

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**BART: Diseño de controladores de un robot acuático
bio-inspirado de nado superficial**

Trabajo de graduación presentado por Josué David del Cid Ramírez
para optar al grado académico de Licenciado en Ingeniería Mecatrónica

Guatemala,

2021

**BART: Diseño de controladores de un robot acuático
bio-inspirado de nado superficial**

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**BART: Diseño de controladores de un robot acuático
bio-inspirado de nado superficial**

Trabajo de graduación presentado por Josué David del Cid Ramírez
para optar al grado académico de Licenciado en Ingeniería Mecatrónica

Guatemala,

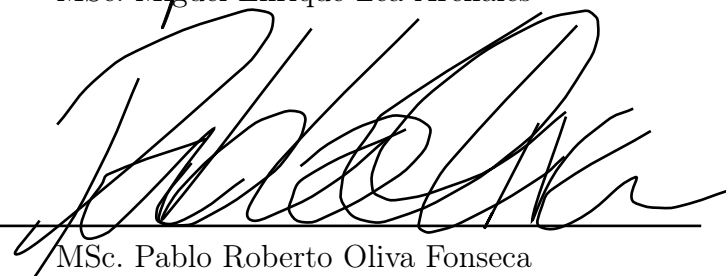
2021


Vo.Bo.:

(f) 
MSc. Miguel Enrique Zea Arenales

Tribunal Examinador:

(f) 
MSc. Miguel Enrique Zea Arenales

(f) 
MSc. Pablo Roberto Oliva Fonseca

(f) 
Dr. Luis Alberto Rivera Estrada

Fecha de aprobación: Guatemala, 15 de enero de 2021.

Todo el trabajo realizado a lo largo de esta tesis es gracias al equipo y laboratorios de la UVG. Actualmente la universidad está profundizando más en proyectos de tecnología, dentro de los cuales cabe resaltar el satélite Cubesat el cual ya fue lanzado al espacio, robots para aplicaciones biomédicas, animatrónicos, robótica swarm y robots bio-inspirados. Este último tema tiene varias ramas de investigación e inspiración en animales como pulgas, liebres, topos, aves y en el caso de este trabajo un escarabajo.

Quiero agradecer a Dios, por haberme permitido llegar este punto, otorgándome cada día la fuerza y sabiduría necesaria para seguir adelante, desde el primer día que comencé la elaboración de este trabajo.

A mis padres, Ivan y Aura, por ser un ejemplo para seguir en cada instante de mi vida. Por su amor, trabajo y sacrificio durante todos estos años, gracias a ustedes he podido llegar hasta aquí. Ha sido un enorme privilegio y orgullo ser su hijo.

A mi hermana, Saraí, por apoyarme incondicionalmente y las palabras de aliento que me impulsaban a seguir a delante.

A mis amigos y amigas por tantos recuerdos compartidos desde el primer día de la carrera, desde esas noches de desvelo hasta viajes inolvidables.

Al ingeniero Miguel Zea, por todo su apoyo, confianza y asesoría brindada al grupo de robótica bioinspirada desde antes de iniciar los trabajos de graduación.

Prefacio	v
Lista de figuras	x
Lista de cuadros	xi
Resumen	xiii
Abstract	xv
I. Introducción	1
II. Antecedentes	3
A. Proyecto BART	3
1. Manta ray robot	10
III. Justificación	11
IV. Objetivos	13
A. Objetivo general	13
B. Objetivos específicos	13
V. Alcance	15
VI. Marco teórico	17
A. Mecanismos	17
1. Grados de libertad (GDL)	17
2. Análisis de posición	18
3. Análisis de velocidad	19
4. Fuerza de arrastre	20
B. Robótica móvil	20
1. Cinemática de robots móviles	20
2. Modelos cinemáticos	21
3. Restricciones cinemáticas de las ruedas	22

C.	Teoría de control	23
1.	Estabilidad	24
2.	Python Dynamics (PyDy)	26
3.	Método de Kane	28
D.	Secciones de Poincaré	30
VII.	Metodología	31
A.	Modelo dinámico	31
B.	Control de velocidad de mecanismos	32
C.	Control de desplazamiento	32
D.	Diseño de placa electrónica	33
E.	Desarrollo de simulaciones	33
VIII.	Dinámica de mecanismo impulsor	35
A.	Mecanismo	35
B.	Python	38
C.	Resultados	39
D.	Fuerza ejercida por el mecanismo	47
IX.	Control por capas	51
A.	Controlador de desplazamiento	52
B.	Integración de controladores	56
1.	Movimiento lineal con giro	56
2.	Desplazamiento horizontal	57
X.	Diseño de placa electrónica	59
A.	Esquemáticos de los circuitos	60
B.	Consumo energético	61
C.	Cálculos para la PCB	62
D.	Listado de componentes	62
E.	Diseño de PCB	63
XI.	Conclusiones	65
XII.	Recomendaciones	67
XIII.	Bibliografía	69
XIV.	Anexos	71
A.	PCB	71
XV.	Glosario	73

Lista de figuras

1.	Prototipos de plataforma BART.[1]	3
2.	Simulación de mecanismo manivela-balancín.[1]	4
3.	Funciones lineales propuestas para describir el comportamiento.[1]	5
4.	Resultados para todos los remos que generaron un comportamiento deseado.[1]	5
5.	Modelo cinemático BART.[1]	6
6.	Trayectoria de simulación y prototipo.[1]	6
7.	Comparación de trayectorias.[1]	6
8.	Comportamiento real y simulado.[1]	7
9.	BART con adaptación de cuatro extremidades.[3]	8
10.	Modelo cinemático equivalente[3]	8
11.	Trayectoria simulada del robot[3]	8
12.	Trayectoria real de BART[3]	9
13.	Sistema motor-mecanismo[1]	9
14.	Trayectoria real de BART usando controlador PID[1]	10
15.	Manta ray Robot[4]	10
16.	Análisis de posición de un mecanismo de 4 barras[5]	18
17.	Velocidad para un eslabón.[5]	19
18.	Marco de referencia global y local.[6]	21
19.	Rueda fija y restricciones.[6]	22
20.	Rueda orientable y restricciones.[6]	23
21.	Rueda orientable descentrada y restricciones.[6]	23
22.	Estabilidad de Lyapunov.[10]	25
23.	Sección de Poincaré.[12]	30
24.	Mecanismo impulsor.	36
25.	Animación en MATLAB.	39
26.	Posición y velocidad angular de juntas independientes.	39
27.	Posición y velocidad angular de juntas dependientes.	40
28.	Trayectoria del efector final.	40
29.	Trayectoria del efector en X y Y.	41
30.	Corte de línea de Poincare.	42
31.	Mapa de retorno de Poincare con coordenadas en Y.	42

32.	Mapa de retorno de Poincare con coordenadas en X.	43
33.	Punto fijo estable X.	43
34.	Punto fijo estable en Y.	44
35.	Análisis de trayectoria.	44
36.	Trayectoria experimental en MATLAB.	45
37.	Mapa de retorno de Poincