electro-mecánico e hidráulico, utilizado para guiar un vehículo sin la asistencia de un ser humano como se muestra en la Figura 11. Actualmente este tipo de equipos es utilizado con el apoyo de un sistema de posicionamiento del tipo RTK que permiten direccionar

al tractor con altísimas precisiones, evitando los errores en las líneas de siembra y pulverizaciones al igual que en la Figura 1.12.

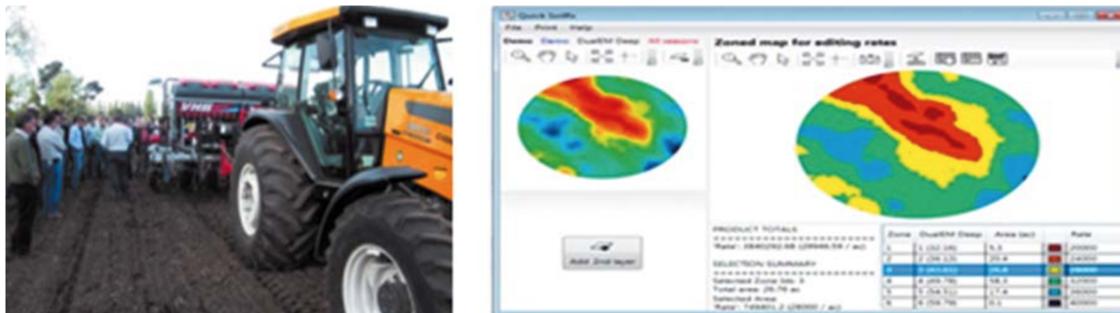
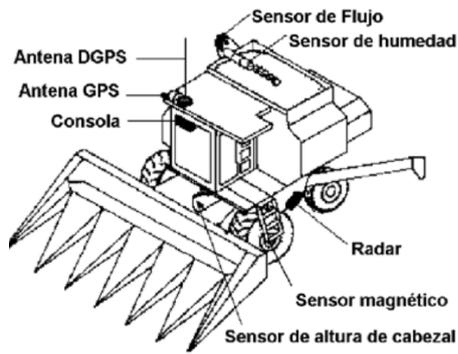


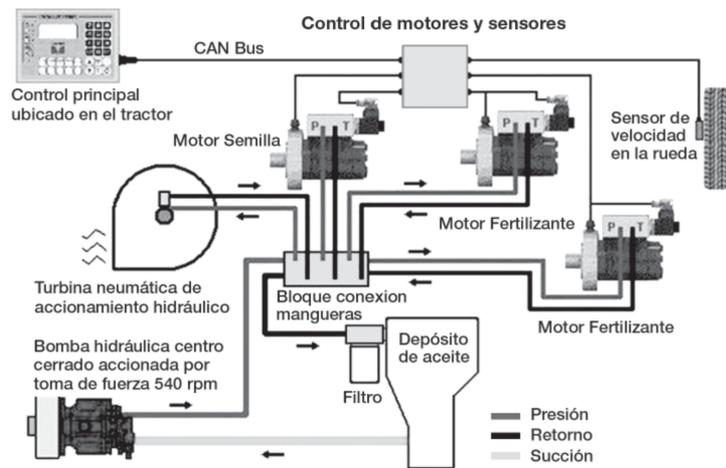
Figura 1.10: Sistema de monitoreo y de siembra variable montado en una sembradora de la empresa BHV, dinámica presentada en Chile, (derecha) y sistema incorporado en plataforma digital (izquierda). [18]

Siendo un sistema de este tipo el encargado de tareas como lo son las de un Banderillero satelital (usado principalmente en pulverizadoras autopropulsadas y en aviones aplicadores) o el de una sembradora inteligente. Los cuales entre sus principales características ofrecen:

- Un sistema auto guía o piloto automático por GPS encargado de que el sistema siga una trayectoria determinada en el área de aplicación.
- Son capaces de variar de manera simultánea e independiente la densidad de siembra y la dosis de 2 tipos de fertilizante mediante monitores GPS que trabaja como navegador y actuador (Figura 1.13).
- Seleccionar la prescripción de semilla y/o fertilizante variable dentro del lote a sembrar con sus correspondientes coordenadas GPS.
- Monitorear la siembra y transmitir la información en tiempo real a un teléfono celular o a una página de Internet, con el fin de actuar antes de que se haya originado algún desperfecto (Figura 1.14).



(a)



(b)

Figura 1.11: Instrumentación convencional para vehículos dedicados a la agricultura de precisión. (a) Sensores necesarios para el control del rendimiento [9]. (b) Esquema del circuito del sistema de la sembradora IOM Agrometal - Verion inteligente [17].



Figura 1.12: Sistema autopiloto montado en tractor y siembra realizada con autopiloto [18].

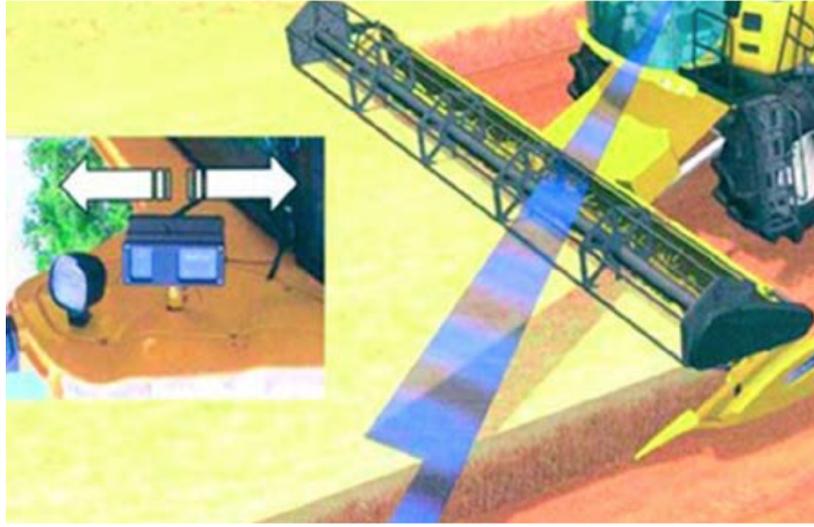


Figura 1.13: Sistema de auto guiado para cosechadoras: SmartSteer [18].

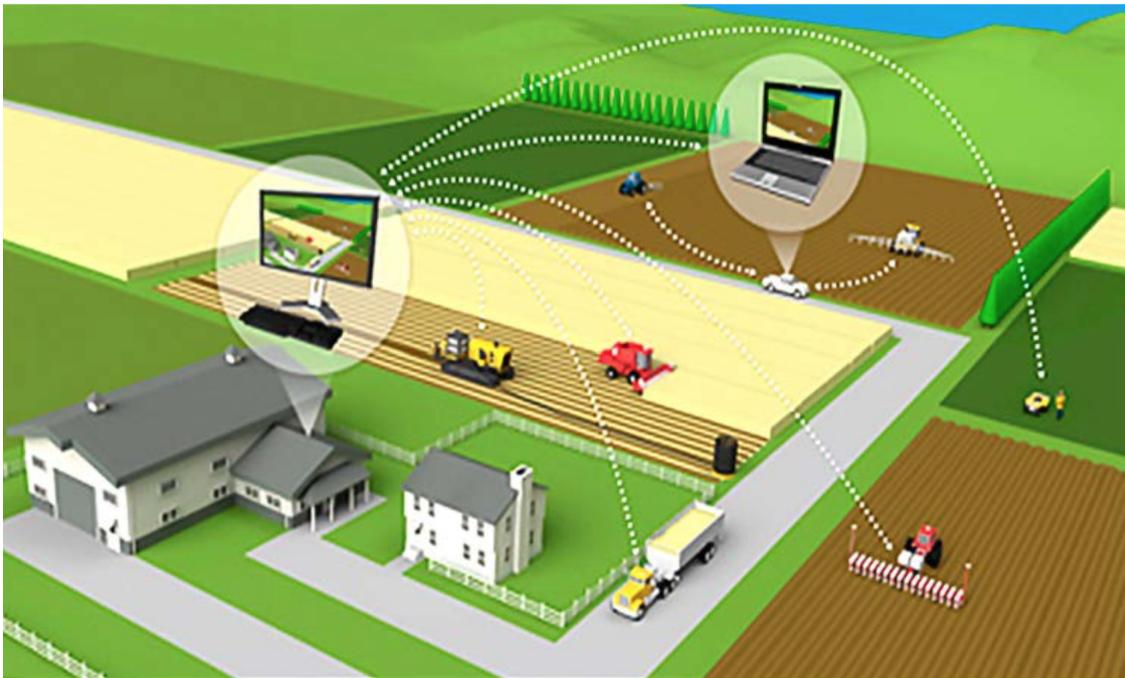


Figura 1.14: Esquema de funcionamiento de Connected Farm [18]

## 1.5. Organización del trabajo

A continuación se es presentada una breve descripción de como se irán abordando los puntos tratados a lo largo de este trabajo.

1. Introducción: Se involucra al lector un poco con los temas que se presentan al hablar de Agricultura de precisión, Vehículos terrestres, Sistemas de navegación y la Presencia de Sistemas de Navegación en la AP.
2. Instrumentación de plataforma: Se muestra el como esta constituido el sistema completo, así como una sección donde se muestran las características mas relevantes de los elementos que le conforman.
3. GPS-RTK: Siendo que el principal sensor de navegación es un Sistema de Posicionamiento Global de tipo RTK se considero necesario el incluir un apartado en el que se hable de su principio de funcionamiento, y factores que afecten su desempeño. Así como mostrar la intervención de este en el proyecto.
4. Modelado: Ya que se pretende la navegación autónoma de un vehículo terrestre, se es incluida una sección en la que por medio del modelado matemático se presenta el comportamiento del mismo. Modelo que posteriormente será considerado al incluir una ley de control para el Sistema de Navegación.
5. Sistema de navegación: El como se constituye y desglosa la arquitectura empleada al momento de realizar el algoritmo e navegación.
6. Simulación: Una vez se ha conocido el comportamiento del vehículo en si, así como la metodología con que este realiza su tarea de navegación, se procede a realizar un proceso de pruebas mediante simulación para validar su comportamiento.
7. Resultados experimentales: concluyendo con el trabajo, se procede a someter el sistema a prueba en el exterior con el fin de comprobar en tiempo real el correcto funcionamiento del algoritmo desarrollado.



---

# Instrumentación de plataforma

---

Partiendo por mostrar como es que esta conformado cada bloque del sistema Vehículo y Base (GCS). Posteriormente, buscando la justificación de porque se han empleada cada uno de los elementos (hardware y software) se muestran sus principales características.

## 2.1. Estación base (GCS)

Adelantando un poco al contenido de el Capitulo 3, siendo que, una de las principales características que plantea el trabajar con sistemas de tipo RTK es que este estará conformado por un GPS receptor denominado Rover el cual se encontrara a bordo del vehículo en cuestión y uno mas, que con una ubicación fija proporcionara correcciones de posición al cual se denominara base. Siendo así este GPS-Base formara parte de un bloque del sistema el cual fungirá como GCS, es decir, como un Centro de control o monitoreo en tierra.

A este GCS se le asignara la tarea de ademas de Asistir en el proceso de correcciones en el sistema RTK de realizar un monitoreo constante de las tareas que estará realizando el vehículo-rover, así como de también dar la posibilidad de controlar desde esta posición al vehículo ya sea asignando tareas o asta de así requerirse pasar a un modo de control manual. Un esquema de los elementos que conforman al GCS son mostrado en la Figura 2.1.



Figura 2.1: Diagrama de interacción de componentes en GCS.

### 2.1.1. Elementos que componen GCS

Separados en dos partes Hardware y Software La lista de componentes es presentada a continuación:

#### *Hardware*

PIKSI (GPS-RTK + Antena)	1
SiK Telemetry Radio	2
PC	

#### *Software*

Piksi Console  
 MissionPlanner  
 Emulador de terminal SSH

Ahora la función de estos componentes se asigna de la siguiente manera:

- Tanto Piksi como la Piksi Console se encargaran uno de proporcionar correcciones al Rover mediante el uso del dispositivo SIK Telemetry mientras el otro de permitir monitorear la salud de la señal (Satélites en común, intensidad de la señal, estado de proceso RTK, monitoreo de posición relativa del Rover, etc.) (Figura2.2).
- La tarea a Realizar por Mission Planner asistido por el otro dispositivo SIK Telemetry será la de monitorear la posición global del vehículo con respecto a la tierra, así como de mantener informado del correcto funcionamiento del autopiloto principal (Pixhawk) el cual se encarga del control de los actuadores así como de la interpretación de los datos obtenidos por los sensores (Figura 2.3).

- A manera de prevención de errores con el sistema de navegación autónomo se ha añadido un mando RC con la finalidad de pasar a control manual de así requerirse.
- Sumando a lo ya mencionado, el agregar una interfaz de Terminal SSH y una red virtual a partir de la misma PC fue con la finalidad de poder monitorear de manera directa el OS dentro del autopiloto mediante el computador de tipo ARM con que ha sido provisto el vehículo. Dando así la una versatilidad mayor a la que ofrece Mission Planner. Sumando la capacidad de poder modificar parámetros en plena tarea, he incluso agregar módulos de control según sean las necesidades como se muestra en la Figura 2.4.

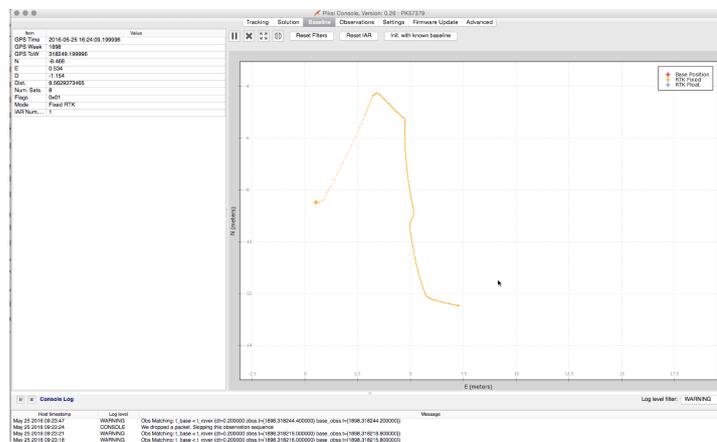


Figura 2.2: Ventana de Piksi Console al monitorear posición relativa de GPS Rover al este estar en desplazamiento y encontrarse bajo estado RTK-FIX.

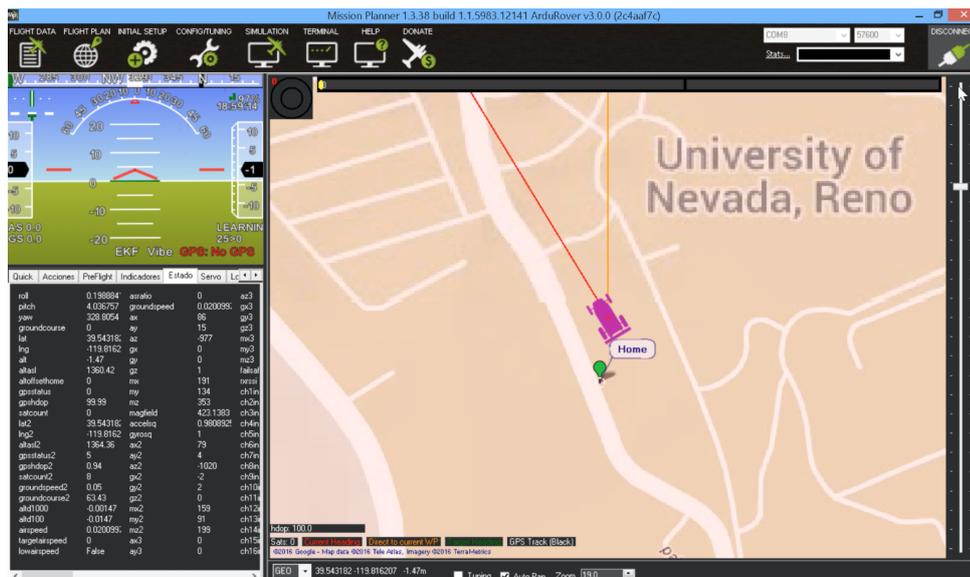


Figura 2.3: Ventana de Mission Planner al realizar monitoreo de posición global del vehículo al este estar en desplazamiento. Prueba realizada en las instalaciones de UNR.

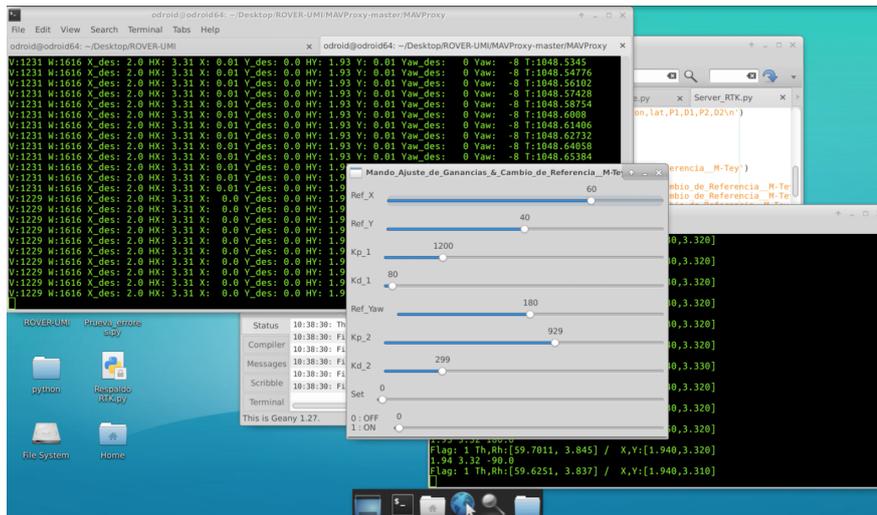


Figura 2.4: Modulo de control agregado para realizar ajuste de ganancias y asignar referencias de prueba, así como mantener un monitoreo en tiempo real de la dinámica del vehículo. Solo accesible mediante el Servidor SSH.

## 2.2. Vehículo (Rover)

Siendo el vehículo el segundo bloque del sistema. Se ha optado por elegir uno con la configuración Skid Steer debido al buen desempeño que estos suelen presentar en diversos terrenos. Este vehículo es provisto de dos actuadores (motores) los cuales ayudados de un sistema de transmisión se encargan de el movimiento de ambos grupos de ruedas (izquierdo y derecho) respectivamente.

### 2.2.1. Elementos que componen al vehículo (Rover)

Al igual que como se realizo con el GCS estos elementos serán divididos y presentados como Hardware y Software.

#### *Hardware*

Odroid-C2	1
Pixhawk	1
Driver Sabertooth	1
PIKSI (GPS-RTK + Antena)	1
RPLidar360	1
SiK Telemetry Radio	2
Adaptador USB Inalámbrico de Alta Ganancia 150Mbps	1

AR8000 8-Channel DSMX Receiver	1
RC SPEKTRUM DX7	1
Lipo Battery Tester Monitor	1

Donde la interconexión es realizada como muestra la Figura 2.5.

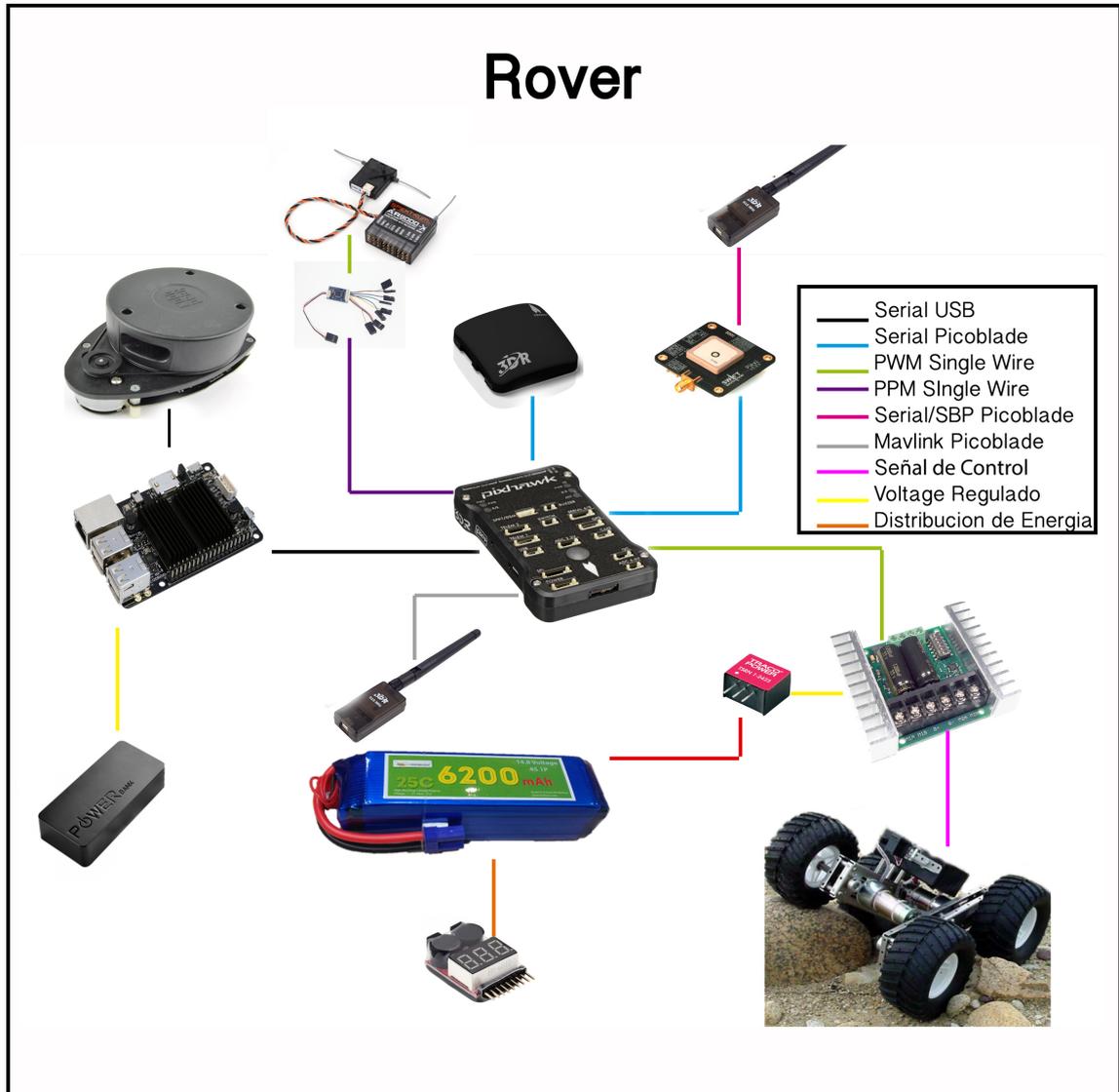


Figura 2.5: Esquema de elementos que conforman al vehículo (Rover).El cuadro que conforma el esquema muestra la forma en que se comunica cada dispositivo con los demás.

### *Software*

Python (Interprete)	MAVProxy/ Opencv
Protocolos de comunicación	Serial / TCP/IP(Sockets) / SSH

Este vehículo fue instrumentado para cumplir con los siguientes bloques (Figura 2.6):

1. Adquisición de datos.
2. Procesamiento y sistema de navegación.
3. Acoplamiento.

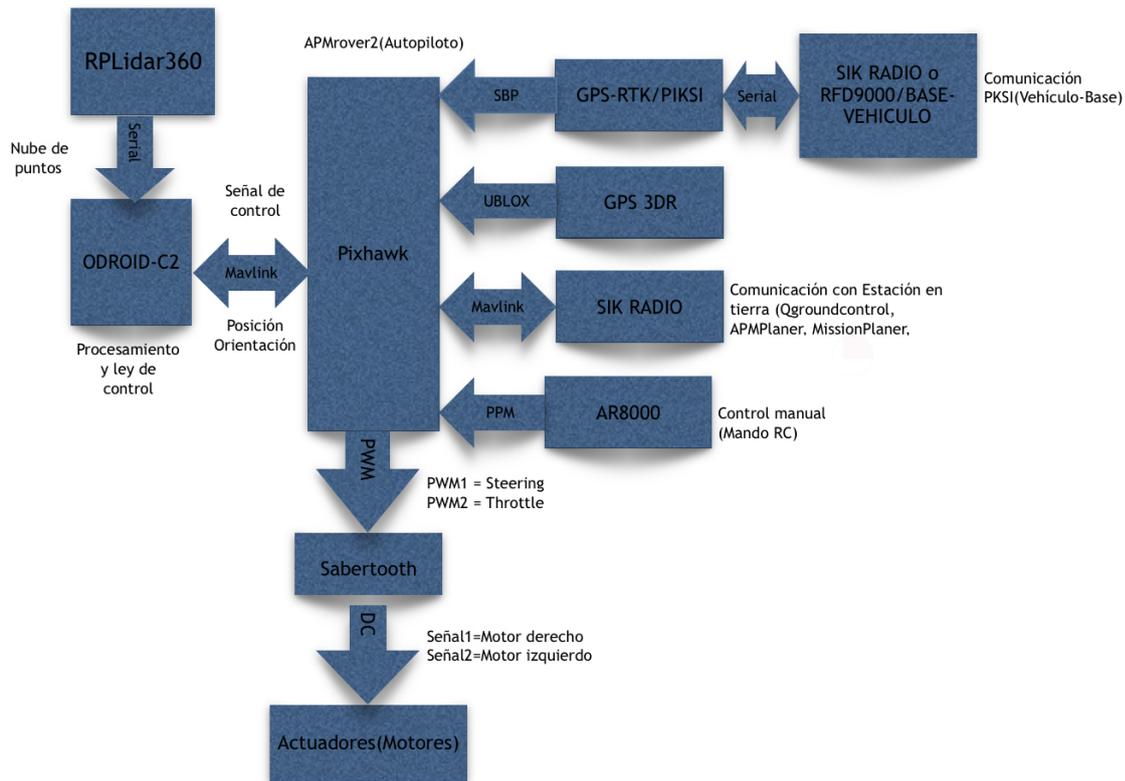


Figura 2.6: Diagrama de interacción de componentes en el vehículo.

### 2.2.2. Descripción del vehículo

Como ya se menciona el deberá ser capaz de cumplir con tres tareas principales, siendo estas:

1. Adquisición de datos.
2. Procesamiento y sistema de navegación.
3. Acoplamiento.

### 2.2.2.1. Adquisición de datos

Comenzando por la adquisición de datos. El vehículo consta de dos sensores principales

1. GPS-RK.
2. Compass (brújula digital).

así como un tercero y no menos importante un Lidar.

Siendo considerados los dos primeros como los mas importantes, ya que estos serán los encargados de dar información sobre la ubicación y orientación del vehículo, mientras que el Lidar servirá para realizar una inspección del entorno (en un radio determinado) y prevenir posibles colisiones (Figura 2.7).

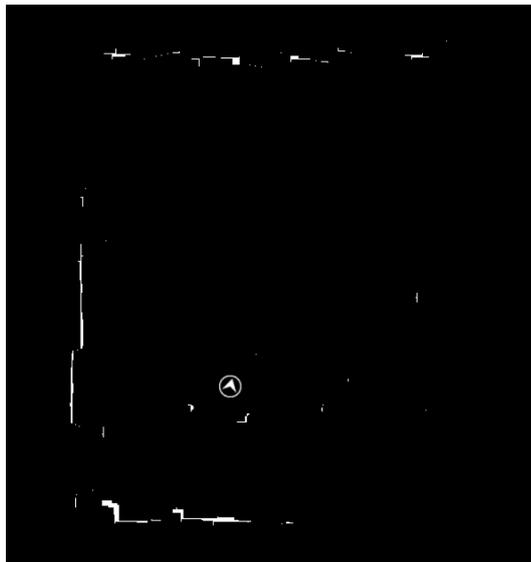


Figura 2.7: Visualización de entorno detectado por vehículo mediante lidar. Modulo creado para trabajar en combinación con el GPS y compass.

Otro sensor que igual conforma esta lista es el AR8000 (receptor de RC) el cual mediante codificación PPM se encargara de informar al vehículo de las señales emitidas por el mando RC al momento de pasar a modo manual.

También se hace mención que la implementación del lidar fue con fines de trabajos futuros, quedando como único antecedente el modulo visto en la Figura 2.7, el cual se encarga mediante fusión de datos entre Lidar, compas y GPS de realizar un bosquejo 2D del entorno que lo rodea y simultáneamente crear un mapa.

### 2.2.2.2. Procesamiento y sistema de navegación

Ahora, siguiendo con la tarea de Procesamiento y sistema de navegación, habrá que definir tres puntos importantes para realizar esta tarea. Es necesario contar con un Lenguaje, un Medio y un Canal, entendiendo estos como:

**Lenguaje:** Será el encargado de estandarizar el código con el que se realizara todos los procesos.

**Medio:** La forma en que se plante el flujo de datos dentro de la arquitectura del proceso.

**Canal:** Será el protocolo que se encargue de la petición entre dispositivos y el centro de procesamiento.

Tomando en cuenta lo anterior una de las herramientas que ayudara al facilitar estas tareas será el uso de una tarjeta de desarrollo (Odroid-C2) la cual de manera análoga a una computadora normal es capaz de realizar tareas complejas sin demandar un gran tamaño. Otro elemento clave es el uso del autopiloto Pixhawk. El cual a parte de contar con una IMU ofrece una gran variedad de firmware especializados a distintos vehículos (Multi-rottores, Ala fijo o Terrestres) dando también la posibilidad de crear un firmware propio. Tomando de estas opciones la de utilizar el firmware que este nos ofrece (APMrover2.0), puesto que lo que se busca es crear un sistema bajo la premisa Plug and Play.

- Bajo el creciente uso del lenguaje interpretado Python así como un gran numero de documentación, es que se opta por tomar este como Lenguaje principal al realizar el código que se encargara de gobernar al vehículo.
- Siendo que lo que se busca es diseñar un sistema fácil de manipular y que de igual forma aproveche todos los recursos brindados. Se opta por basar el diseño en una arquitectura modular. Donde por medio de Sockets TCP/IP se maneja el flujo de datos entre módulos, los cuales serán tratados como Servidor y Clientes.
- Teniendo ya un lenguaje (Python), y un medio de comunicación (socket) se es necesario determinar el canal por el que se realizara la comunicación entre los distintos dispositivos. El cual, por las características de estos es elegido como canal el protocolo de comunicación serial.

Teniendo ya los puntos necesarios, es posible describir el como trabajara el Sistema de procesamiento.

1. Se definen las tareas: Manipulación de Pixhawk, lectura de sensores, comunicación con GCS.
2. Crear módulos de operación: Interfaz con Pixhawk-Sistema de navegación, Procesamiento con Lidar, Procesamiento con GPS-RTK.
3. Definir Servidor y Clientes: Servidor (Interfaz con Pixhawk-Sistema de navegación), Clientes (Modulo Lidar-GCS y modulo GPS-RTK-GCS).

En resumen a lo anterior, el modulo encargado de navegación y administración de la información será a su vez l encargado de la manipulación del Autopiloto (Pixhawk), mientras que el procesamiento del Lidar y del GPS serán tratados como subprocesos los cuales se comunicaran con el servidor para complementar el sistema de navegación (Figura 2.8).

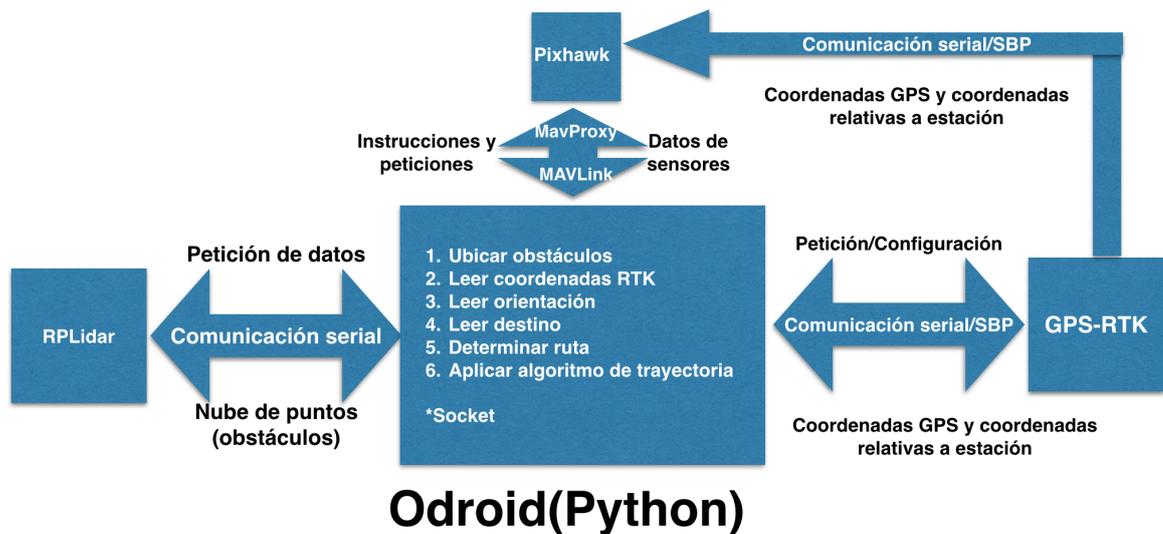


Figura 2.8: Diagrama de procesamiento realizado a bordo del vehículo.

### 2.2.2.3. Acoplamiento

Llegado a este punto, el Acoplamiento será el encargado de interpretar las señales y ejecutar las ordenes mediante los actuadores (sistema de motores y transmisión).

Teniendo el inconveniente de que el Pixhawk trabaja con una configuración de señales para un vehículo de tipo Ackerman, es que se opta por utilizara el driver Sabertooth 2x25

V2. El cual bajo sus características permite interpretar señales de control de tipo PWM como las generadas por el Pixhawk y mediante una función de mixer convertir un par de señales de tipo Ackerman a una salida compatible con vehículos de tipo diferencial o lo que es igual, de tipo Skid Steer.

Teniendo así el como se comporta el sistema completo (Figura 2.9).



Figura 2.9: Foto de sistema RTK.

## 2.3. Descripción de componentes

A continuación se muestra una breve descripción de los componentes presentes en el sistema.

### 2.3.1. Odroid-C2

Dispositivo basado en una arquitectura de 64 bits (Figura 2.10), capaz de realizar tareas avanzadas, soportando la ejecución de distribuciones Linux o bien de así requerirse una versión ARM de Windows 10. Entre sus principales características se encuentran:

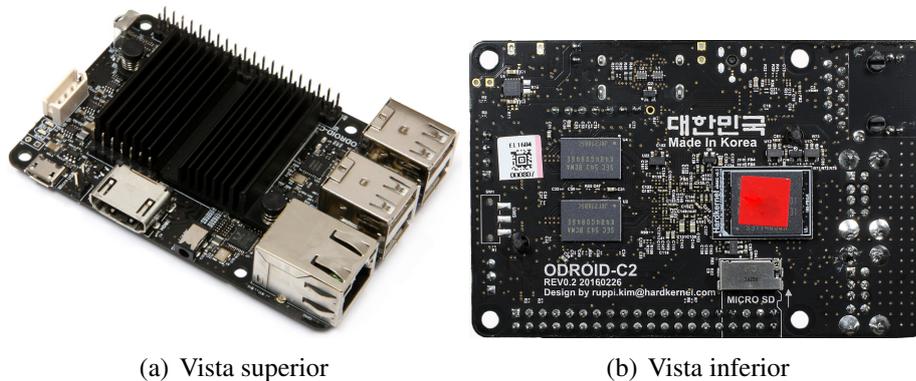


Figura 2.10: Odroid-C2 .

- Procesador Amlogic ARM® Cortex®-A53(ARMv8) 1.5Ghz quad core.
- GPU Mali-450 GPU
- SDRAM 2Gbyte DDR3
- 40pin GPIOs + 7pin I2S.
- USB 2.0 Host x 4, USB OTG.

### 2.3.2. PIXHAWK

Sistema de autopiloto (Figura 2.11)avanzado diseñado por el proyecto PX4 en hardware abierto y fabricado por 3D Robotics. Cuenta con procesador avanzado y tecnología de sensores de ST Microelectronics® y un sistema operativo de tiempo real NuttX, que ofrece un rendimiento increíble, flexibilidad y fiabilidad para controlar cualquier vehículo autónomo.



(a) Vista superior



(b) Vista lateral(salidas PWM)

Figura 2.11: Pixhawk .

- Procesador STM32F427 Cortex M4 core con FPU
- Memoria FLash de 2 MB/ RAM 168 MHz/256 K
- Co-procesador 32 bit STM32F103 a prueba de fallos
- 14 PWM (8 a prueba de fallos y manual sobreescritura, 6 auxiliares)
- 5x UART(Puertos Seria), 2x CAN, SPI, I2C®, puerto microUSB externo, entrada RSSI(PWM o Voltaje), ADC(3.3V y 6.6V), entrada PPM.Compatible con Satélite Spektrum DSM/DSM2/DSM-X® con entrada a DX8.Compatibilidad con E/S Futaba S.BUS®.
- Botón externo de seguridad para fácil encendido de motores.
- Giroscopio ST Micro L3GD20 3-axis 16-bit.
- ST Micro LSM303D 3-axis 14-bit acelerometro / magnetómetro
- Acelerómetro/giróscopo MPU 6000 3-axis.

### 2.3.2.1. Magnetometro (brújula digital)

Se llama magnetómetro a los dispositivos que sirven para cuantificar en fuerza o dirección la señal magnética de una muestra. El magnetismo varía de lugar en lugar y a las diferencias en el campo magnético terrestre (la magnetósfera) y puede ser causada por las diferentes naturalezas de las rocas y la interacción entre las partículas cargadas del Sol y la magnetósfera de un planeta.

### 2.3.3. Piksi-RTK

Receptor GPS (Figura 2.12) de bajo costo y alto rendimiento con RTK(Real Time Kinematics) funcional para posicionamiento relativo con precisión de centímetros. Con un factor de forma pequeño y bajo consumo de energía al actualizar datos.

- Precisión: 1-5cm (Posición Horizontal) y 8-15cm (Posición Vertical).
- Velocidad de solución a 10 Hz.
- Software y diseño de tarjeta Open-source.
- Bajo consumo - 500mW
- Conectividad mediante USB y dual UART
- Entrada para conexión de antena externa

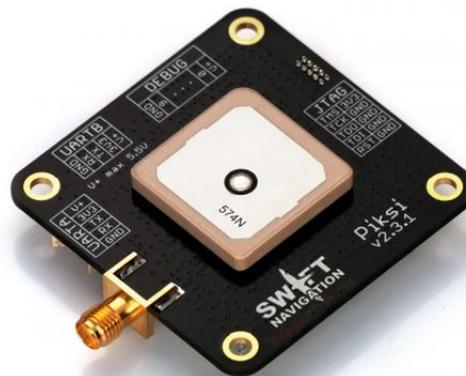


Figura 2.12: Dispositivo Piksi de Swift NAV

### 2.3.4. Sabertooth 2x25 V2

El Sabertooth 2X25 (Figura 2.13) resulta ser uno de los drivers de motor dual mas versátiles, eficientes y fáciles en el mercado. Es adecuado para los robots de alta potencia. Permite controlar dos motores con: tensión analógica, control de radio, serie y serie por paquetes. Sabertooth tiene modos de funcionamiento independientes dirección y velocidad, por lo que es ideal para robots de tipo diferencial. Donde la opción de entrada R/C es usada para la lectura de un radio estándar (pwm 50 80Hz) como señales de control y

pasando por la opción Mixed la cual pasa a convertir de throttle y turning a una mezcla para obtener una configuración de control para un robot diferencial.

- Modos de entrada: analógica, R / C, serie simplificada o serie en paquetes.
- 25A continuo, 50A pico por canal.
- 25A, 50A máximo por canal 6-30V nominal, 33.6V máximo absoluto continua.
- Unidad regenerativa síncrona.
- Frecuencia de conmutación ultrasónico.
- Protección térmica y de sobrecorriente.
- Modo de protección de litio.

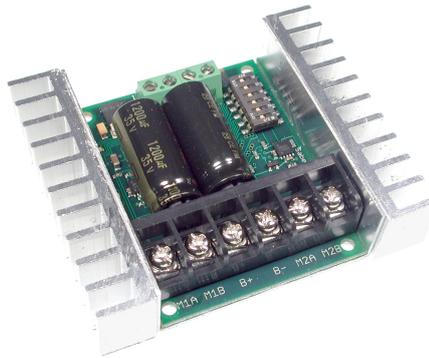


Figura 2.13: Driver Sabertooth 2x25 V2

### 2.3.5. RPLidar 360

El escáner láser RPLidar 360 (Figura 2.14) es un escáner de 360 grados 2D a bajo costo.

- Escaneo láser omnidireccional.
- Configuración de velocidad de barrido vía PWM del motor.
- Opera a 5.5Hz (2000 muestras/segundo).
- 6m como rango de operación.
- Nube de puntos en formato de coordenada polares.



Figura 2.14: RPLidar 360

### 2.3.6. Telemetría Sik

La telemetría Sik (Figura 2.15) ofrece una fácil configuración para la comunicación y monitoreo a distancia entre una estación terrestre(GCS) sea APMplaner, MissionPlaner, etc., y un autopiloto tal como PIXHAWK.

- Disponible en frecuencias de 900Hz o 433Hz.
- Recepción sensible a -121 dBm.
- Poder de transmisión arriba de 20dBm (100mW).
- Vinculación serial en modo transparente.
- Transmisión de datos sobre 250Kbps.
- Firmware libre.
- Configuración de radio vía comandos AT.
- Configuración remota vía comandos RT.
- Basado en el modulo de radio HM-TRP con microcontrolador Si1000 8051 y modulo de radio Si4432.

### 2.3.7. Receptor SPECKTRUM AR8000

- Velocidad de cuadro a 11 ms
- Resolución de 2048bits



Figura 2.15: Telemetria Sik

- Compatible con registro de vuelo y telemetría
- 8 canales
- Modulación: DSM2®/ DSMX®
- Voltaje de operación: 3.5-9.6V



Figura 2.16: Receptor SPECKTRUM AR8000

### 2.3.8. Traco Power TSR 1-2450

Siendo uno de los dispositivos (Figura 2.17) que por excelencia se encuentran en el mercado al momento de desear reguladores para alimentar dispositivos electrónicos con exigentes especificaciones, tal es el caso del controlador Sabertooth.

- Convertidor DC/DC: salida fija.
- Convertidor DC/DC: tipo SIP.
- Potencia máxima de salida: 5w.
- Voltaje de salida nominal: 5V.
- Corriente máxima de salida: 1A.
- Voltaje mínimo de entrada: 6.5V(DC).
- Voltaje máximo de entrada: 36V.



Figura 2.17: Traco Power TSR 1-2450

### 2.3.9. Bateria HeliBoss

- Capacidad mínima: 6200mAh
- Configuración: 4SIP/14.8V/4Cell
- Carga constante: 25C



Figura 2.18: Bateria HeliBoss (4S, 6200mA, 25C)

### 2.3.10. POWERBANK

Batería externa o power bank (Figura 2.19) es una batería extra o un cargador externo para cualquier aparato electrónico tal es el caso de dispositivos móviles (celular, tablet, etc.).

Propiedades:

- 4000 mAh.
- Output 5 Vcc 2Am.
- Input 5Vcc 1Am.



Figura 2.19: POWERBANK (4000mAh)

En este caso ha sido separado en cuatro secciones:

1. Sistema Operativo (SO)
2. Lenguaje de programación
3. Protocolos de comunicación
4. Software externo

### 2.3.11. Lenguaje de programación

**Python** es un lenguaje de programación interpretado cuya filosofía es basada en una sintaxis que favorezca un código legible, usa tipado dinámico y es multiplataforma. Soporta:

- Orientación a objetos
- Programación imperativa
- Programación funcional

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python SoftwareFoundation License, que es compatible con la Licencia publica general de GNU a partir de la versión 2.1.1.

## **2.3.12. Protocolos de comunicación**

### **2.3.12.1. Secure Shell (SSH)**

SSH (Secure SHell, en español: intérprete de órdenes seguro) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos, y también puede redirigir el tráfico de X (Sistema de Ventanas X) para poder ejecutar programas gráficos si tenemos ejecutando un Servidor X (en sistemas Unix y Windows).

### **2.3.12.2. Comunicación TCP/IP por SOCKET**

El uso de este protocolo fue empleado tomado como referencia la estructura de ROSS. Donde mediante clientes y servidores [6][3][32]ejecutados de manera simultánea en distintos procesos es que se logra una equilibrada sincronización en el flujo de datos. Siendo así que se planteo una estructura donde se mantienen en ejecución diferentes procesos encargados cada uno de tareas particulares siendo estas desde lectura de sensores, interfaz gráfica de manipulación, algoritmo de navegación, etc. Las cuales cada cierto tiempo se comunican con un código principal encargando de reunir toda la información.

El socket designa un concepto de comunicación por el cual dos programas pudiendo ser localizados dentro de la misma computadora o de así ser en distintas computadoras pueden realizar intercambio de información (flujo de datos).

Para que la comunicación entre dos programas pueda ser realizada es necesario cumplir con algunos requisitos:

- Que los programas sean capaces de localizarse.
- Que los programas sean capaces intercambiar datos acorde a su finalidad.

Para ello son necesarios dos recursos que originan el concepto de socket:

1. Un par de direcciones IP que identifiquen la computadora de origen y la remota.
2. Un par de números de puerto, que identifican a un programa dentro de cada computadora.

Las propiedades del socket serán determinadas por el protocolo en que se implementen. Siendo el más utilizado el TCP y teniendo como alternativa el UDP

Al llevarse a cabo bajo el protocolo TCP, los sockets cumplen con las siguientes propiedades:

- Son orientados a la conexión
- Se garantiza la transmisión de todos los paquetes sin errores ni omisiones.
- Se garantiza que cada paquete será recibido en el mismo orden en que fue enviado

Los sockets permiten crear una estructura cliente-servidor (Figura 2.20). Donde la comunicación dará inicio por uno de los programas al que se denomina programa cliente. Mientras que el segundo se mantendrá en espera hasta que el cliente inicie la comunicación, denominando a este como programa servidor.

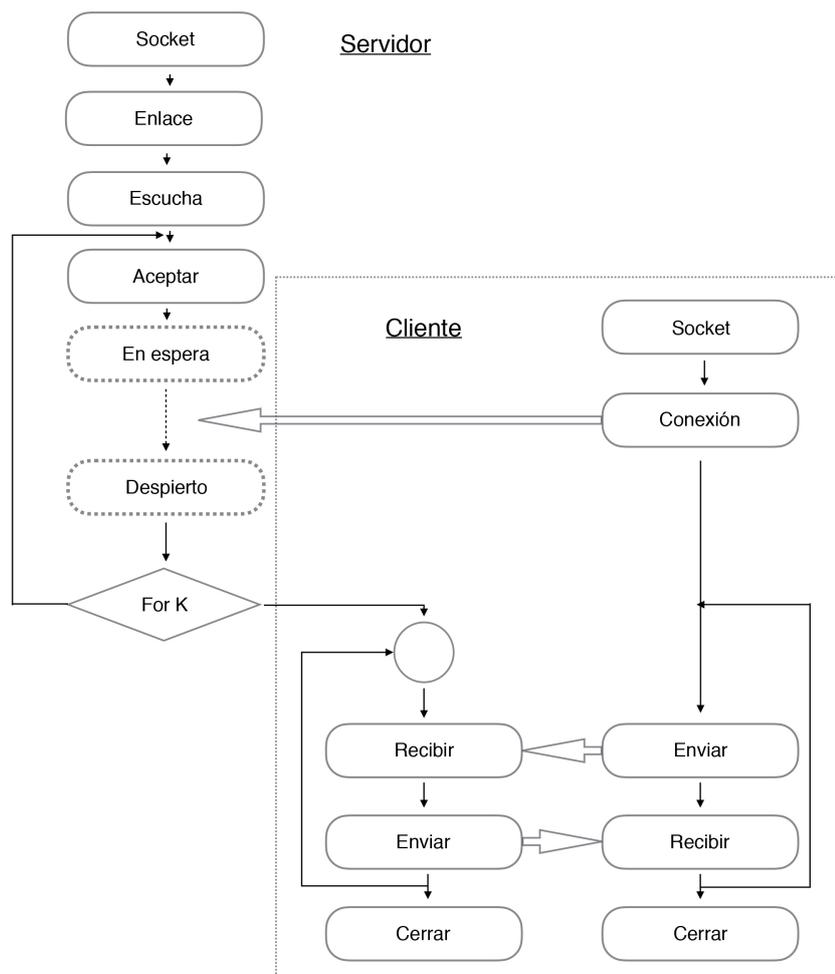


Figura 2.20: Proceso de comunicación a través de Sockets

### 2.3.13. Piksi Console

La Consola GUI (Figura 2.21) por sus siglas en inglés (Graphic User Interfase) es una representación gráfica de los procesos que lleva a cabo el dispositivo Piksi. Esta muestra información y así mismo permite realizar ajustes a el Hardware.

La consola esta formada por siete pestañas:

- Tracking: muestra que señales de satélite esta siguiendo el dispositivo Piksi.
- Solution: Proporciona la posición absoluta del dispositivo Piksi.

1 Seguimiento: muestra qué señales se Piksi su seguimiento por satélite. 2 Solución: proporciona la posición absoluta de su Piksi. 3 Línea de base: da la RTK vector de alta precisión entre sus dos Piksisis. 4 Observaciones: muestra la información que su Piksisis dos están enviando a la otra. 5 Configuración: le permite ver y modificar la configuración de dispositivos de su Piksi. 6 Actualización de firmware: le permite ver y actualizar las versiones de su software Piksi. 7 Avanzado: proporciona algunas funciones especializadas para usuarios avanzados (Sistema de monitor, PAS de retransmisión, la consola de Python).

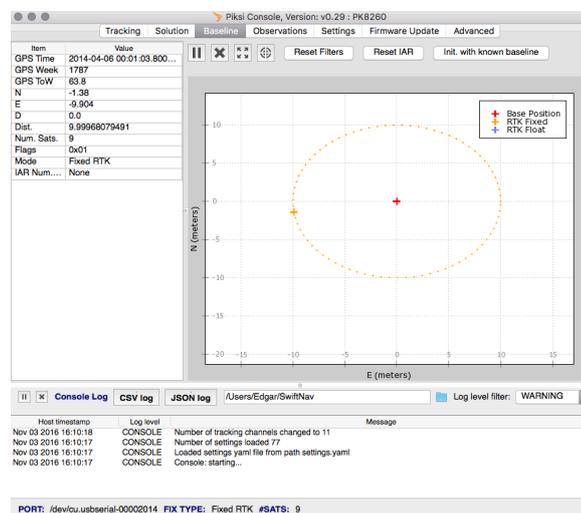


Figura 2.21: Consola de Piksi (se muestra la ventana de Baseline una vez se a obtenido RTK aclarando que la presente se encuentra en modo simulación)

### 2.3.14. MissionPlanner

Mission Planner es un GCS (Figura 2.22) para Planeadores, Multirrotores y Rover. Siendo este únicamente compatible con Windows. Mission Planner puede ser utilizado

como una herramienta de configuración o como un complemento para un control dinámico sobre el vehículo. Mostrando un poco de lo que se puede hacer con este:

- Carga de firmware (APMrover2) a el autopiloto (PX4) el cual controlara a el vehículo.
- Instalar, configurar y ajustar propiedades (software ) para un optimo funcionamiento.
- Planeación, guardar y cargar misiones automáticas.
- Descargar y analizar mission logs creados por el autopiloto.
- Así mismo siendo el uso de una telemetría el monitoreo de status del vehículo en operación.



Figura 2.22: Mission Planner (ventana de inicio)

### 2.3.15. MAVProxy

Este paquete de software inicialmente es GCS para los sistemas UAV basado en MAVLink. MAVProxy es un GCS de total funcionamiento para UAV. Cuya intención es ser un GCS minimalista, portátil y extensible para cualquier UAV que soporte el protocolo MAVLink (como el APM).

Características:

- Se trata de una aplicación basada en consola y linea de comandos.
- Puede ser conectado en red y ejecutar a través de cualquier número de ordenadores.

- Es portable; por lo que es posible ejecutar en cualquier SO POSIX con Python, Pyserial, y selección de llamado de función, lo que implica Linux, OS X, Windows y otros.
- El diseño ligero significa que puede funcionar en netbooks pequeños con facilidad.
- Es compatible con la carga de módulos, y cuenta con módulos para el soporte de consola / s, mapas móviles, joysticks, rastreadores de antena, etc.

*Licencia:* MAVProxy se distribuye bajo la Licencia Pública General de GNU versión 3 o posterior



---

# GPS-RTK

---

Navegación satelital por Real-Time Kinematics (RTK) es una técnica usada para afinar la precisión de un GNSS [10][21][13][15][30][1]. EN lugar de solo usar la información contenida en las señales recibidas, es también utilizada la medición de fase de la onda portadora, Resultando en una precisión de centímetros. Sin embargo, esta precisión es solo lograda con la posición relativa de dos receptores, siendo la comparación de mediciones de dichas fases necesarias para remover el error

Optando por la incursión en sistemas de posicionamiento global es que se elige por dotar al sistema con un sistema de este tipo RTK. El cual, gracias a su precisión dará una mejor percepción al vehículo optimizando el sistema a la hora de navegar, permitiendo así la opción de resolver tareas con mayor precisión.

## 3.1. Concepto general de GPS

El Sistema de Posicionamiento Global (GPS por sus siglas en inglés) es una constelación formada por 24 satélites, los cuales actúan como punto de referencia tal que un receptor GPS puede calcular la información "precisa" de su posición. Siendo el sistema propiedad, operado y controlado por el Departamento de Defensa de los Estados Unidos (DoD). Sin embargo, este no queda restringido dando la posibilidad de ser usado por cualquier civil en todo el mundo.

### 3.1.1. Bloques de una estructura GPS

Siendo un sistema GPS conformado por tres bloques:

1. Espacio: Consta de un mínimo de 24 NAVSTAR operacionales. Donde los satélites

son agrupados en 56 órbitas, cada una con inclinación de 55 grados con respecto al plano ecuatorial. Cada satélite transmite una señal de radio marcada con un único código de identificación.

2. Control: El bloque de control lleva la tarea de operar como el "cerebro" del GPS. Siendo el departamento de Defensa de los Estados Unidos el que controla el sistema utilizando una estación maestra y cuatro estaciones mas.
3. Usuario: Toda aquella persona con un receptor GPS le es permitido usar el sistema GPS, resultando la señal usada tanto por civiles como por militares. Sus aplicaciones incluyen topografía, agricultura, aviación, servicios de emergencia, recreación y vehículos de seguimiento.

### 3.1.2. Estructura de la señal de satélite

Cada satélite transmite dos únicos códigos. El primero, y mas simple código es llamado el código C/A (por sus siglas en inglés Coarse Acquisition). El segundo código es llamado el código P (Precise). Estos códigos son modulados en dos ondas portadoras, L1 y L2 (Figura 3.1).

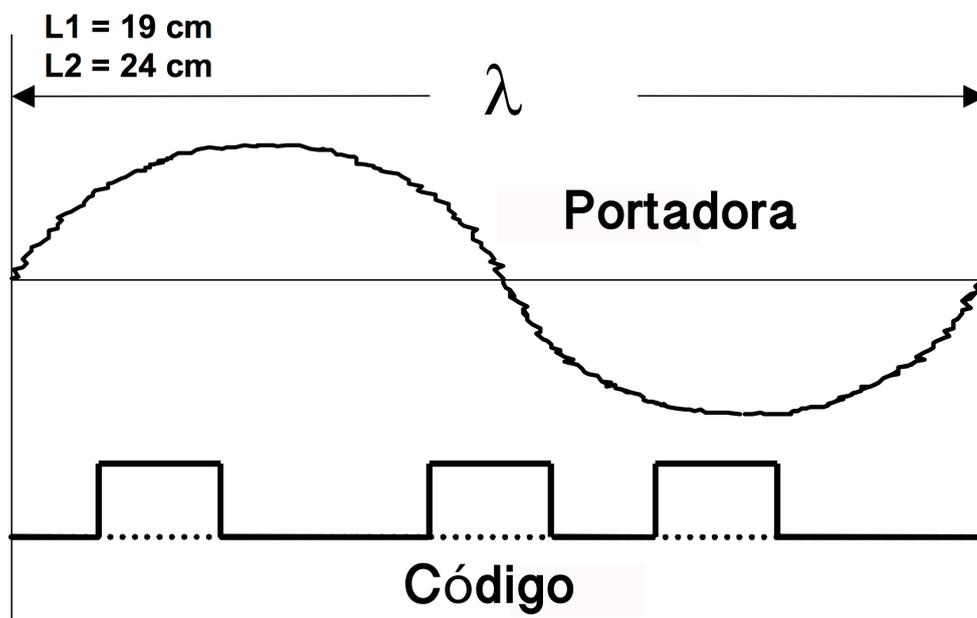


Figura 3.1: Estructura de la Señal de Satélite. Relación entre Código y señal portadora .

L1 es la primer L-band portadora, con una longitud de onda de 19cm y una frecuencia de 1575.42 MHz. Siendo esta es modulada por el código C/A, código P, o código Y, y

un mensaje de navegación de 50 bits por segundo. L2 es la L-band portadora secundaria con una longitud de onda de 24cm. y una frecuencia de 1227.6 MHz. Siendo esta modulada por código P o Y, y un mensaje de navegación de 50 bits por segundo. Código C/A es modulado en una señal L1. Siendo el código es un string de 1023 bits digitales en un patrón tan complejo que puede considerarse como aleatorio, por lo que resulta ser llamado como código pseudo-aleatorio (PRN por sus siglas en inglés) o ruido.

Los mensajes de Navegación están conformados un tiempo de sistema, un reloj de corrección de parámetros, un modelo ionosférico por retraso de parámetros, y detalles sobre el estado del satélite. La información es usada para procesar la señal GPS al obtener la posición de usuario y velocidad.

La posición de cada satélite se conocida es conocida en todo momento. Por lo que ellos actúan como "puntos conocidos." en el espacio. Conociendo la distancia(rango) del GPS APC (por sus siglas en inglés Antena Phase Center) cada uno de los satélites visibles por el receptor calculan esta posición. Siendo la posición resuelta por triangulación y después determinado el rango de cada satélite visible.

Siendo los rangos medidos usando el código o la fase de la onda portadora.

### 3.1.3. Rango de satélite basado en la medición de código

El receptor GPS tiene un código generador C/A que produce el mismo código PRN que el hecho por el satélite. El código recibido por el satélite es comparado con el código que es generado por el receptor, desfasando una replica del código asta que es correlacionada con el código SV. como se muestra en la Figura. 3.2

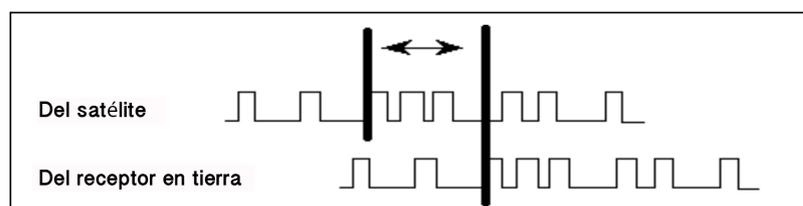


Figura 3.2: Medición de código, desfase presente entre señal del satélite y la generada por el receptor.

### 3.1.4. Rango de satélite basado en medición de fase portadora

Se tiene que al menos dos receptores siguen las señales portadoras a el mismo tiempo. La fase portadora es seguida por ambos receptores mientras el cambio de seguimiento de

fase es grabado todo el tiempo por ambos receptores al ser conocida la longitud de onda de L1 y L2

Al hecho de determinar el numero total de ciclos entre la antena y el satélite es conocido como Integer Ambiguity Search. Resolver el Integer Ambiguity resulta esencial para las peticiones que requieren precisión de nivel centimétrico.

Siendo un ejemplo de esto las peticiones por los sistemas RTK (por sus siglas en inglés Real-Time Kinematic), donde el receptor-rover resuelve el Integer Ambiguity y determina el Integer durante el proceso inicial.

El resultado de mediciones basadas en una phase portadora resulta como Baseline. Siendo esto resultado de tomar dos GPS observando señales simultáneamente . Cuando los datos de estos dos receptores es combinado, el resultado es un Baseline o vector 3D entre las dos estaciones. Siendo esto la posición de un GPS relativa a el otro.

### 3.1.5. Sistemas de coordenadas GPS

Teniendo como sistemas de coordenadas GPS:

- ECEF (Earth Centered Earth Fixed):
- Referencia Elipsoide (Reference Ellipsoid):
- ECEF y WGS-84
- Geoid and GPS Height:

### 3.1.6. Fuentes de error en GPS

Una breve descripción de algunas de las fuentes de error mas comunes [5][15][1][25][17] serian:

#### **Error en el reloj del satélite**

Como consecuencia de una sincronización pobre entre el reloj de diferentes satélites puede tener como consecuencia el inducir un error. Previendo, cada satélite tiene cuatro relojes atómicos, y un sistema que vigila que la sincronización en los diferentes satélites sea de un rango de solo 1 nanosegundo.

**Calendario astronomico** Así mismo, la precisión de la posición calculada de los satélites naturales afecta la precisión de los satélites de este sistema.

**Retraso Ionosférico** La ionosfera se extiende a través de varias capas al rededor de 50 a 100 Km. El gas en esta región es ionizado por los rayos ultra violeta del sol, causando la liberación de los electrones, que a su vez afecta a la propagación de las señales electromagnéticas. El error se encuentra en el orden de 1-10 m, siendo posible alcanzar los 50m. Una aproximación de la refracción ionosférica vertical es conocida con la Klobuchar model y es usada al contar el error inducido por retraso. El coeficiente de este modelo es transmitido por el mensaje de navegación del GPS y este es implementado en los receptores GPS.

**Retraso por la Troposfera** Se extiende desde el nivel del mar asta una altura de unos 15 Km. La troposfera induce un error que puede variar de entre 2.4 m a 25 m, dependiendo del ángulo entre los satélites y el horizonte. Temperatura, presión y humedad, afectan el retraso, siendo posible modelar y usar para mitigar el error impuesto.

**Reducción de precisión** En adición a la precisión del pseudorange individual, la geometría de la constelación de satélites dada afecta la precisión de la solución. Esta GDOP (por sus siglas en inglés Geometric Dilution of Precision) no puede ser corregida por modelo.

## 3.2. GNSS diferencial y RTK

Al combinar mediciones de dos receptores separados, error y ruido de observaciones comunes del mismo satélite pueden ser removidas mediante un proceso llamado Single Differencing. Así mismo errores de receptores específicos pueden ser eliminados comparando observaciones de diferentes satélites, igualmente llamado Double Differencing. Como consecuencia, a un pseudo rango de doble diferencia son: alto grado, sistemas libres de error originado de los satélites y receptores.

Posición diferencial con GNSS, o DGNSS, requiere de una referencia o estación base receptora que transmita correcciones a un receptor. Esto significa que solo la navegación relativa de los dos receptores es beneficiada del incremento de precisión. Sin embargo, si la correcta posición de la base es conocida, como consecuencia la posición global del otro puede ser extrapolada con la misma precisión. DGNSS con medición de fase es usada en aplicaciones que requieran de una cinemática con mayor precisión, y este método es usualmente denotado como el Real-Time Kinematics (RTK) técnicamente cuando el rendimiento requerido es en tiempo real. Como se menciona anteriormente este también

involucra el IAR.

### **3.2.1. IAR (Integer Ambiguity Resolution)**

La fase portadora observable del GPS receptor es usada para medir el rango del satélite, y es responsable de mejorar el nivel de precisión. Estas mediciones de rango son usadas al calcular el vector entre la APC de la base y el APC del rover.

#### **3.2.1.1. Medición de fase**

La onda portadora del GPS es una onda polarizada circularmente hacia la derecha. La medición de fase ocurre de la siguiente manera:

1. La primera medición observada es una fase parcial de la portadora del GPS.
2. Después, el receptor mide la longitud de onda entera de la onda portadora.
3. El receptor continuamente cuenta toda la longitud de onda. El receptor no conoce (ambigüedad) el número exacto del ancho de la longitud de onda total (integer) entre el SV y GPS APC.

#### **3.2.1.2. Diferencia de fase portadora**

- Cada receptor mide la misma señal de un satélite.
- Si la señal a un receptor es usada como referencia, una diferencia de fase entre dos señales pueden ser calculados.
- Diferenciación de fase es aplicado a todos los satélites en común.

Diferencia de fase portadora es el método utilizado para resolver el IAR, Figura 3.3.

Cuando una señal es transmitida del satélite, esta viaja al receptor. La señal en uno de los receptores es usada como una señal referencia. Cuando esta referencia es comparada a la señal recibida en el otro receptor, esa es una diferencia de la señal de fase entre los dos receptores. Las diferencias de fase son derivadas con los otros satélites vistos.

#### **3.2.1.3. Solución de Ambigüedad**

El proceso de resolver el integers puede ser dividido en los siguientes pasos:

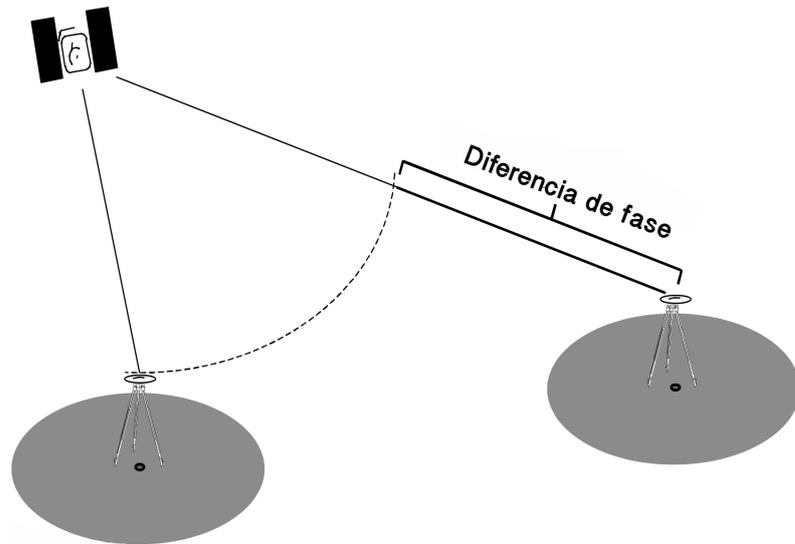


Figura 3.3: Diferencia de fase portadora entre receptor Rover y receptor GCS

1. **Estimación inicial del integer:** Estos son calculados usando pseudo rangos ambiguos (basado en el código C/A). El error esperado de esta estimación define un volumen de búsqueda al rededor de la posición desconocida de la antena rover. Este volumen considera contener una matriz 3-D de puntos, cada una asociada con una combinación de integer.
2. **Búsqueda de la mejor combinación:** La búsqueda del integer compone una serie de cálculos de la posición del rover usando todas las combinaciones simples. La correcta combinación será definida a si misma basada en pruebas estadísticas definiendo la calidad de la solución. Siendo cualquiera de estas pruebas posible si la solución es sobredeterminada, lo cual requiere de al menos cinco satélite visibles en común por los receptores GPS

#### 3.2.1.4. Solución Float y FIX

Después de que el IAR ha sido resuelto, el tipo de solución es conocido como una solución Float. Esto implica que la ambigüedad de los valores tiene una parte decimal (los valores no son enteros). Siguiendo la búsqueda del integer la ambigüedad de integer es resuelta, y el sistema RTK dice ser inicializado. La solución llega a ser una solución Fix (los integer son no tienen parte fraccionaria). Dando como resultado que la señal mejore a nivel de centímetros.

### **3.2.1.5. Calcular solución de baseline**

Una vez el IAR es resuelto, el baseline es calculado estableciendo la posición APC del rover en relación al APC de la base.

El proceso de calcular la solución es:

1. Calcular diferencia de fase
2. Resolver y aplicar el IAR a diferencia de fase.
3. Determinar la precisión a los satélites en común provenientes de la posición errante (rover).
4. Proveer posición de la estación base y la estación de observación.
5. Calcular el baseline y comprobar la inversa entre la posición de la base y el rover.

## **3.3. RTK como parte del Sistema de Navegación**

Se ha mencionado el funcionamiento que hay detrás de un sistema RTK, así como el dispositivo a emplear y sus características. Faltando la parte de como este es incluido al proyecto.

Como ya se menciona el sistema RTK suele ser empleado para la navegación debido a la alta precisión que ofrece al entregar coordenadas de referencia. Razón por la cual se opta por elegirlo como sensor encargado de realimentar la posición del vehículo empleada por el Sistema de Navegación. Colocando a bordo el que se ha denominado como receptor Rover.

Una de las ventajas que ofrece el sistema Pixsi es la versatilidad que tiene al ser compatible con varios lenguajes , sea C, Java, Python o incluso Haskell. Brindando La opción de crear un modulo para su lectura a través de la tarjeta Odroid o incluso desde el mismo Pixhawk. Eligiendo como opción la segunda, puesto que dentro del firmware APMRover existe ya integrado un modulo el cual, junto con la librería sbp de SwiftNAV entregan de manera transparente la información obtenida por el receptor Pixsi, desde posición del vehículo (formato NED), velocidad de desplazamiento, y asta una bandera la cual indicara el momento en que este pase de un estado Float a un Fix.

Teniendo resuelto el problema de la lectura solo es necesario hacer una aclaración al momento de leer la información entregada por el Rover y la entregada por la Base.

Puesto que al trabajar estos con una posición relativa uno con respecto del otro los valores arrojados no resultaran del todo igual por excepción de la magnitud formada por las tres coordenadas. Esto debido a que la percepción de un receptor con respecto del otro resulta inversa, lo que para uno representaría un movimiento hacia el norte, para el otro representaría un movimiento hacia el sur. Esto es representado mediante la Figura 3.4

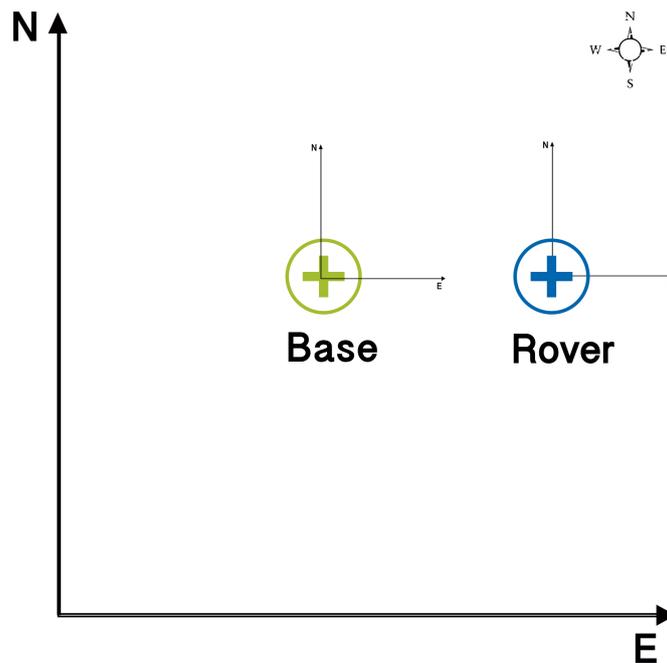
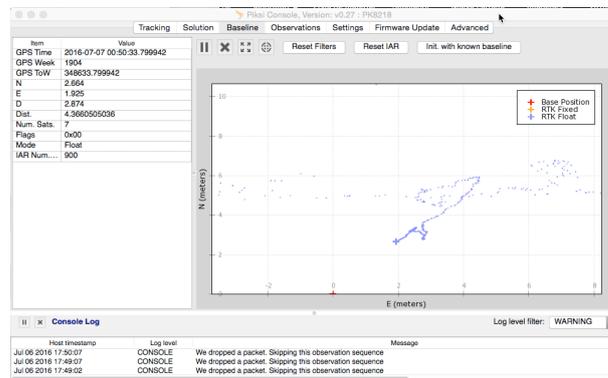


Figura 3.4: Como se muestra en la imagen, al ser relativa la distancia entre estos dos puntos, la posición de uno con respecto del otro dependerá del receptor del que se este tomando la medición. Puesto que aquí se muestra como el Rover se encuentra al Este de la Base y de igual forma la Base se encuentra al Oeste del Rover.

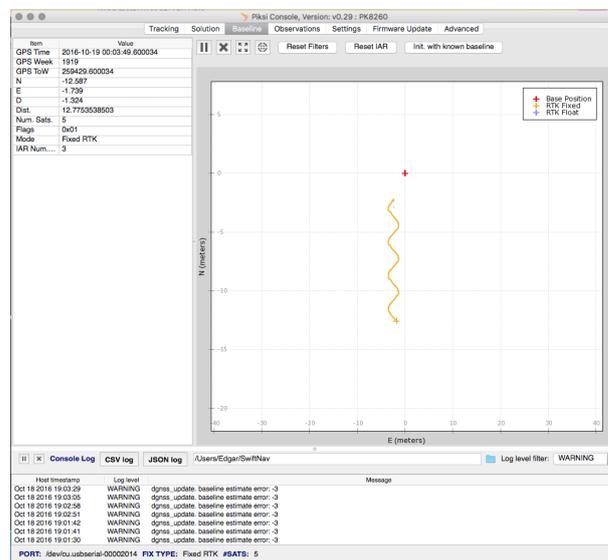
Este resulta un punto muy importante, puesto que, al querer realizar el monitoreo desde la Piksi Console esta dará la idea de que el control esta funcionando de manera negativa.

Otro punto a mencionar al ser utilizado y monitoreado este dispositivo hay que tener presentes las características mencionadas anteriormente para el funcionamiento de un Sistema RTK. Puesto que al monitorear el proceso de solución RTK se deberá observar en dos datos importantes el IAR y el Numero de Satélites Observados. Ya que como se ha mencionado, para llevar a cabo el proceso RTK mediante dos receptores se necesita de al menos una vista de cinco satélites en común. Teniendo en cuenta que si el numero es inferior jamas se llevara a cabo el enlace y el intento de prueba será una perdida d tiempo. Mientras que el IAR dará pauta a conocer el momento en que se este próximo a pasar

de un estado Float a un estado Fix, Siendo este el que indica el numero de operaciones flotantes a resolver antes de concluir el proceso. Ambos valores son desplegados en la Piksi Console como se muestra en la Figura 3.5.



(a)



(b)

Figura 3.5: En ambas imágenes se muestra el monitoreo de los receptores Piksi, tomando como diferencia que la imagen (a) muestra un estado Float mientras que la (b) un estado Fix. Así mismo se muestra como en la imagen (a) previo a pasar a un estado Fix a diferencia de (b) en la sección IAR refleja un numero elevado indicando los flotantes por resolver.

De igual manera dos observaciones que vale la pena mencionar son las principales o mas comunes razones por las que el sistema no trabajar de manera eficiente:

1. Ruido por radiación electromagnética: Principalmente provocado por una mala colocación del receptor cerca de dispositivos que generen un campo electromagnético (motores, compass, telemetría, entre otros).
2. Ruido por radiación térmica: Principalmente provocado por la radiación solar, la cual a su vez también genera ruido electromagnético.

---

# Modelado

---

En este capítulo se presenta el desarrollo del modelo matemático del vehículo, el cual ha sido considerado para diseñar el esquema de control.

Se parte del hecho de que el vehículo consta de una configuración de tipo Skid Steer [28] por lo que el modelado del sistema resulta análogo al de un vehículo con configuración de tipo diferencial [21][7][2][25][19][10][24] .

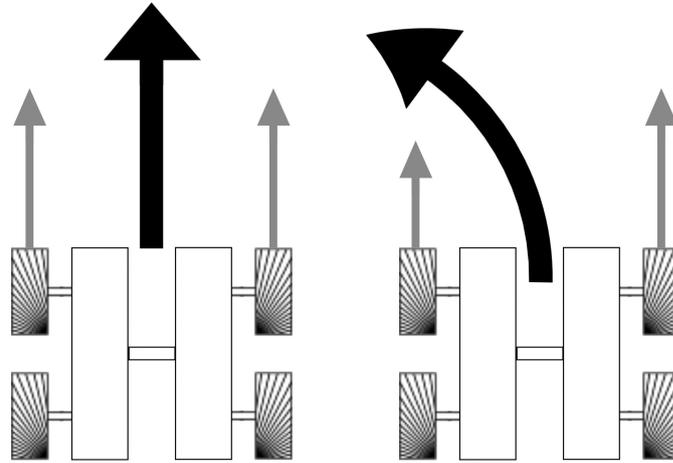
## 4.1. Modelado del vehículo

Siendo una configuración Skid Steer, su comportamiento está dado por el accionamiento simultáneo de las ruedas a cada lado del vehículo. La navegación queda definida por el control de la velocidad generada por el grupo de ruedas izquierdo y derecho. De tal forma que el giro del vehículo a de ser consecuencia por la diferencia de velocidades de ambos grupos de ruedas (Izquierdo y Derecho). Teniendo la ventaja de poder girar sobre su centro como se muestra en la Figura 2.1.

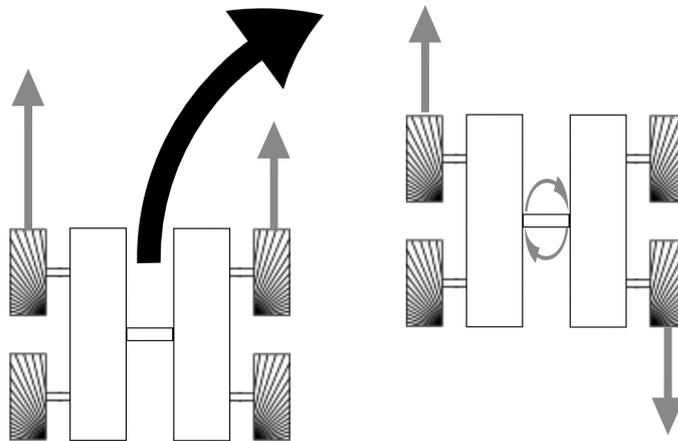
### 4.1.1. Modelo cinemático

La cinemática es la parte de la mecánica que trata del movimiento en sus condiciones de espacio y tiempo, sin tener en cuenta las causas que lo producen. Su análisis consta de las variables de posición junto con sus derivadas (velocidad, aceleración, etc.). Los elementos básicos de la cinemática son el espacio, el tiempo y un móvil [2][24]. Dentro de la cinemática existen dos enfoques llamados cinemática directa y cinemática inversa.

El problema de la cinemática directa consiste en determinar cual es la posición y orientación de un cuerpo, con respecto a un sistema de coordenadas fijo, que sin pérdida

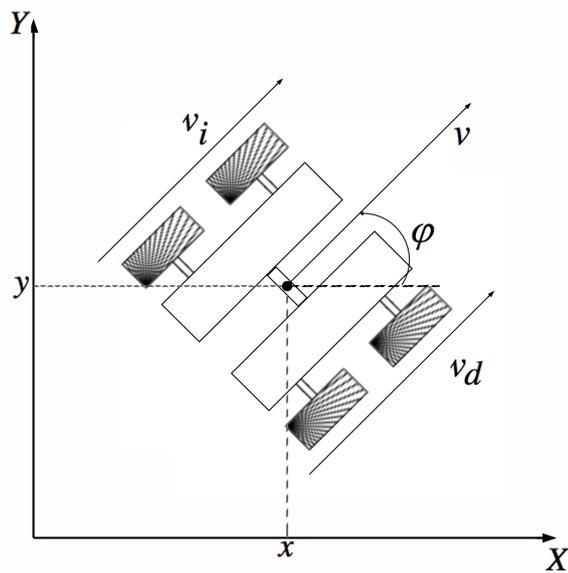


(a) Manteniendo igual velocidad en (b) Disminuyendo velocidad en el conjunto de ruedas izquierdo se provoca un desplazamiento en línea recta sea adelante o atrás dependiendo radio de giro en contra de las manecillas del reloj.

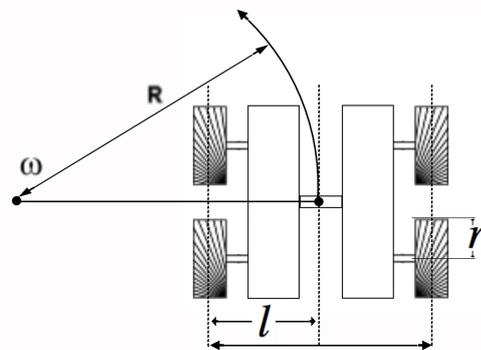


(c) Disminuyendo velocidad en el conjunto de ruedas derecho se provoca un desplazamiento con un radio de giro en sentido a las manecillas del reloj. (d) Manteniendo la misma velocidad en ambos extremos pero con sentido distinto se consigue una rotación sobre el propio eje.

Figura 4.1: Esquema de movimientos según efecto de las velocidades en las ruedas.



(a) 1



(b) 2

Figura 4.2: Diagramas de fuerzas que intervienen en el vehículo.

de generalidad es el origen del plano XY, el cual se toma como referencia [24]. En la robótica móvil a diferencia de los brazos manipuladores, con la cinemática directa es posible verse como una transformación que mapea pares de trayectorias de velocidad  $[v(t), w(t)]$  del vehículo, es decir:

$$\Psi : [v(t), w(t)] \rightarrow X(t) = [x(t), y(t), \varphi(t)] \quad (4.1)$$

A continuación se ve el análisis del vehículo sobre el plano XY, como muestra la Figura 2.2. Donde posición y orientación de la plataforma pueden ser descritos por el par coordenado  $(x, y)$  que conforman el punto medio dado por el eje que une ambos conjuntos de ruedas, con respecto al sistema de referencia fijo XY y el ángulo  $\varphi$  que forma el eje de simetría del vehículo con respecto al eje X positivo. De dicha figura son definidas las siguientes variables:

- $v(t)$  Velocidad traslacional del punto medio del vehículo.
- $v_d(t)$  Velocidad traslacional con respecto al conjunto de ruedas derecho.
- $v_i(t)$  Velocidad traslacional con respecto al conjunto de ruedas izquierdo.
- $\varphi$  Posición angular con respecto al punto medio del vehículo.
- $w(t)$  Velocidad angular del punto medio del vehículo.
- $R$  Radio de giro.
- $r$  Radio de las ruedas.
- $l$  Distancia entre el eje medio del vehículo y las ruedas.

Donde las velocidades sobre X e Y es decir  $\dot{X}$  y  $\dot{Y}$  conforman la velocidad traslacional  $v$ . Mientras que  $w$  que denota a la velocidad angular es definida como la razón de cambio de  $\varphi$ , es decir  $\dot{\varphi}$ . Partiendo del esquema mostrado en la Figura 2.2, se tiene que  $\dot{x}$ ,  $\dot{y}$ ,  $\dot{\varphi}$  pueden ser escritas de la siguiente forma:

$$\begin{aligned} \dot{x} &= v \sin(\varphi) \\ \dot{y} &= v \cos(\varphi) \\ \dot{\varphi} &= w \end{aligned} \quad (4.2)$$

Considerando el modelo cinemática dado por (2.2), tenemos que las variables de interés resultan ser  $v$  y  $w$ . Sabiendo que, tanto velocidad traslacional ( $v$ ) como velocidad angular ( $w$ ) del vehículo son dadas por la diferencia de velocidades angulares correspondientes al lado izquierdo y derecho dadas por los conjuntos de ruedas  $w_i$  y  $w_d$ .

De la ecuación (2.2) se obtiene la relación entre  $v$  y  $w$  con respecto a  $w_i$  y  $w_d$ . Utilizando las coordenadas generalizadas  $(x, y, \varphi)$ , se encuentra que la posición de los puntos de contacto de cada una de las llantas respecto al plano de referencia XY está dada por:

$$\begin{aligned}\vec{r}_i &= (x - l \operatorname{sen}(\varphi))\hat{x} + (y + l \operatorname{cos}(\varphi))\hat{y} \\ \vec{r}_d &= (x - l \operatorname{sen}(\varphi))\hat{x} + (y + l \operatorname{cos}(\varphi))\hat{y}\end{aligned}\quad (4.3)$$

Donde  $\hat{x}$  y  $\hat{y}$  son vectores unitarios que indican las componentes asociadas a los ejes de referencia XY.  $\vec{r}_i$  y  $\vec{r}_d$  indican la posición de en el plano XY del punto de contacto de cada conjunto de ruedas izquierdo y derecho respectivamente. Además, se tiene que si  $m$  es la masa de cada una de las llantas, entonces se tiene que la energía cinética del sistema esta dada como

$$T = \frac{m}{2} (|\dot{r}_i|^2 + |\dot{r}_d|^2) = m(\dot{x}^2 + \dot{y}^2 + l^2 \dot{\varphi}^2) \quad (4.4)$$

De acuerdo con la teoría de LaGrange de la mecánica clásica [2], las ecuaciones de movimiento del sistema físico están dadas por

$$\begin{aligned}\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{x}} \right) - \left( \frac{\partial T}{\partial x} \right) &= Q_x \\ \frac{d}{dt} \left( \frac{\partial T}{\partial \dot{y}} \right) - \left( \frac{\partial T}{\partial y} \right) &= Q_y \\ \frac{d}{dt} \left( \frac{\partial T}{\partial \dot{\varphi}} \right) - \left( \frac{\partial T}{\partial \varphi} \right) &= Q_\varphi\end{aligned}\quad (4.5)$$

donde  $Q_x, Q_y, Q_\varphi$ , son las denominadas componentes de la fuerza generalizada, las cuales están denotadas por:

$$\begin{aligned}Q_x &= \vec{F}_i \cdot \frac{\partial \vec{r}_i}{\partial x} + \vec{F}_d \cdot \frac{\partial \vec{r}_d}{\partial x} = (\vec{F}_i + \vec{F}_d) \cdot \hat{x} \\ Q_y &= \vec{F}_i \cdot \frac{\partial \vec{r}_i}{\partial y} + \vec{F}_d \cdot \frac{\partial \vec{r}_d}{\partial y} = (\vec{F}_i + \vec{F}_d) \cdot \hat{y} \\ Q_\varphi &= \vec{F}_i \cdot \frac{\partial \vec{r}_i}{\partial \varphi} + \vec{F}_d \cdot \frac{\partial \vec{r}_d}{\partial \varphi} = l (\vec{F}_i - \vec{F}_d) \cdot (\operatorname{cos}\varphi \hat{x} + \operatorname{sen}\varphi \hat{y})\end{aligned}\quad (4.6)$$

Usando las ecuaciones (2.3) y (2.7), se tiene que las ecuaciones de movimiento están

dadas por:

$$\begin{aligned}
 2m\ddot{x} &= (\vec{F}_i + \vec{F}_d) \cdot \hat{x} \\
 2m\ddot{y} &= (\vec{F}_i + \vec{F}_d) \cdot \hat{y} \\
 2m\ddot{\varphi} &= l(\vec{F}_i - \vec{F}_d) \cdot (\cos\varphi\hat{x} + \sin\varphi\hat{y})
 \end{aligned} \tag{4.7}$$

Puesto que  $\vec{F}_i, \vec{F}_d$  son las fuerzas que actúan sobre el par de llantas del lado izquierdo y derecho respectivamente, dando las velocidades  $\vec{v}_i, \vec{v}_d$  se tiene que:

$$\begin{aligned}
 \vec{F}_i &= m\vec{v}_i = mv_i(\cos\varphi\hat{x} + \sin\varphi\hat{y}) \\
 \vec{F}_d &= m\vec{v}_d = mv_d(\cos\varphi\hat{x} + \sin\varphi\hat{y})
 \end{aligned} \tag{4.8}$$

Donde  $v_i, v_d$  son las magnitudes de aceleraciones correspondientes al par de llantas izquierdo y derecho respectivamente. Con el fin de simplificar se aplica una integración con respecto al tiempo a (2.4) y(2.5)

$$\begin{aligned}
 2\dot{x} &= (v_i + v_d)\cos(\varphi) + c_x \\
 2\dot{y} &= (v_i + v_d)\sin(\varphi) + c_y \\
 2l\dot{\varphi} &= (v_d - v_i)c_\varphi
 \end{aligned} \tag{4.9}$$

Teniendo que si :  $w_i$  y  $w_d$  son las velocidades del conjunto de llantas izquierdo y derecho respectivamente, la condición de rodadura sin deslizamiento es equivalente a:

$$\begin{aligned}
 v_i &= rw_i \\
 v_d &= rw_d
 \end{aligned} \tag{4.10}$$

Considerando las constantes de integración  $c_x, c_y,$  y  $c_\varphi$  igual con cero, se tiene que las ecuaciones del modelo cinemática están dadas por:

$$\begin{aligned}
 \dot{x} &= \frac{(w_d + w_i)r}{2}\cos\varphi \\
 \dot{y} &= \frac{(w_d + w_i)r}{2}\sin\varphi \\
 \dot{\varphi} &= \frac{(w_d - w_i)r}{2l}
 \end{aligned} \tag{4.11}$$

Recordando de que la magnitud  $v$  se define por:

$$v = \sqrt{\dot{x}^2 + \dot{y}^2} = \frac{(w_d + w_i)r}{2} \tag{4.12}$$

Mientras que  $w$  representa la magnitud de velocidad angular con que el vehículo rota con respecto al plano XY. Teniendo así la relación:

$$w = \frac{(w_d - w_i)r}{2l} \quad (4.13)$$

Reescribiendo (2.11) y (2.12) en su forma matricial se tiene:

$$\begin{bmatrix} v \\ w \end{bmatrix} = T \begin{bmatrix} W_d \\ W_i \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{2l} & -\frac{r}{2l} \end{bmatrix} \begin{bmatrix} W_d \\ W_i \end{bmatrix} \quad (4.14)$$

Dado que T es una matriz no singular para todo  $r > 0$ ,  $l > 0$ , implica que mediante la correcta selección de  $w_i$  y  $w_d$  puede ser obtenido el perfil de velocidad deseado para  $v$  como para  $w$ .

Por otro lado recordando que el problema de la cinemática inversa es el de encontrar el perfil de velocidad [24]  $[v_d(t), w_d(t)]$ , tal que sea obtenida posición y orientación deseada para el vehículo, es decir:

$$\Psi[v_d(t)w_d(t)] = X_d(t) \quad (4.15)$$

$$\begin{bmatrix} \dot{X}_d(t) \\ \dot{Y}_d(t) \\ \dot{\varphi}_d(t) \end{bmatrix} = \begin{bmatrix} \cos(\varphi_d)(t) & 0 \\ \text{sen}(\varphi_d)(t) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_d(t) \\ w_d(t) \end{bmatrix} \quad (4.16)$$

Debido a la forma en que se instrumento el vehículo, la cual fue presentada en el Capítulo 2, es que, para el diseño de el esquema de control del vehículo, a sido empleado el modelo cinemática dado por (2.2) y (2.16)



---

## Sistema de navegación

---

Para la implementación del sistema de navegación al vehículo, se ha optado por una arquitectura de navegación deliberativa la cual será basada bajo el paradigma de "senso-rización-planificación-actuación"[23]:

1. Se considera el tener conocimiento completo del entorno, lo cual incluye el punto de inicio, entorno por el cual se desea pasar y un punto meta. En primer instancia se considera a los obstáculos como puntos conocidos.
2. Se realiza un plan de ejecución para llevar a cabo la tarea, es decir, se asigna la trayectoria a seguir evitando los obstáculos.
3. Se procede con el seguimiento de la trayectoria aplicando un algoritmo de retroalimentación el cual servirá para seguir la trayectoria de la manera mas exacta acorde al objetivo deseado.

Siendo considerada esta arquitectura como la mas apropiada, al ser que a priori se conoce el entorno en el que se desplazara el vehículo (área de cultivo o de prueba en este caso), las zonas a evitar y las que hay que recorrer, lo cual permite planificar la trayectoria, a modo de obtener el mejor resultado posible. En la Figura 4.1 se muestra un esquema de la arquitectura que seguirá el algoritmo de navegación.

En esta arquitectura, aparece un primer bloque encargado de analizar el entorno y obtener un mapa. A partir del cual se encargara de generar una trayectoria para llevar a cabo la tarea. Una vez obtenida la trayectoria de referencia, a partir de este punto el bloque de control se encarga de determinar las señales de control a seguir por los actuadores del vehículo (motores). Por ultimo, al determinar posición, velocidad y orientación del

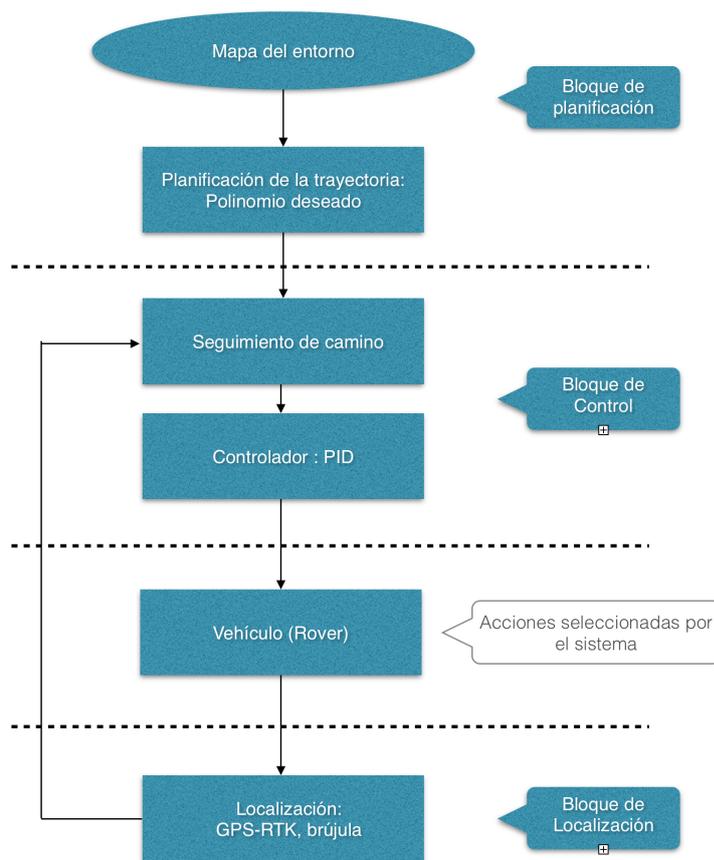


Figura 5.1: Diagrama que muestra el algoritmo designado a seguir para la navegación del vehículo.

vehículo se utilizara los sensores ya designados conformados por el sistema GPS-RTK [2][35][23][37] y una brújula digital la cual se encuentra embebida dentro del dispositivo Pixhawk.

## 5.1. Cálculo de trayectoria de referencia

### 5.1.1. Selección de la trayectoria

Siendo la agricultura de precisión el objetivo de este sistema y tomado como asignada la tarea de cultivo, es que al seleccionar el patrón para diseñar la trayectoria es el de siembra por surcos (Figura 4.2). Resultando este como el mas común al cultivar distintas hortalizas o incluso al realizar inspección de diversos plantíos [34]:

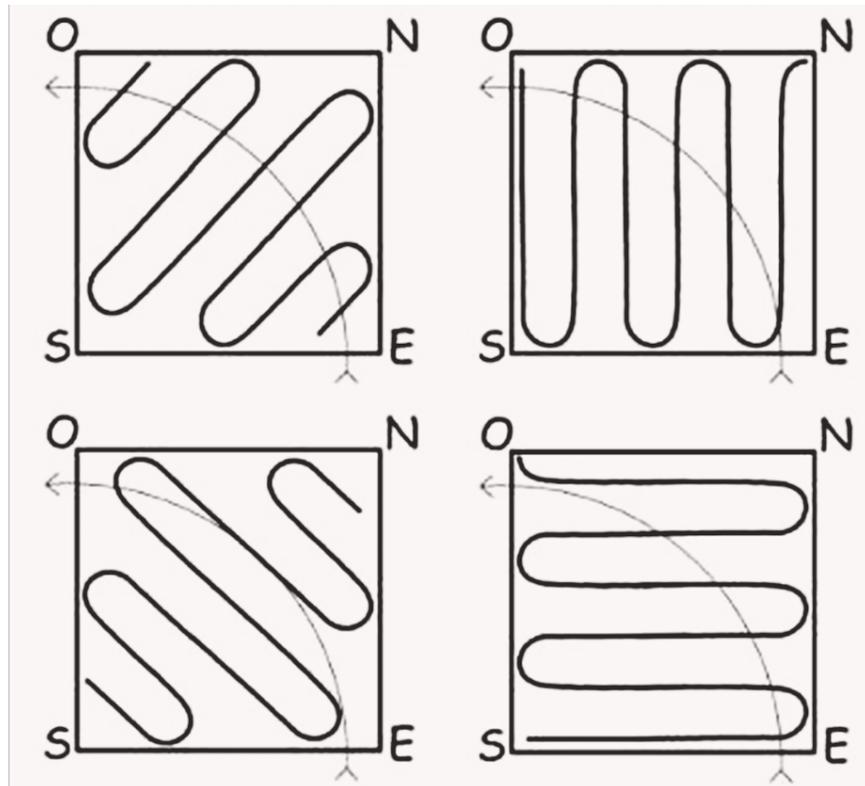


Figura 5.2: Trayectoria de siembra por surcos. La orientación del surco dependerá de la caída del terreno y el arco de insolación del sol.

### 5.1.2. Generación de la trayectoria

Habiendo sido determinada la trayectoria con la cual se llevara a cabo la tarea en el área de pruebas, se ha de interpolar los puntos que conforman el patrón [17], para generar

la trayectoria de referencia que se utilizara por el sistema de control de navegación del vehículo.

Para dicho objetivo se encuentran 2 posibles métodos:

1. Recorrer con el GPS la trayectoria e ir guardando los puntos que serán los que sirvan de referencia a la hora de realizar la trayectoria.
2. Utilizar métodos geométricos para calcular la trayectoria. Siendo tomados como referencia el punto de inicio y el primer punto de giro, para así generar matemáticamente una trayectoria GPS exacta [9][13][15][8][23][17].

Haciendo la consideración de que el primer método involucra la tarea de realizar dos recorridos, un primero para memorizar la ruta deseada y un posterior en el que se realiza la tarea. Es que se opta por elegir el segundo método donde conociendo el punto de partida y el punto final se asignara un polinomio a seguir omitiendo así el primer trayecto. En la Figura 4.3 se muestran los parámetros a considerar para la creación del polinomio de acuerdo con el trayecto mostrado en la Figura 4.2.

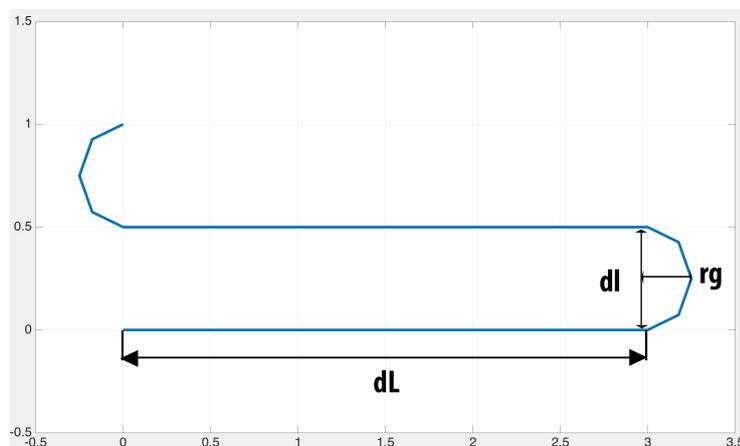


Figura 5.3: Parámetros a considerar para el cálculo del polinomio a aplicar a la tarea de siembra.

donde:

- $dL$  = Desplazamiento longitudinal.
- $dl$  = Desplazamiento lateral.
- $rg$  = Radio de giro.

Utilizando de manera iterativa el polinomio definido a partes mostrando en (4.1) y (4.2) se obtiene el esquema de asignación de puntos mostrado en la Figura 4.4, teniendo

como resultado una trayectoria como la que se observa en la Figura 4.5 así como un patrón obtenido mediante el algoritmo es presentado en la Figura 4.5.

$$x(t, i) = \begin{cases} \text{si } t \leq t1 & x = (t - T * i) * a \\ \text{si } t \leq t2 & x = dL + rg * \cos(tg * ((t - T * i) - t1) + ph) \\ \text{si } t \leq t3 & x = dL - ((t - T * i) - t2) * a \\ \text{si } t \leq t4 & x = rg * \cos(-tg * ((t - T * i) - t3) + ph) \end{cases} \quad (5.1)$$

$$y(t, i) = \begin{cases} \text{si } t \leq t1 & y = i \\ \text{si } t \leq t2 & y = i + rg + rg * \sin(tg * ((t - T * i) - t1) + ph) \\ \text{si } t \leq t3 & y = i + 2 * rg \\ \text{si } t \leq t4 & y = i + rg * 3 + rg * \sin(-tg * ((t - T * i) - t3) + ph) \end{cases} \quad (5.2)$$

donde:

$x_{des}$  Posición longitudinal deseada

$y_{des}$  Posición lateral deseada

$dL$  Desplazamiento Longitudinal.

$tL$  Tiempo en recorrer desplazamiento longitudinal.

$a$  Crecimiento de magnitud Longitudinal.

$tl$  Tiempo en recorrer desplazamiento lateral.

$t1$  Tiempo en recorrer segmento 1.

$t2$  Tiempo en recorrer segmento 2.

$t3$  Tiempo en recorrer segmento 3.

$T$  Tiempo total por periodo.

$rg$  Radio de giro.

$tg$  Frecuencia de giro.

$n$  Numero de patrones deseados.

$\alpha$  Angulo de rotación.

$Mr$  Matriz de rotación.

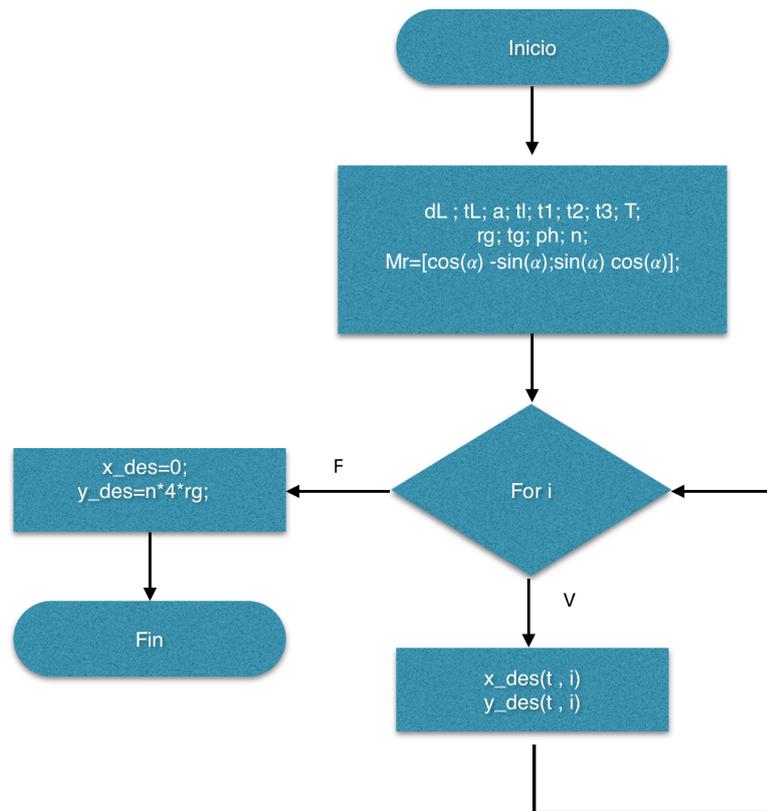


Figura 5.4: Estructura de algoritmo para crear trayectoria representada mediante diagrama de flujo.

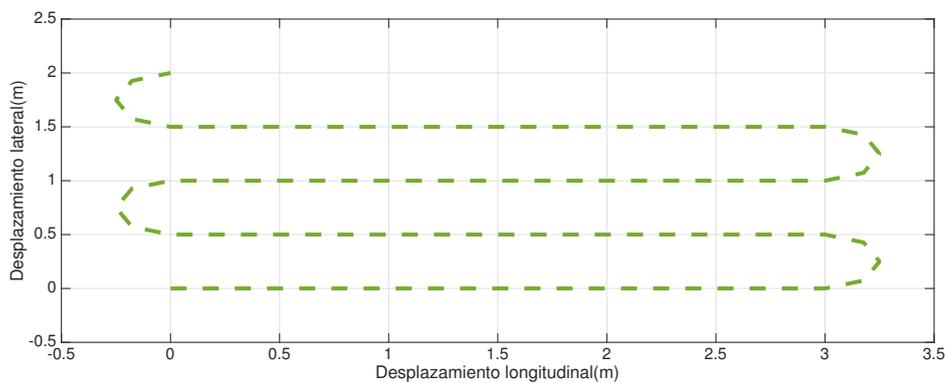


Figura 5.5: Trayectoria de prueba generada mediante polinomio donde  $dL=3m$ ,  $dl=0.5m$  y  $rg=0.25m$ .

## 5.2. Sistema de control

Siendo el sistema de control de un vehículo dividido en dos partes generalmente las cuales suelen estar relacionadas entre si. Por un lado se encuentra el controlador de posición y por otro lado el controlador de posición.

El algoritmo de seguimiento de puntos debe ser capaz de controlar tanto el sistema de desplazamiento (throttle) como el de orientación (steering) tal que el vehículo pueda reducir la distancia entre el punto de partida y el punto final (puntos a y b respectivamente). Siendo definidos movimiento traslacional y rotacional por GPS-RTK (NED relativo) e IMU (N como referencia) respectivamente. Al ser el vehículo pilotado de manera autónoma el sistema de control considerara al GPS-RTK y a la IMU para calcular los errores de seguimiento en heading ( $\varphi$ ) y posición ( $N_R, E_R$ ).

Primero se plantea el definir el error tanto en distancia como en orientación [17][23]:

### 5.2.1. Definición de error

Siendo este compuesto por la diferencia en posición y orientación como se muestra en la Figura 4.6, donde  $Wp_A$  y  $Wp_B$  denotan los puntos de partida y destino respectivamente.

Partiendo por el error en heading ( $\theta$ ). Este es definido como una diferencia entre el ángulo compuesto ( $\beta$ ) y el heading o ángulo de yaw ( $\varphi$ ) con respecto al Norte, como se describe en (4.3) (Figura 4.6). El heading o yaw es directamente medido de la IMU. Por lo que el ángulo compuesto es calculado de la diferencia entre el punto actual de el vehículo (Rover) y el punto al que se desea llegar como es expresado en (4.4).

$$\theta = \beta - \varphi \quad (5.3)$$

$$\beta = \tan^{-1}[(E_B - E_R)/(N_B - N_R)] \quad (5.4)$$

donde:

$(N_R, E_R) = (N,E)$  de la posición del vehículo.

$(N_B, E_B) = (N,E)$  de la posición referente al punto de destino.

Siguiendo a calcular el error de distancia, definiendo a este como la diferencia entre el punto de destino y la posición de relativa del vehículo (4.5) y (4.6).

$$e_N = N_B - N_R \quad (5.5)$$

$$e_E = E_B - E_R \quad (5.6)$$

donde:

$e_N =$  Denota la diferencia entre el vehículo y el punto de destino con respecto al Norte.

$e_E =$  Denota la diferencia entre el vehículo y el punto de destino con respecto al Este.

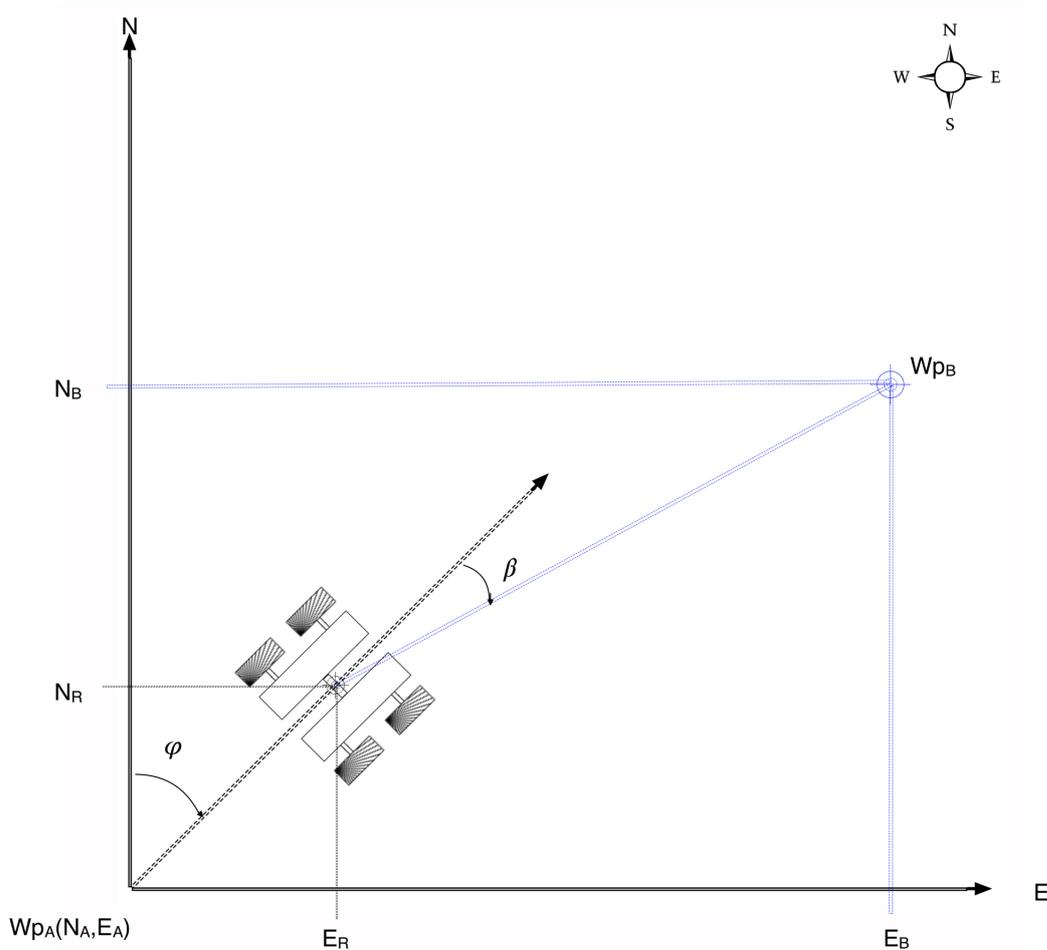


Figura 5.6: Diagrama de referencias. Se muestra al vehículo en relación con el punto de partida y el punto final ( $Wp_A$ ,  $Wp_B$ ) respectivamente, denotando así las diferencias de posición y orientación.

## 5.2.2. Control Proporcional Integral Derivativo (PID)

En aplicaciones de robótica es muy común emplear el esquema de control PID [2][17][27] (Proporcional, Integral y Derivativo), al requerir controlar posición y/o velocidad. El empleo de un controlador PID tiene por ventaja su robustez, así como el hecho de que al emplear este tipo de controladores, es que se pueden ajustar los valores de retroalimentación para poder obtener la respuesta deseada.

El control PID genera una respuesta de control en base al error, tomando como entrada el valor deseado y el valor actual.

Un controlador de tipo PID esta conformado por tres consignas de control (proporcional, integral y derivativa como se muestra en la Figura 4.7), donde cada una de estas es encargada de realizar una función específica:

- Proporcional: Funciona como un amplificador de ganancia ajustable, varía instantáneamente con el error y alcanza un valor estacionario cuando el error lo alcanza.
- Integral: Suele ser denominada como acción de control de reajuste, variando la señal de salida del control en relación a la señal de error. Esta acción de control incrementa gradualmente su fuerza a medida que el error es diferente de cero, ayudando a reducir el error estacionario.
- Derivativa: Suele ser denominada como control de velocidad, donde la magnitud de la salida del controlador es proporcional a la velocidad de cambio de la señal de error. La acción de control derivativa tiene un carácter de previsión, dado que es proporcional a la tendencia de cambio de la señal de error.

Siendo la ley de control PID definida por (4.7):

$$\tau = K_p \tilde{q} + K_v \dot{\tilde{q}} + K_i \int_0^t \tilde{q}(\sigma) dt \quad (5.7)$$

El lazo de control trabaja con el modelo cinemático y las trayectorias deseadas para el vehículo produciendo como salidas de control los perfiles de velocidad deseados  $[v \ w]$  [23][17].

Los controladores PID que se encargaran de controlar steering y throttle respectivamente serán los encargados de minimizar el error de heading ( $\theta$ ) y el error de distancia ( $e_N$ ,  $e_E$ ) tal que ambos errores vallan decrecentando a cero de manera gradual [2][17].

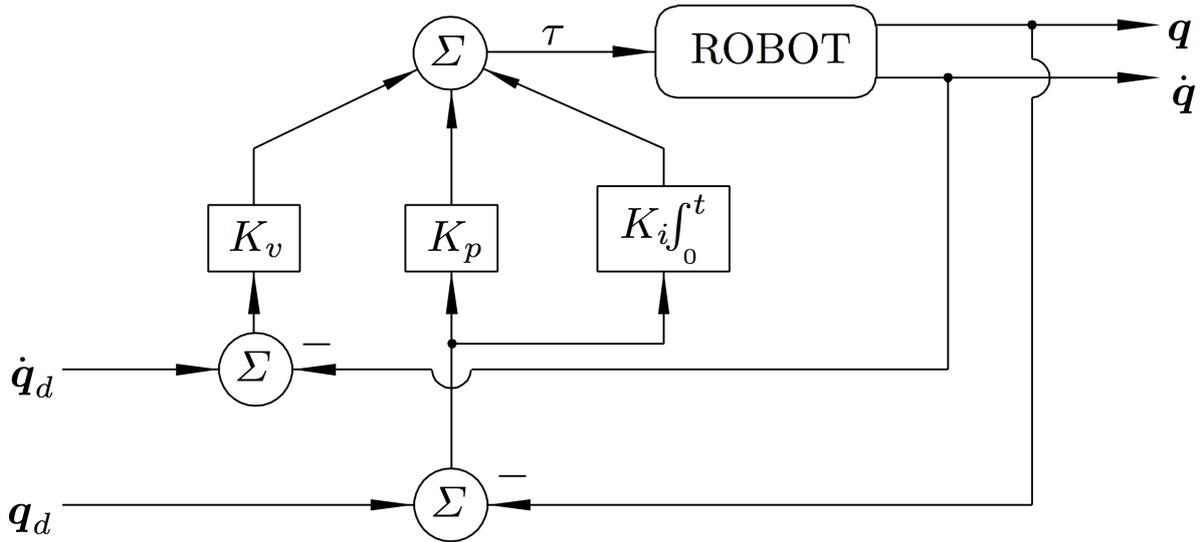


Figura 5.7: Diagrama a bloques de un robot en el cual interviene un controlador de tipo Proporcional Integral Derivativo.

Siendo así las ecuaciones que gobiernen el control del vehículo (4.8) (4.9) (4.10) y (4.11) como resultado de sustituir (4.5)(4.6) y (4.3) en (4.7) respectivamente:

$$u_N = K_p e_N + K_d \dot{e}_N + K_i \int_0^t e_N(\sigma) dt \quad (5.8)$$

$$u_E = K_p e_E + K_d \dot{e}_E + K_i \int_0^t e_E(\sigma) dt \quad (5.9)$$

$$U_2 = K_p \theta + K_d \dot{\theta} + K_i \int_0^t \theta dt \quad (5.10)$$

Siendo este un sistema subactuado la señal de control para la posición será dada por (4.11)

$$U_1 = f(u_N, u_E, \beta) \quad (5.11)$$

donde  $\beta$  determina la prioridad ejecución entre (4.8) y (4.9).

### 5.3. Localización

Al ser usado el sistema RTK como punto de referencia absoluta para el posicionamiento del vehículo y la central inercial (IMU) es que hay que hacer las siguientes consideraciones:





---

# Simulación

---

En los capítulos anteriores (2 y 4) se es presentado tanto el modelado del vehículo visto dese el enfoque tanto cinemático como dinámico optando por tomar el primero para el desarrollo del proyecto basando la selección en cuestiones practicas por su simplicidad al momento de emplear, así como el diseño del algoritmo propuesto para la navegación respectivamente. Siendo, la intención de este capitulo presentar a nivel de simulación el correcto funcionamiento del sistema asta ahora ya planteado.

## 6.0.1. Resultados experimentales

Con el fin de validar el método de navegación propuesto para el vehículo, se procede a realizar simulaciones en las que se es sometido el modelo cinemático del vehículo obtenido en el Capitulo 2, el cual es ahora gobernado por la ley de control PID, a realizar distintas trayectorias. Teniendo estas una evolución de tipo: recta (Figura 6.1, Figura 6.2), pendiente (Figura 6.3, Figura 6.4), senoidal (Figura 6.5, Figura 6.6) y hélice (Figura 6.7, Figura 6.8). Mostrando los resultados obtenidos mediante gráficas en las cuales se observa el comportamiento por parte de  $[X \ Y \ \varphi]$  al seguir la trayectoria deseada, así como el que sería el desplazamiento resultante del vehículo. Buscando obtener un resultado que se asemeje mas al esperado por parte del vehículo, es que se añade una señal de ruido acotado al comportamiento de orientación, con la cual se trata de emular (no del todo) las perturbaciones provocadas por la superficie de desplazamiento.

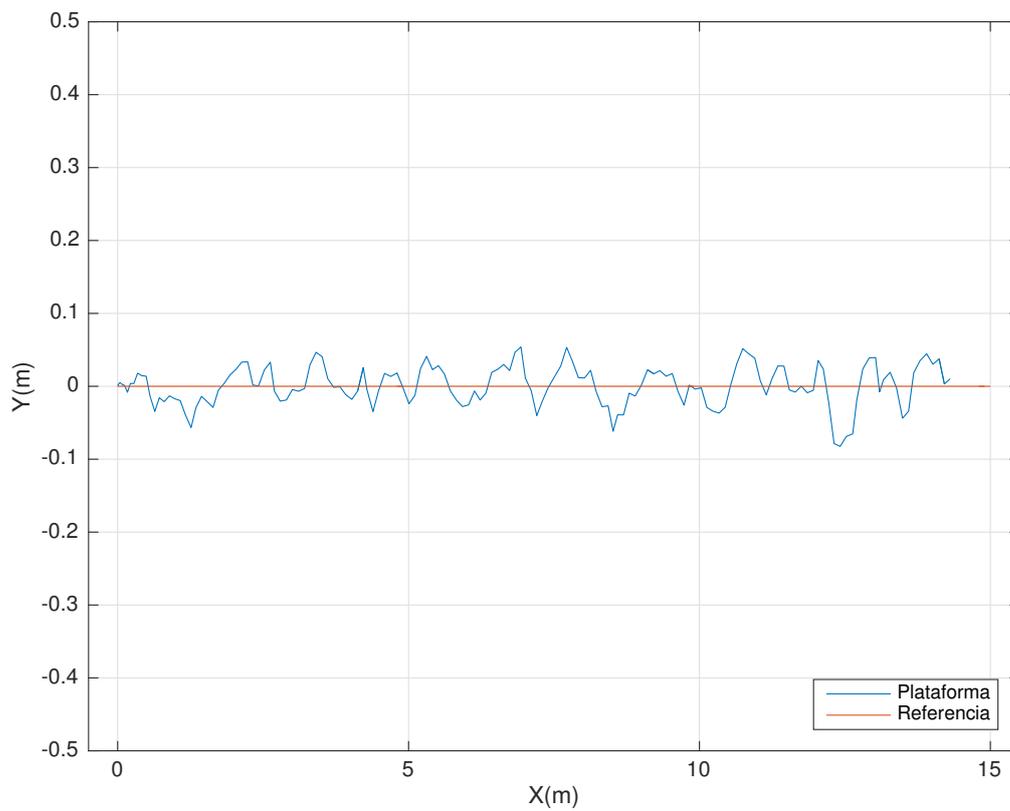


Figura 6.1: Trayectoria de prueba 1: Recta, se parte de condiciones iniciales iguales con cero. Demandando un desplazamiento constante por parte de X mientras la variable Y se mantiene fija con cero.

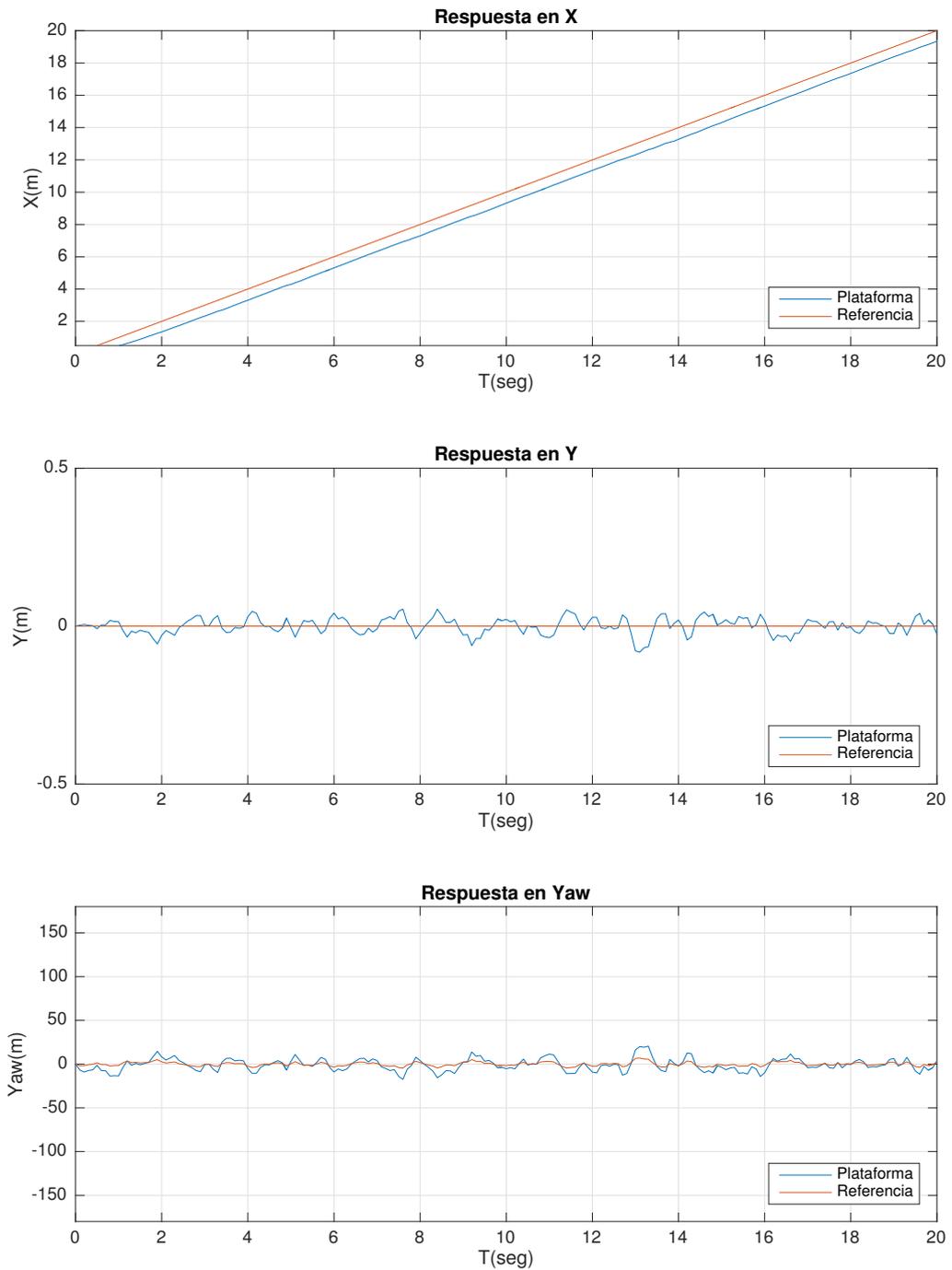


Figura 6.2: Respuesta de los estados  $[X Y \varphi]$  vista de manera separada al realizar la trayectoria de prueba 1. Donde se muestra como la línea que representa al vehículo (Plataforma) trata de seguir la señal deseada con un pequeño desfase por parte de la respuesta en  $X$ . Mientras que para  $Y$  y  $\varphi$  aunque no se muestra del todo uniforme, mantiene una oscilación acotada al rededor de cero.

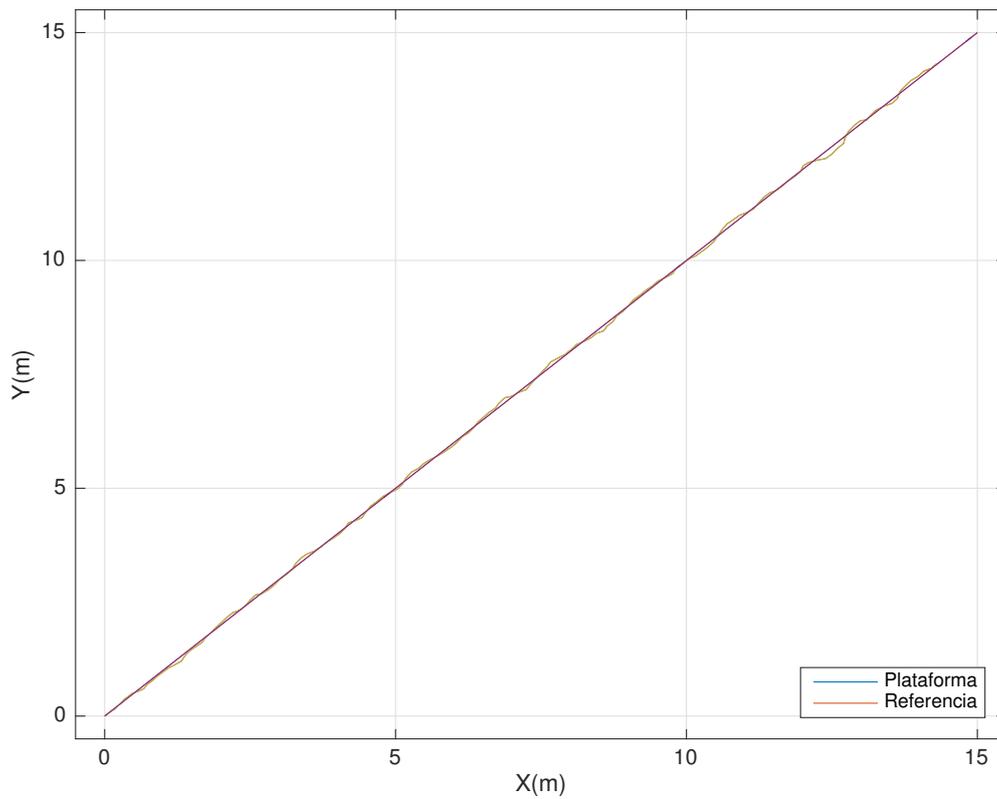


Figura 6.3: Trayectoria de prueba 2: Rampa, se parte de condiciones iniciales iguales con cero. Demandando un desplazamiento constante por parte de [X,Y].

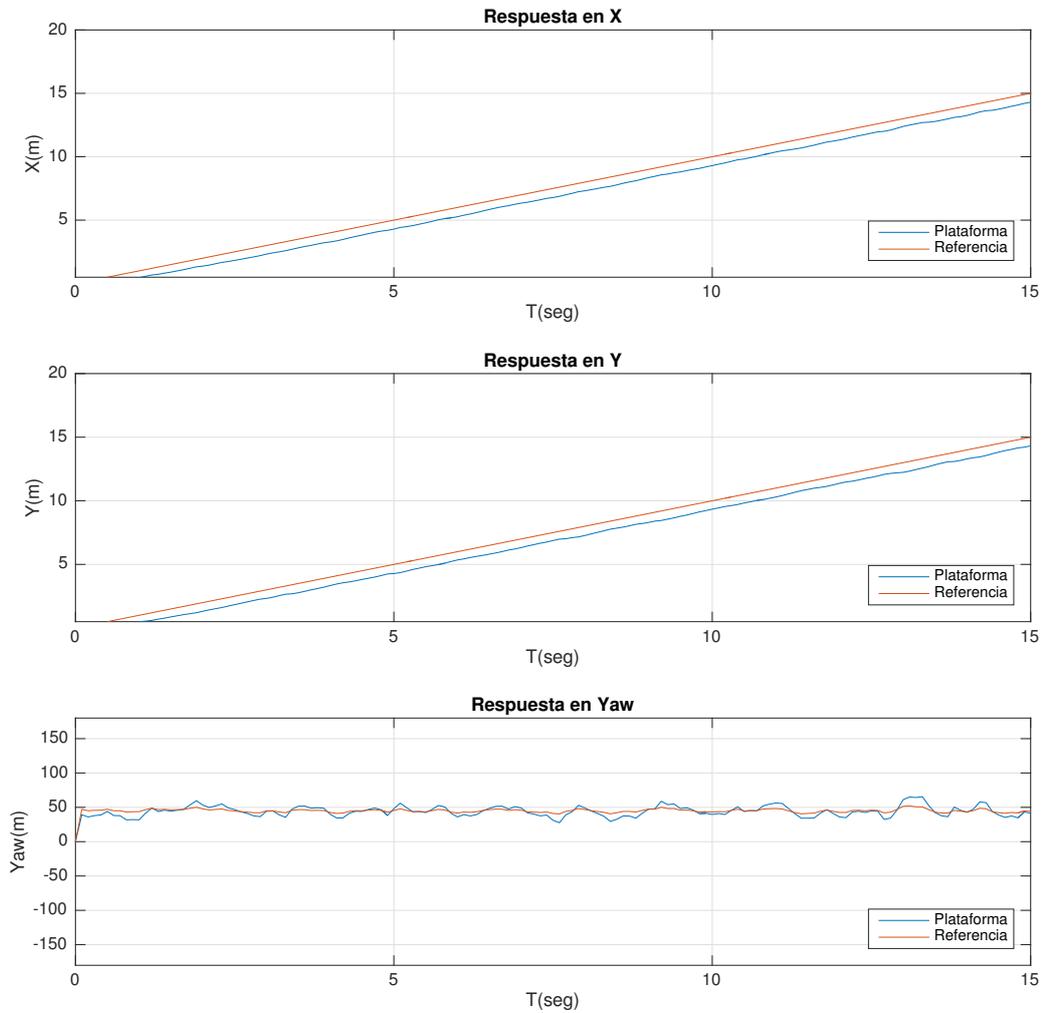


Figura 6.4: Respuesta de los estados  $[X \ Y \ \varphi]$  vista de manera separada al realizar la trayectoria de prueba 2. Donde se muestra como la línea que representa al vehículo (Plataforma) trata de seguir la señal deseada con un pequeño desfase por parte de la respuesta en  $[X, Y]$ . Mientras que para  $\varphi$  aunque no se muestra del todo uniforme, mantiene una oscilación acotada al rededor de 45.

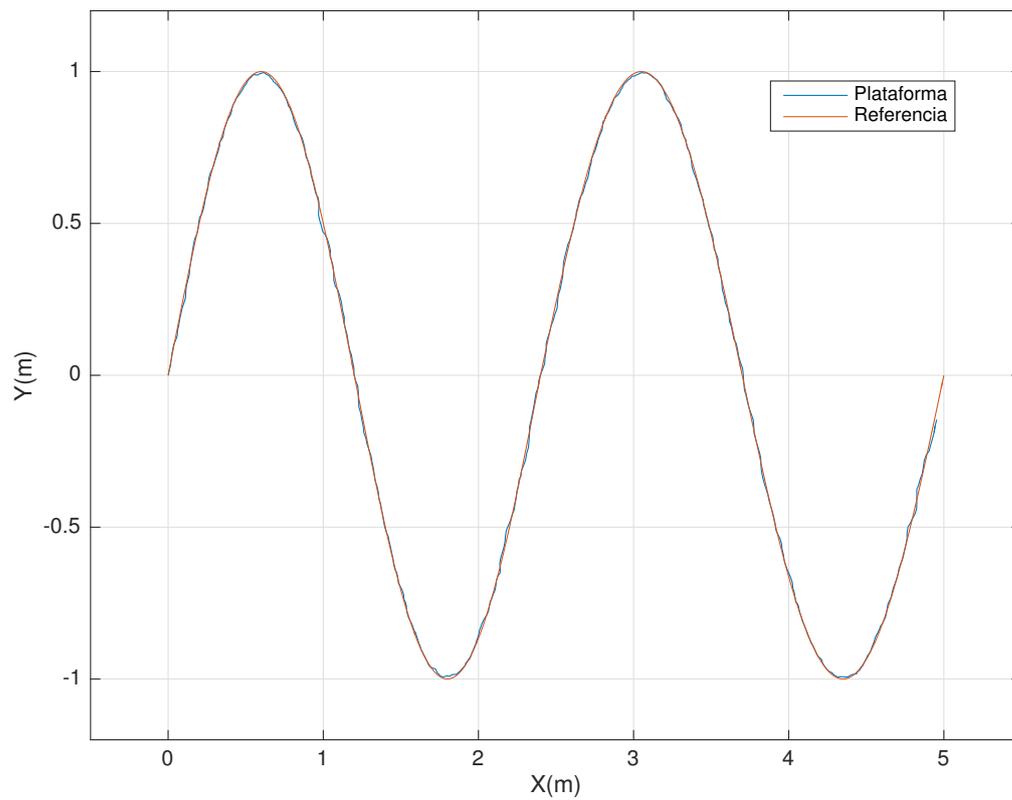


Figura 6.5: Trayectoria de prueba 3: Seno, se parte de condiciones iniciales iguales con cero. Demandando un desplazamiento constante por parte de X, mientras que para Y se pide seguir una función seno.

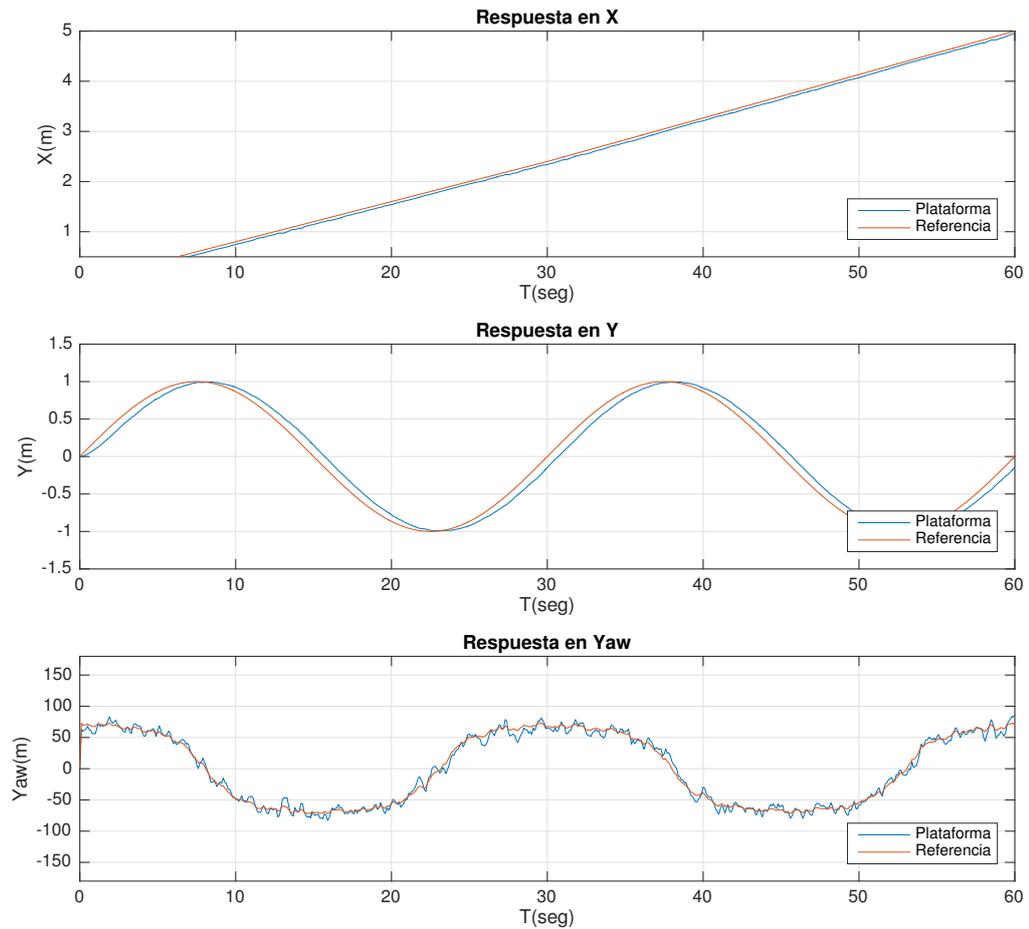


Figura 6.6: Respuesta de los estados  $[X Y \varphi]$  vista de manera separada al realizar la trayectoria de prueba 3 Donde se muestra como la línea que representa al vehículo (Plataforma) trata de seguir la señal deseada con un pequeño desfase por parte de la respuesta en  $[X, Y]$ . Mientras que  $\varphi$  aunque no se muestra del todo uniforme, mantiene una oscilación acotada al rededor de la orientación deseada provocada por la trayectoria.

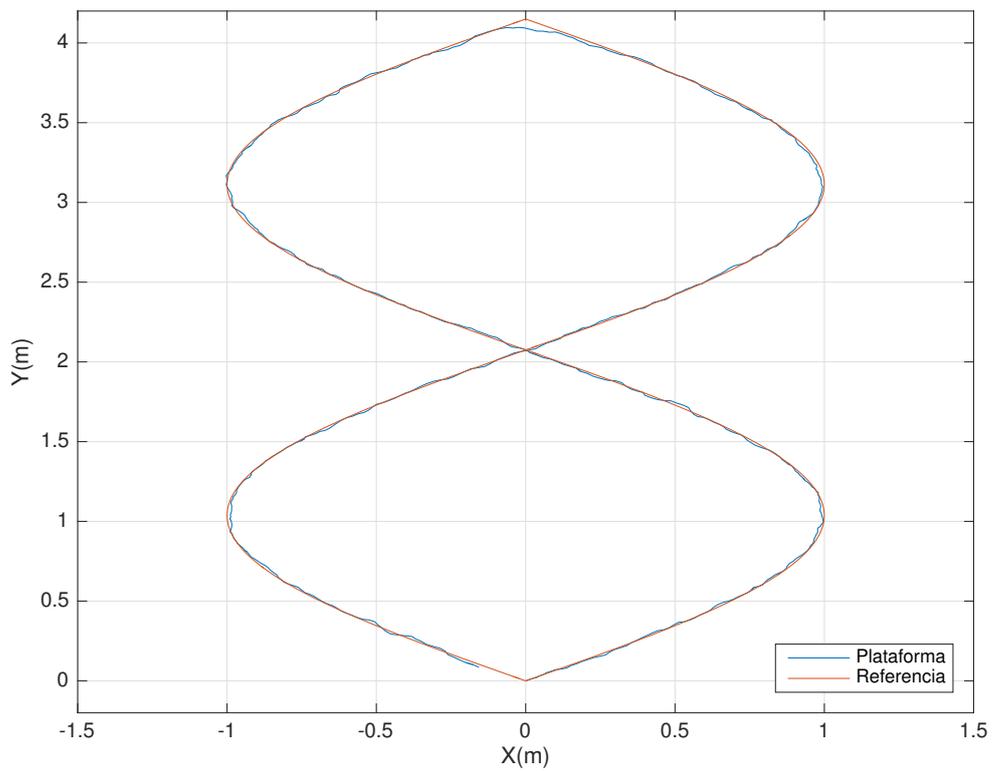


Figura 6.7: Trayectoria de prueba 4: Hélice u Ocho, se parte de condiciones iniciales iguales con cero. Demandando un desplazamiento senoidal para X, mientras que para Y se pide seguir una función triangular.

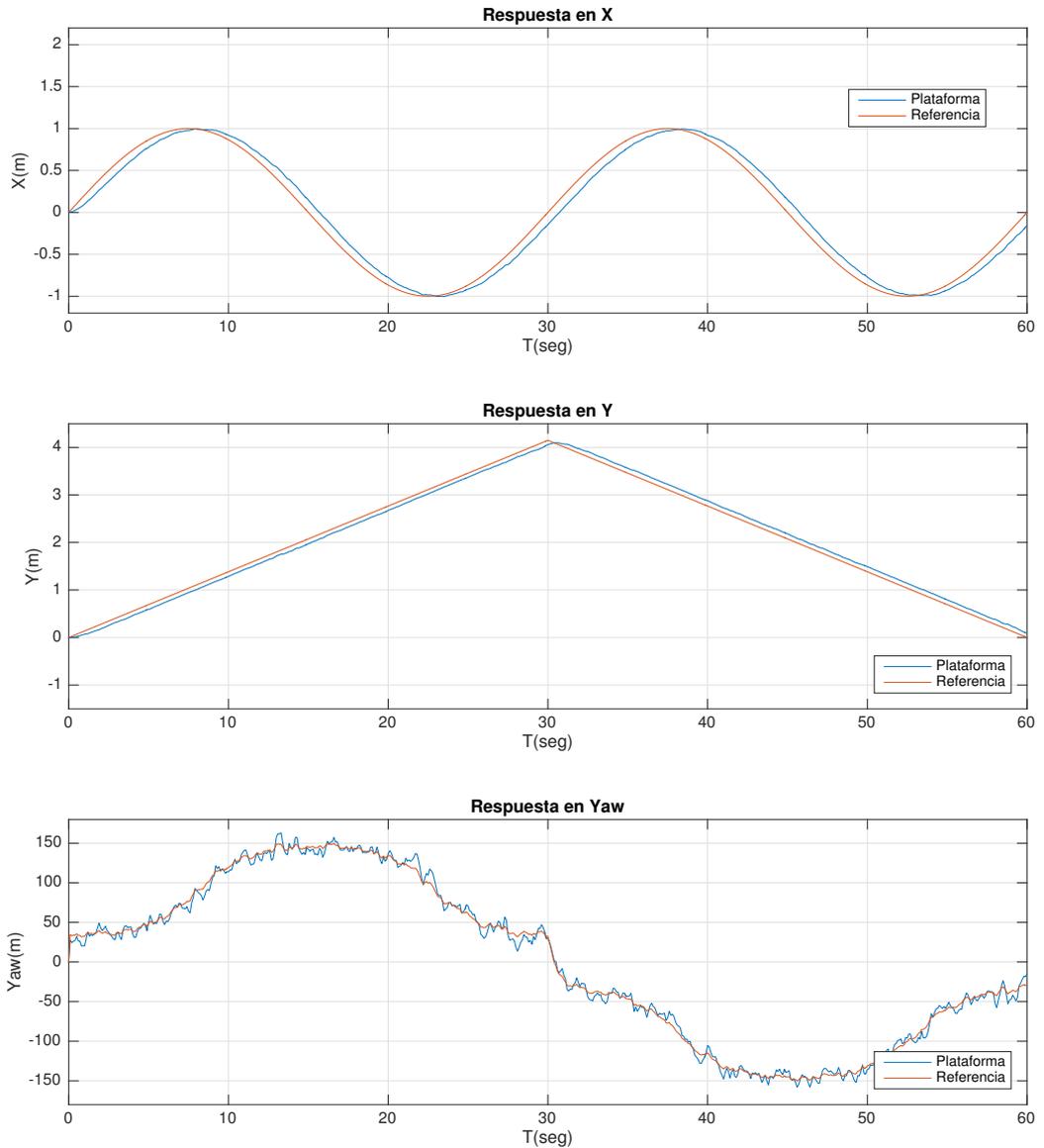


Figura 6.8: Respuesta de los estados  $[X Y \varphi]$  vista de manera separada al realizar la trayectoria de prueba 4 Donde se muestra como la línea que representa al vehículo (Plataforma) trata de seguir la señal deseada con un pequeño desfase por parte de la respuesta en  $[X, Y]$ . Mientras que  $\varphi$  aunque no se muestra del todo uniforme, mantiene una oscilación acotada al rededor de la orientación deseada provocada por la trayectoria.

## 6.1. Simulación 3D bajo entorno Blender

Blender es una suite de creación 3D gratuita y de código abierto. Soporta la totalidad de fuentes de modelado 3D, aparejo, animación, simulación, renderizado, composición y seguimiento de movimiento, incluso edición de video y creación de juegos [16].

Buscando complementar la simulación con un entorno en el que así como el algoritmo de navegación y modelo del vehículo son probados, de igual manera se pueda ejecutar el código bajo Python que será implementado en el vehículo. Es que se opta por trabajar con Blender, realizando un pequeño simulador de pruebas. El cual así como permitirá realizar pruebas de simulación también brinda la posibilidad comprobar el correcto funcionamiento del código sin necesidad de realizar pruebas previas en el vehículo. Mostrando en la Figura 6.9 una vista general del entorno de simulación utilizado.

El entorno de simulación consta de tres bloques:

1. Modelo 3D: Se diseña y coloca el entorno de prueba deseado así como variables propias del entorno, tales como: gravedad, viento, humedad, colisión, masa, sistema de referencia, entre otros.
2. Script de control: Siendo esta la principal parte de interés, donde se incluirá el código que funcionara de manera análoga tanto en el simulador como en el vehículo.
3. Interfaz de edición lógica: Se agregan las variables las cuales regirán la dinámica del vehículo (PID, torque, fuerza, velocidad angular, velocidad traslacional) o de así requerirse una interfaz de control (teclado, joystick, voz, etc.).

## 6.2. Conclusión

De los resultados obtenidos tanto en las gráficas como en el Simulador, se puede observar, que en efecto, el sistema de navegación propuesto para el vehículo cumple con la tarea requerida, siendo capaz de seguir una trayectoria asignada y pudiendo corregir errores provocados por perturbaciones (ruido incorporado a la simulación).

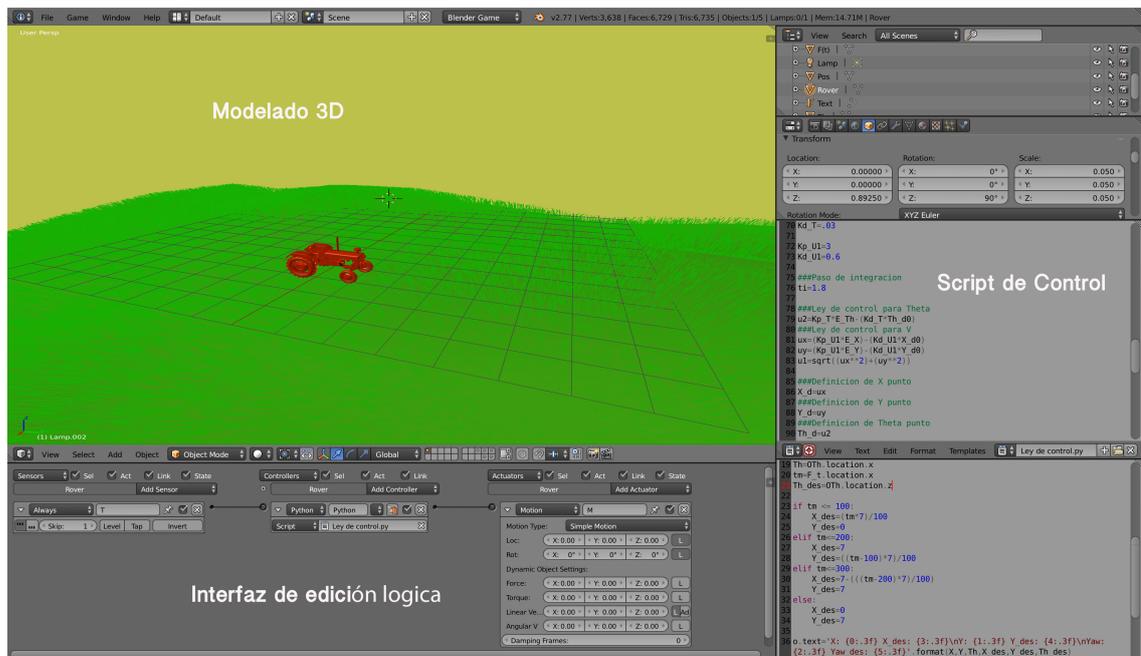


Figura 6.9: Vista del entorno de simulación 3D desarrollado bajo Blender.

---

## Resultados experimentales

---

En esta sección se muestran los resultados experimentales mediante la presentación de las gráficas del desplazamiento del vehículo llevado al exterior (Figura 7.1), así como del comportamiento de los estados bajo el control del Sistema de Navegación. Al comparar los resultados teóricos (simulación) con los experimentales, se puede notar que efectivamente el Sistema de Navegación propuesto tuvo el mismo comportamiento al ser programado en el vehículo. Teniendo como única diferencia la velocidad de ejecución del proceso, ya que en la plataforma la velocidad de ejecución es afectada por la velocidad mas baja por parte de los sensores involucrados en el sistema.

Las pruebas consistieron en someter al Sistema de Navegación a los mismos patrones presentados en el apartado de Simulación para así poder comparar la eficiencia real contra la emulada. De igual manera se agregan dos pruebas mas, una bajo el criterio de regulación y las segunda aplicando el patrón propuesto al realizar una tarea de cultivo.

### 7.1. Problema de regulación

En pocas palabras, el problema de regulación consistirá en dada una referencia fija el vehículo será capaz de llegar a esta a partir de un punto cualquiera. Comprobando así que el vehículo es capaz de dirigirse referencias asignadas de manera dinámica. A continuación se muestran los resultados obtenidos mediante tres gráficas las cuales indicaran: desplazamiento del vehículo (Figura 7.1), comportamiento de los estados y señal de control (Figura 7.2), así como comportamiento del error (Figura 7.3).

Donde el comportamiento de las variables de estado, así como el de las señales de control fue el siguiente:



Figura 7.1: Foto de sistema tomada en área de prueba.

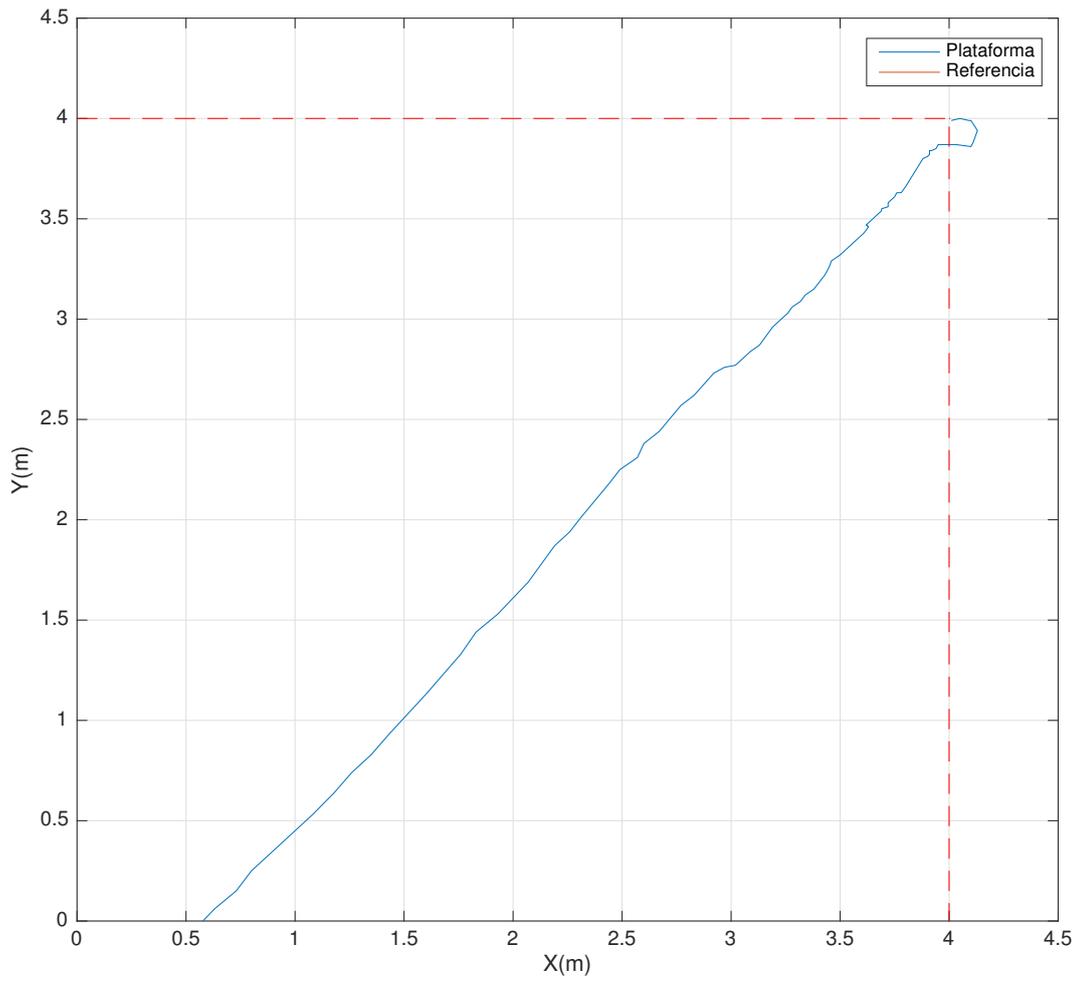


Figura 7.2: Desplazamiento realizado por el vehículo al dirigirse a un punto deseado denotado por las coordenadas (4,4) teniendo como punto de partida la ubicación denotada por las coordenadas(0.6,0).

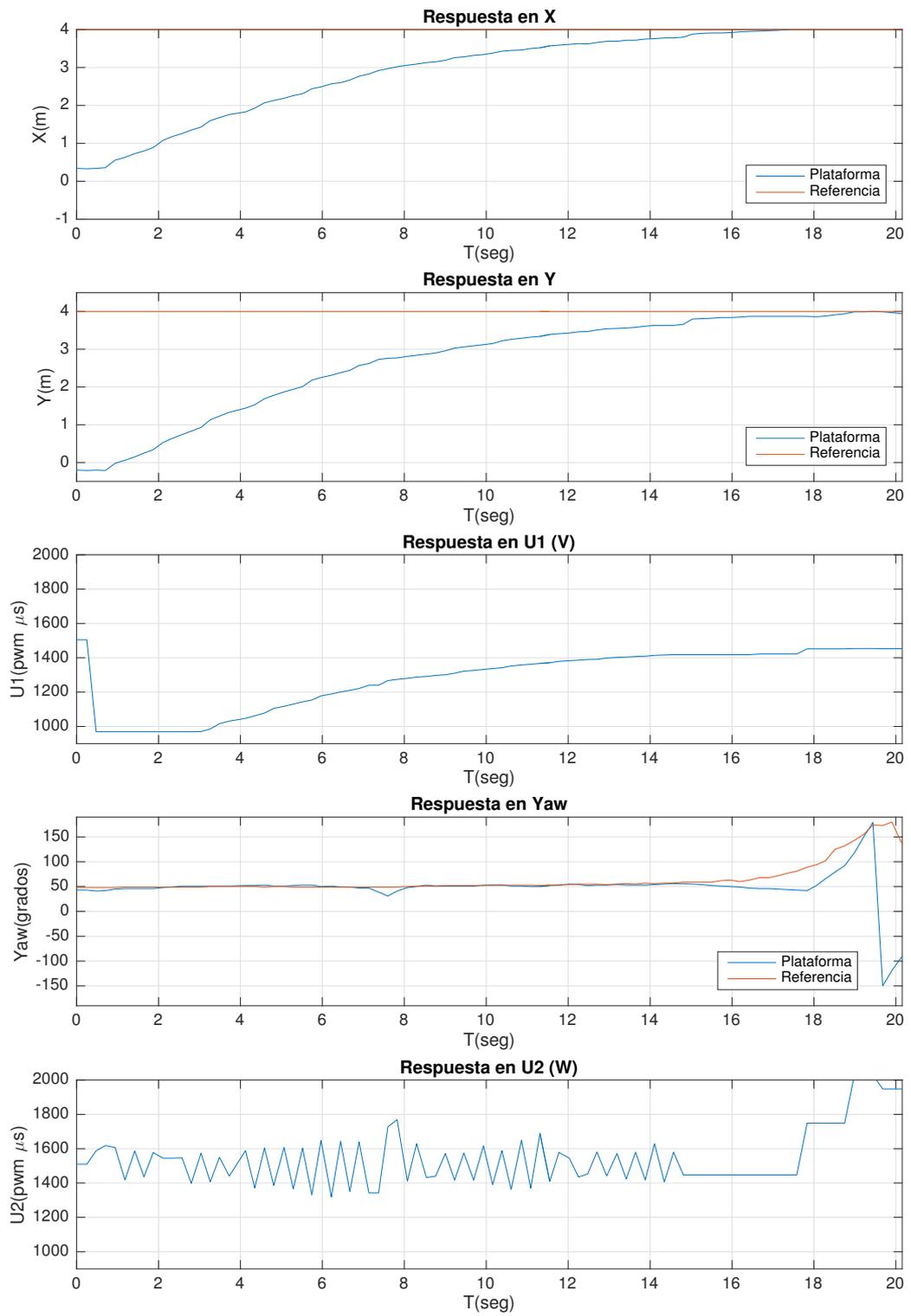


Figura 7.3: Respuesta de las variables de estado (X,Y, Yaw) así como de las señales de control ( $U_1, U_2$ ).

Comportamiento del error:

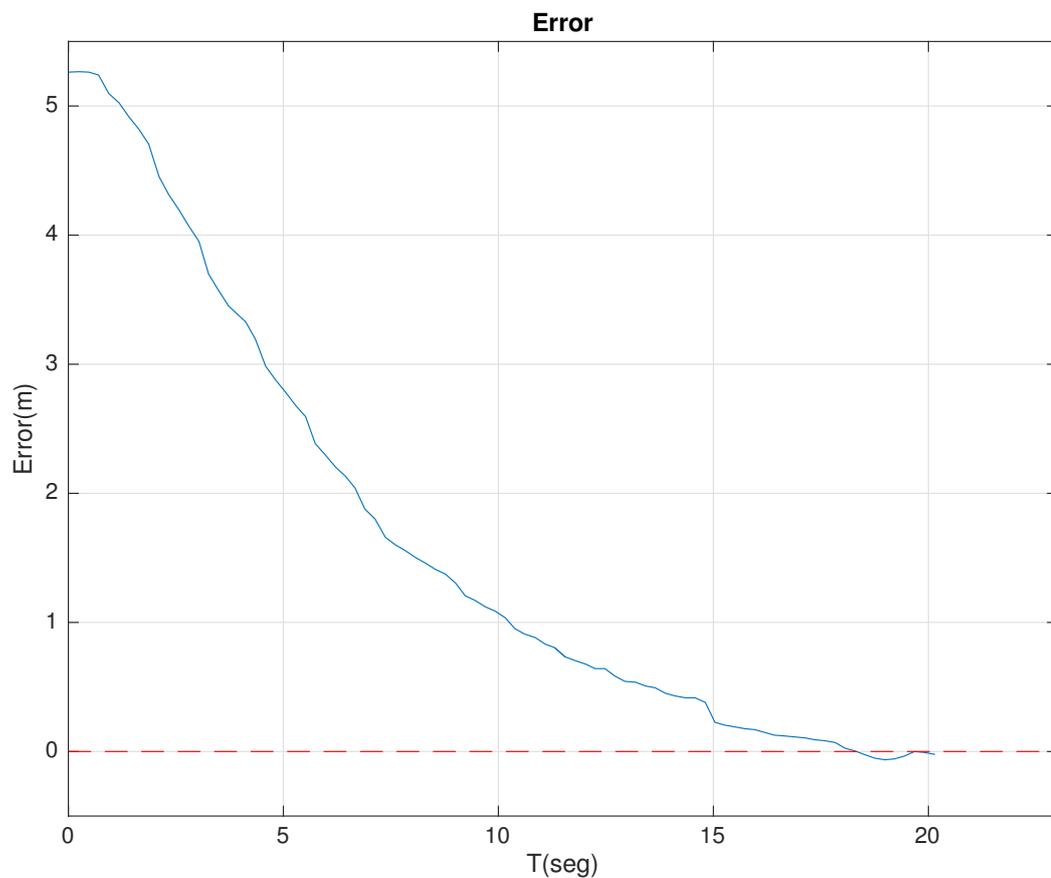


Figura 7.4: Evolución del error durante el desplazamiento del vehículo, mostrando como este tiende a cero, quedando con un valor de estado estacionario de .022m.

## 7.2. Patrón de cultivo

Una vez se ha comprobado que el vehículo es capaz de corregir su posición asíndolo desplazarse a un punto deseado. Se prosigue con realizar el patrón que se designo para dar solución a la tarea de cultivo o inspección de manera autónoma según el algoritmo planteado (Figura 7.4).

De igual manera se presenta una gráfica mostrando el error en el transcurso del recorrido (Figura 7.5) y el comportamiento de las variables de estado, así como el de las señales de control se presentan en la (Figura 7.6):

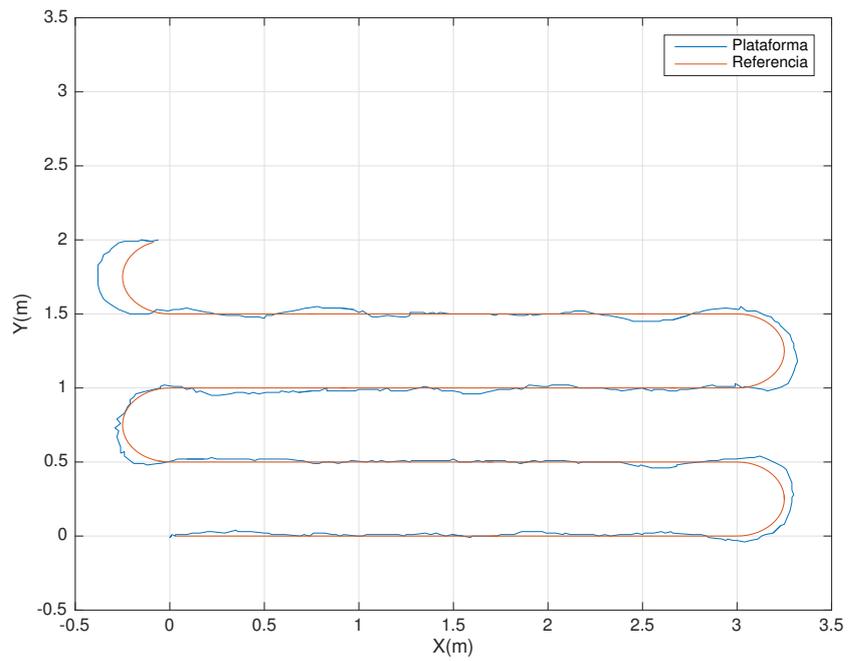


Figura 7.5: Desplazamiento del vehículo siguiendo el algoritmo asignado para realizar la tarea de cultivo de manera autónoma.

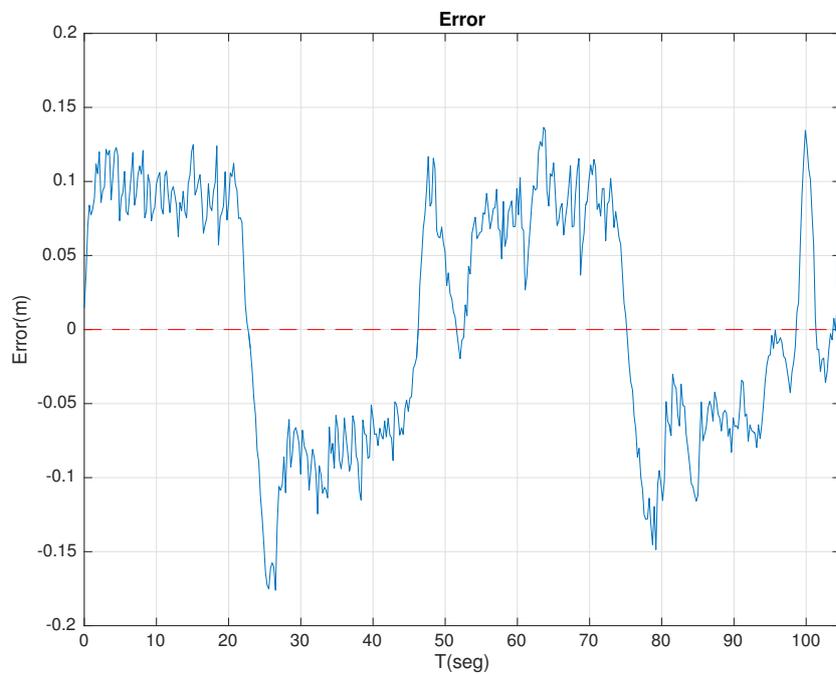


Figura 7.6: Error de posición a lo largo de la trayectoria, donde el valor máximo, mínimo y promedio fueron (0.13664m,  $31794e^{-4}$ m, 0.0769m) respectivamente.

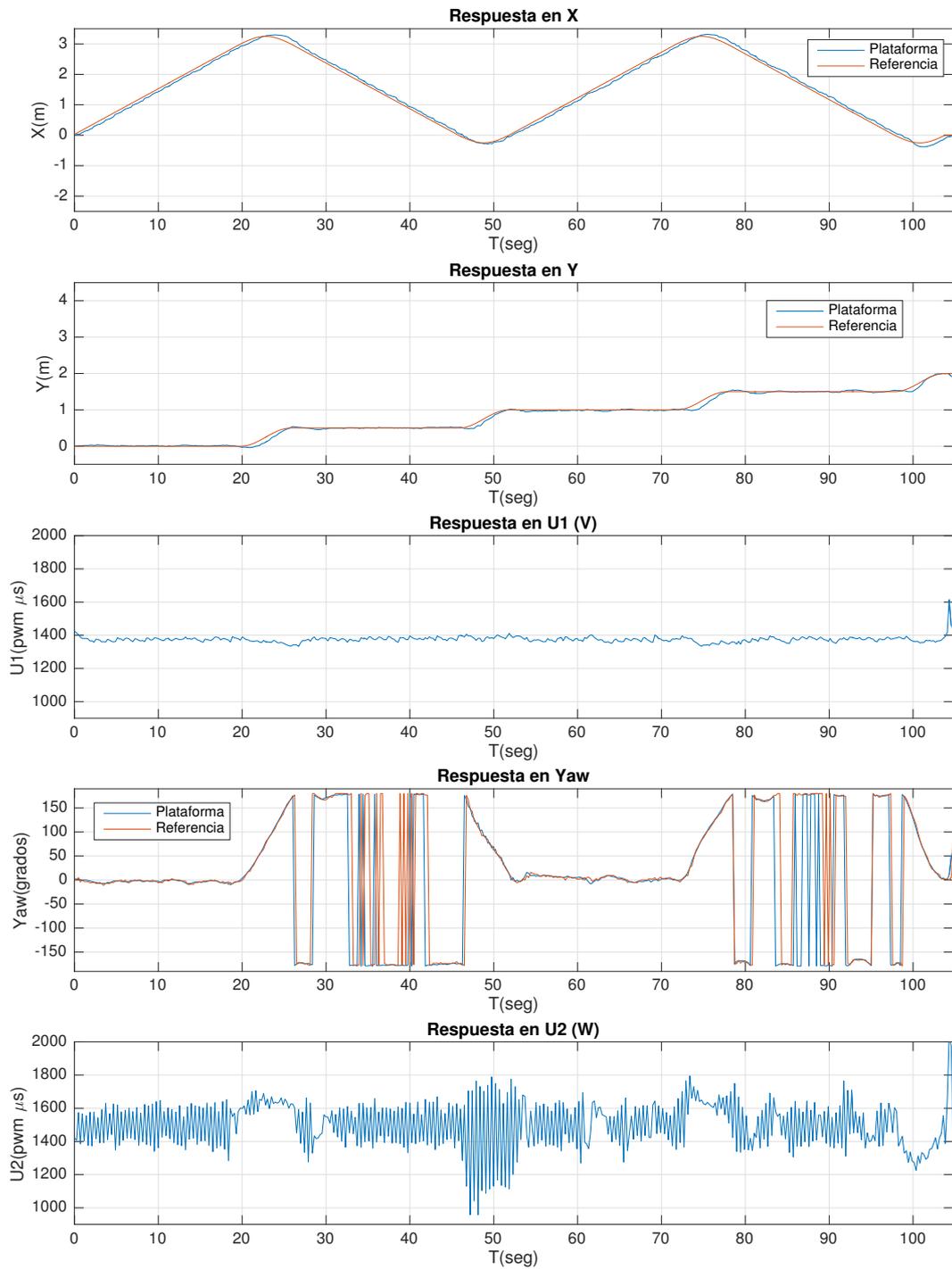


Figura 7.7: Respuesta de las variables de estado ( $X, Y, \text{Yaw}$ ) así como de las señales de control ( $U_1, U_2$ ).

### 7.3. Pruebas extra

Como se ha hecho mención al principio del capítulo, con el fin de comparar los resultados obtenidos en simulación con el comportamiento de la plataforma, se somete el vehículo a los mismos patrones (recta, pendiente, seno y hélice). Con lo cual se demuestra que el algoritmo es lo suficientemente eficiente como para realizar cualquier tarea deseada dentro de las capacidades físicas del vehículo.

Las trayectorias resultantes fueron las presentadas en la Figura 7.7 (recta), Figura 7.9 (pendiente), Figura 7.11 (seno) y Figura 7.13 (hélice). De igual manera se añade una gráfica para cada trayectoria que muestra el error de posición (Figura 7.8, Figura 7.10, Figura 7.12 y Figura 7.14).

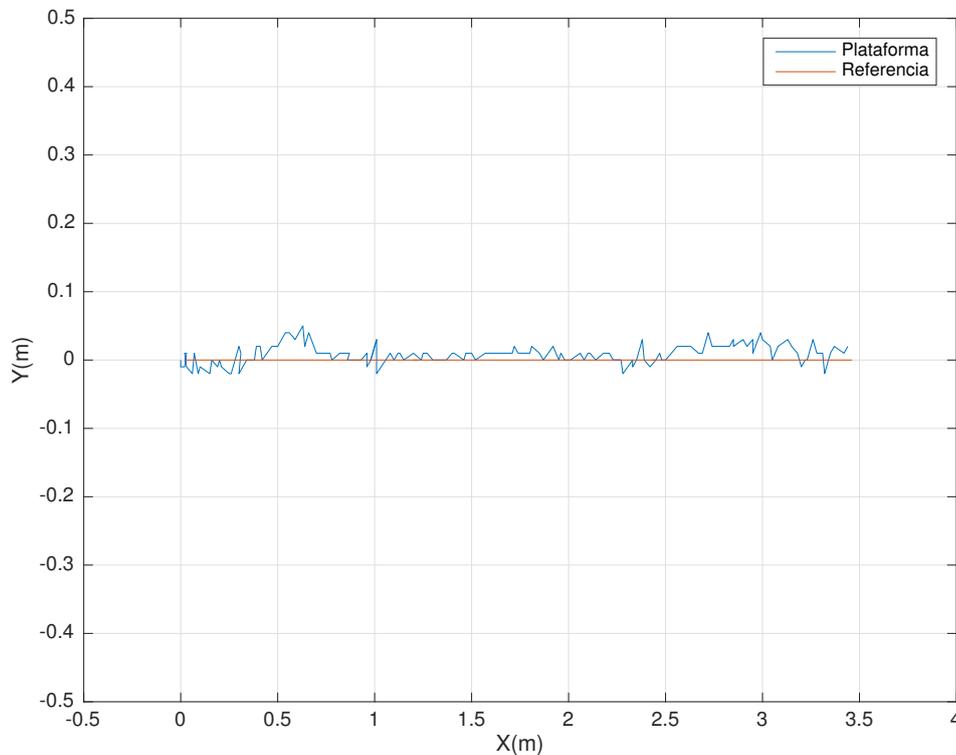


Figura 7.8: Trayectoria uno (recta). Desplazamiento bajo evolución de una variable con condiciones iniciales igual con cero.

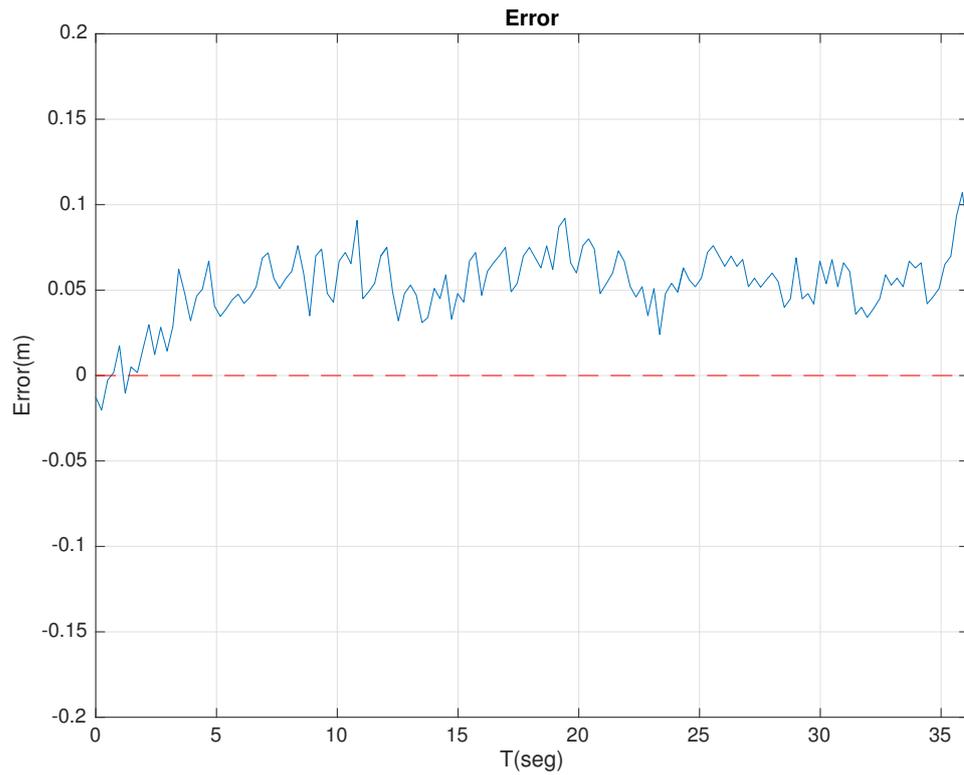


Figura 7.9: Error de seguimiento para la trayectoria uno, donde el valor máximo, mínimo y promedio fueron (0.1071m, 0.0202m, 0.0533m) respectivamente.

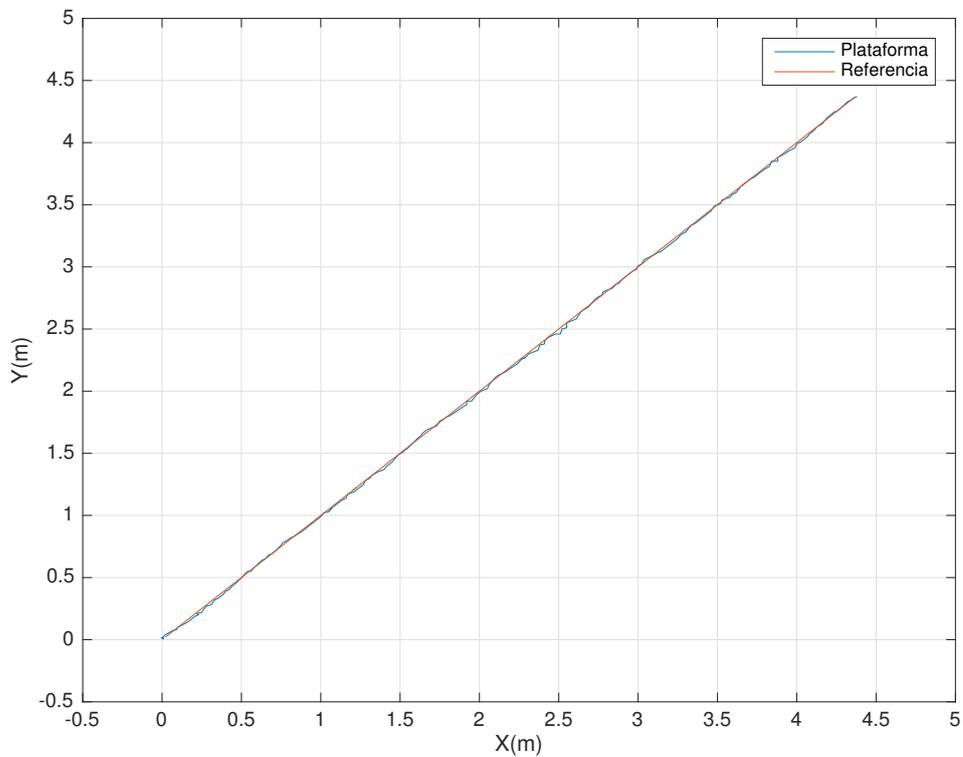


Figura 7.10: Trayectoria dos (Pendiente). Desplazamiento bajo evolución de dos variables con condiciones iniciales igual con cero.

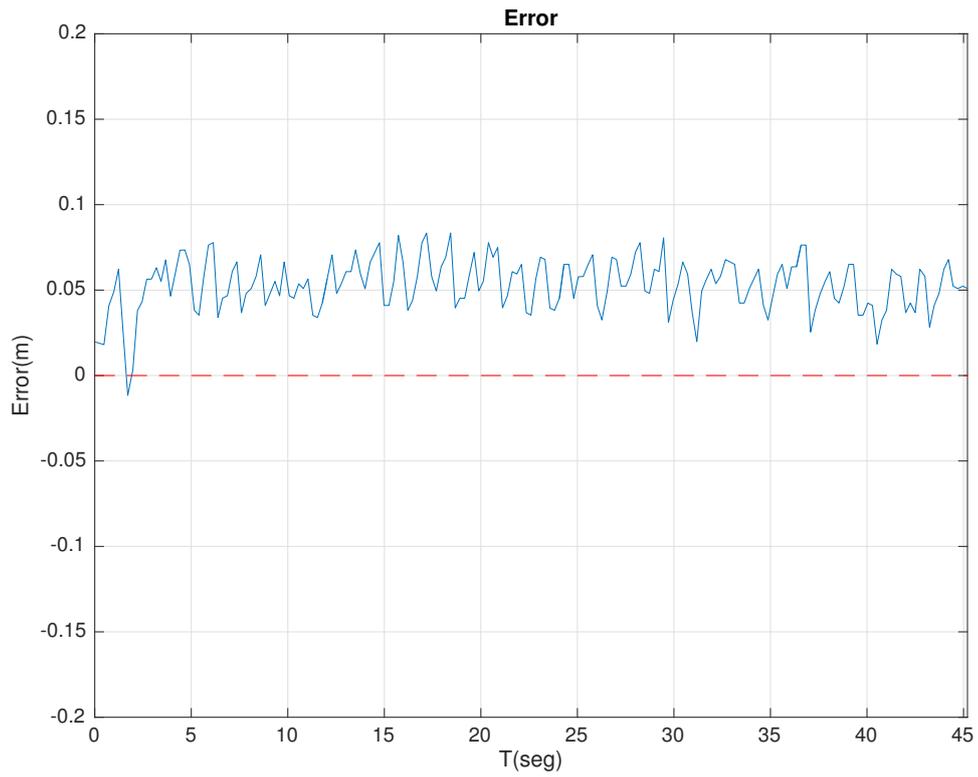


Figura 7.11: Error de seguimiento para la trayectoria dos, donde el valor máximo, mínimo y promedio fueron (0.0834m, -0.0117m, 0.0537m) respectivamente.

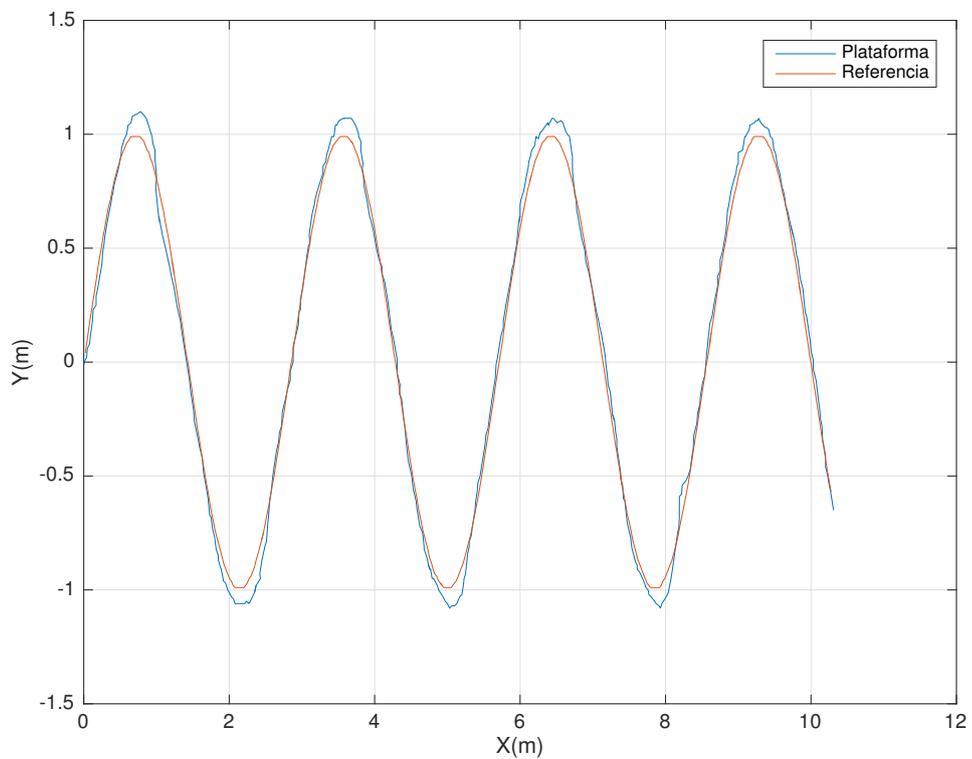


Figura 7.12: Trayectoria tres (seno). Desplazamiento bajo evolución de dos variables con condiciones iniciales igual con cero.

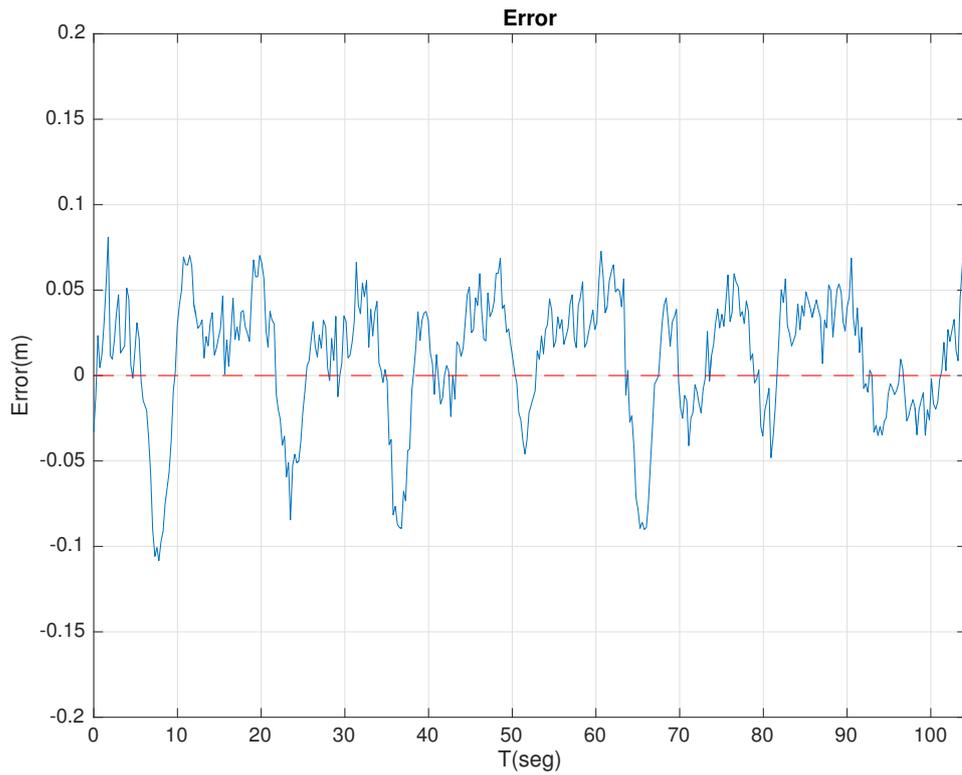


Figura 7.13: Error de seguimiento para la trayectoria tres, donde el valor máximo, mínimo y promedio fueron (0.0957m, -0.1083m, 0.0181m) respectivamente.

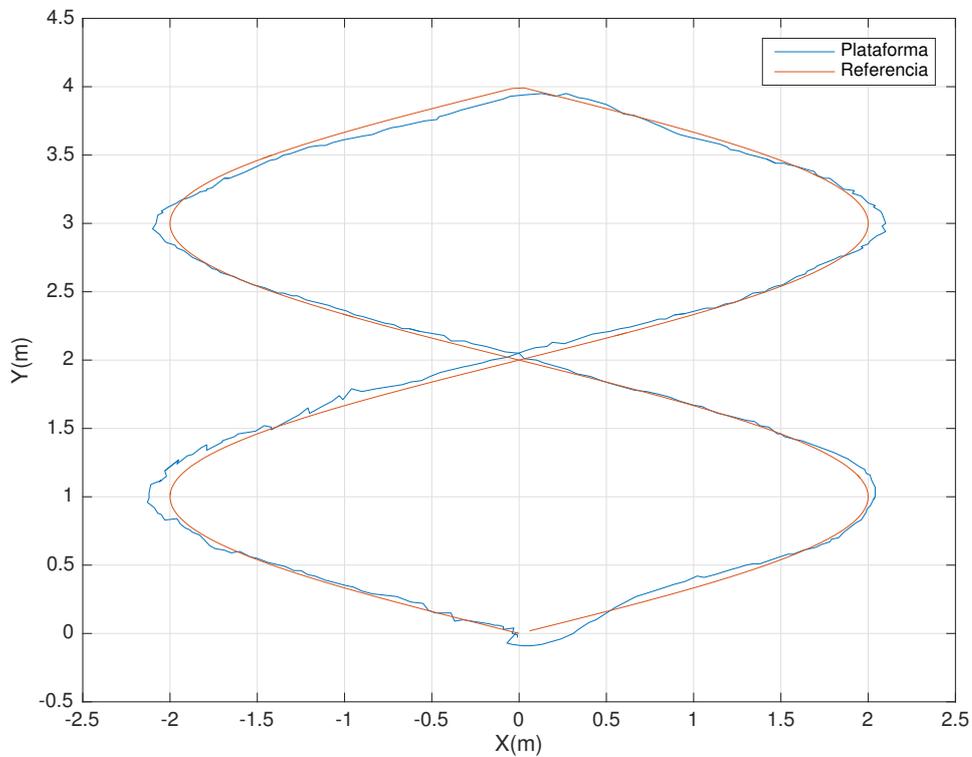


Figura 7.14: Trayectoria cuatro (hélice). Desplazamiento bajo evolución de dos variables con condiciones iniciales igual con cero.

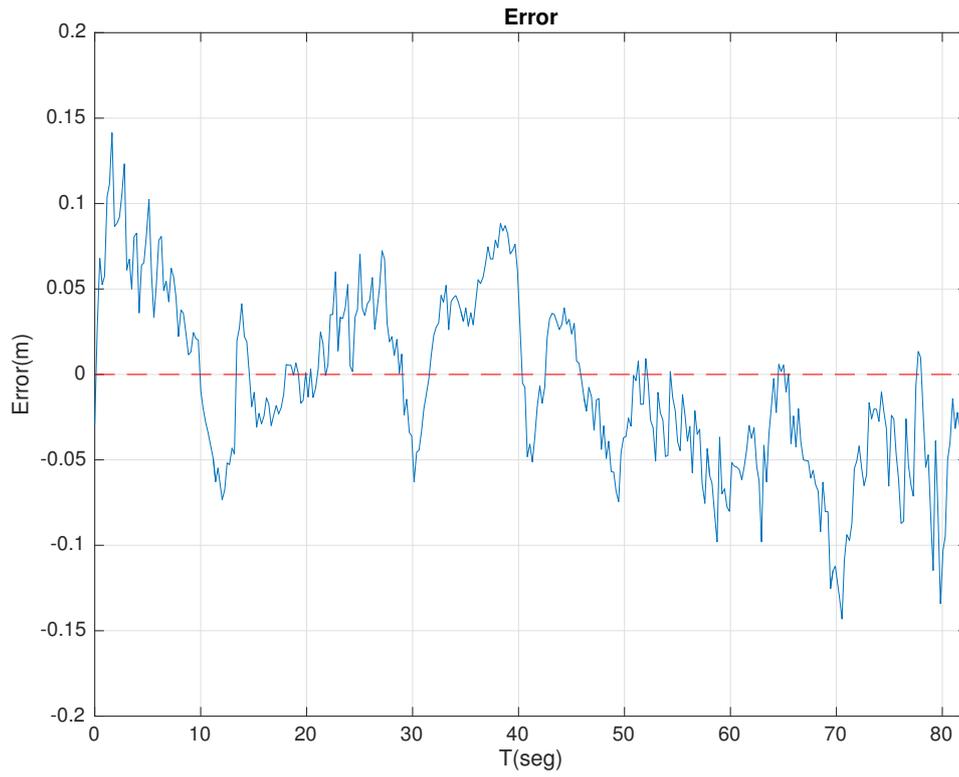


Figura 7.15: Error de seguimiento para la trayectoria cuatro, donde el valor máximo, mínimo y promedio fueron (0.1414m,  $1.4527e^{-4}$ m, 0.0371m) respectivamente.

---

## Conclusiones

---

- Con el fin de incursionar en la aplicación de las tecnologías en el área de la agricultura, es que se logro implementar un sistema de navegación bajo la premisa plug and play. El cual fue diseñado bajo una arquitectura fácil de migrar entre las distintas plataformas sea un vehículo terrestre, un aéreo o un acuático.
- Siendo así, se obtuvo un vehículo terrestre capaz de realizar trayectorias con precisión de centímetros, el cual se utilizará como base para un vehículo más grande que pueda ocuparse en terrenos agrícolas para esparcir fertilizante, o fungicidas en áreas específicas detectadas.
- Aunque el uso de los sistemas RTK brindan una gran precisión, uno de sus principales inconvenientes como ya se menciona, es que para un correcto funcionamiento por parte los receptores GPS es necesario tener un buen estado climático, puesto que al haber presente un alto grado de radiación solar en ocasiones resulta de un gran tiempo de espera antes de alcanzar la solución del IAR y pasar a modo Fix o, incluso en su defecto el no ser posible. Resultando este un inconveniente al momento de plantear el sistema a vehículos aéreos, ya que largas esperas requieren de un consumo de energía y al ser estas plataformas de bajo espacio para fuentes de energía tiene como resultado el limitar el tiempo de autonomía
- La arquitectura con que se desarrollo la programación resulto de gran conveniencia. Ya que al tratarse de una estructura modular permite la fácil inclusión de nuevos módulos para ampliar la complejidad de las tareas a que se someta el sistema. Tal fue el caso del modulo que se implemento para el ajuste de ganancias y el usado al crear el bosquejo con el Lidar.

- Al momento de implementar un autopiloto se trabajó sobre la manipulación de un firmware (APMRover2) y no del diseño de un nuevo. Por lo que se logró conseguir un mayor aprovechamiento por parte de las herramientas que ofrece Pixhawk, siendo desde la manipulación y calibración de sensores de navegación, su compatibilidad con diversos sistemas de monitoreo (GCS) tal es el caso de Mission Planner o APMPlanner. Razón por la cual resulta fácil migrar entre los diferentes firmware.
- El diseño del simulador en Blender resultó de gran ayuda. Ya que al haberse realizado tanto con el mismo lenguaje (Python) y la misma estructura de código con que se trabajó en el vehículo, dio la viabilidad de probar la funcionalidad del script antes de migrarse a la plataforma ahorrando así tiempo al momento de realizar las pruebas de campo.

# Bibliografía

---

- [1] Bøe Røli J. *Cooperative Control and RTK Navigation System for Multirotors*. Norwegian University of Science and Technology, June 2015.
- [2] Camarena J. F., Guerrero G.V. & García C. D. *Análisis Cinemático, Dinámico y Control en Tiempo Real de un Vehículo Guiado Automáticamente*. Tesis de Maestría en Ciencias, Centro Nacional de Investigación y Desarrollo Tecnológico, Departamento de Mecatrónica.
- [3] Cena G., Cibrario I. & Valenzano A. *A socket interface for CAN devices*. *Computer Standards & Interfaces* 29 (2007) 662–673.
- [4] Chakraborty N., Ghosal A. *Dynamic Modeling and Simulation of a Wheeled Mobile Robot for Traversing Uneven Terrain Without Slip*. Department of Mechanical Engineering, Indian Institute of Science, Bangalore–560012, India.
- [5] Chao L., Feng Z., Jingxiang G., Jian W. *Some problems of GPS RTK technique application to mining subsidence monitoring*. *International Journal of Mining Science and Technology* 22 (2012) 223–228
- [6] Coffield D. & Shepherd D. *Tutorial guide to Unix sockets for network communication*. Butterworth & Co (Publishers) Ltd., 1987
- [7] Delgado–Mata C., Velázquez R. & Gutiérrez C. *A Differential–Drive Mobile Robot Driven by an Ethology Inspired Behavior Architecture* Iberoamerican Conference on Electronics Engineering and Computer Science, 2012 .
- [8] García E. & Flego F. *Agricultura de Precisión*. Tecnología Agropecuaria.

- [9] Gan-Mor S., Rex L. C., & Upchurch B. L. *Implement lateral position accuracy under RTK–GPS tractor guidance*. Computers and Electronics in Agriculture 59 (2007) 31–38
- [10] Gracia L. I. & Torneroi J. *Modelado Cinemático y Control de Robots Móviles con Ruedas*. Tesis Doctoral, Departamento de Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia.
- [11] Guerrero J.F., Villarreal M. G. & Sánchez J. P. *Seguimiento de trayectorias de un robot móvil (3,0) mediante control acotado*. Revista Iberoamericana de Automática e Informática industrial 11 (2014) 426–434.
- [12] Hong S. & Kim J. *Visual SLAM with keyframe Selection for Underwater Vehicle*. 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), 2016.
- [13] Hongtao X. *Application of GPS–RTK Technology in the Land Change Survey*. International Workshop on Information and Electronics Engineering (IWIEE), 2012.
- [14] Isogai M., Ito M., Nawa Y & Aiki K. *Navigation Control for an Exploration Rover with Microwave Doppler Sensors*. IEEE 3rd International Conference on System Engineering and Technology 2013.
- [15] Jilek T. *Autonomous field measurement in outdoor areas using a mobile robot with RTK GNSS*. IFAAC–PapersOnLine 48–4 (2015) 480–485.
- [16] K. Ogata. *Ingeniería de control moderna*. Pearson Educación, 1998.
- [17] Kelly R., Santibáñez V. & Loría A. *Control of Robot Manipulators in Joint Space*. Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE), Springer-Verlag London Limited 2005.
- [18] Magdalena C. & Chartuni E. *Manual de Agricultura de Precisión*. Instituto Interamericano de Cooperación para la Agricultura (IICA), 2014.
- [19] Mario Cosenza *Mecánica Clásica*. Universidad de los Andes, Facultad de Ciencias, A–2016.
- [20] Matías L. M. *Agricultura de Precisión*. Ciencia y Tecnología 10, ISEU 2010.

- [21] Omae M., Hasimoto N., Fujioka T. & Shimizu. H. *The application of RTK–GPS and steer-by-wire technology to the automatic driving of vehicles and an evaluation of driver behavior*. IATSS research Vol.30 No.2 2006.
- [22] Park J., Lee J-H., Sang Hyuk Son\* .*Poster Abstract: A Survey of Obstacle Detection using Vision Sensor for Autonomous Vehicles* IEEE 22nd International Conference on Embedded and Real-Time Computing Systems and Applications, 2016 .
- [23] Parrón F., González R., Rodríguez F. & Donaire J. G. *Sistema de Localización para Robots Móviles Basado en GPS. Aplicación a Campos de Golf*. Ctra. Sacramento s/n, E-04120, Almería.
- [24] Pozo A. *Modelo Cinemático y Dinámico del mini Robot Móvil RICIMAF*. RIELAC, Vol.XXXIII 3/2012 p.49-62 Septiembre - Diciembre ISSN: 1815-5928.
- [25] Pozo Ruz A. *Sistema Sensorial para Control y Localización de Vehículos en Exteriores*. Tesis Doctoral, Málaga, mayo de 2001.
- [26] Prempraneerach P., Janthong M., Klanthip T., Boonyarak S., Choosui C., Phothongkum K., Timpitak S. & Kulvanit P. *Navigation Control for an Exploration Rover with Microwave Doppler Sensors*. IEEE 3rd International Conference on System Engineering and Technology, 2013 .
- [27] Prempraneerach P., Janthong M., Klanthip T., Boonyarak S., Choosui C., Phothongkum K., Timpitak S. & Kulvanit P. *Autonomous Way-Point Tracking Navigation of Surveying Surface Vessel with Real-Time Positioning System*. by the Hydro and Agro Informatics Institute (HAI) with contract number HAI 018/2556.
- [28] Raja R., Dutta A. & Venkatesh K.S. *New potential field method for rough terrain path planning using genetic algorithm for a 6–wheel rover*. Robotics and Autonomous Systems 72 (2015) 295–306.
- [29] Raja R., Dutta A., Venkatesh K.S. *New potential field method for rough terrain path planning using genetic algorithm for a 6-wheel rover*. Robotics and Autonomous Systems 72 (2015) 295–306.
- [30] Rieftdorf A., Daub C., & Loeff P. *Precise Positioning in Real–Time using Navigation Satellites and Telecommunication*. 3rd Workshop on Positioning, Navigation and Communication (WPNC'06).

- [31] RÍOS L.H. *Modelo Matemático para un Robot Móvil*. Scientia et Technica Año XIV, No 38, Junio de 2008.
- [32] Rivera D., Eduardo Achá, Javier Bustos J. & José Piquer *Analysis of Linux UDP Sockets Concurrent Performance*. 33rd International Conference of the Chilean Computer Science Society, 2013.
- [33] -32 Suroveca R., Gmitterkoa A., Vackováa M., Virgalaa I., Pradaa E. & Pipíka T. *Design of robot vehicle undercarriage with ability to operate in broken terrain*. MMaMS 2012.
- [34] TAKAI R., Barwid O. & Noguchi N *Autonomous Navigation System of Crawler-Type Robot Tractor*. 18th IFAC World Congress Milano (Italy) August 28 – September 2, 2011.
- [35] Takai R., Barwid O., Ishii K., Noguchi N. *Development of Crawler-Type Robot Tractor based on GPS and IMU*. ifac-agricontrol-2010003-01dec.
- [36] Trimble Navigation Limited. *Real-Time Kinematic Surveying Training Guide*. Trimble Navigation Limited, 1998-2003, .
- [37] Yingst R.A., Cohen B.A., Hynek B., Schmidt M.E., Schrader C. & Rodriguez A. *New potential field method for rough terrain path planning using genetic algorithm for a 6-wheel rover*. Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE), Springer-Verlag London Limited 2005.
- [38] Zheng Q., Li S., Hou B., Lang H. & Wang Y. *An Autonomous Robot Tracking System through Fusing Vision with Sonar Measurements*. IEEE, 2016.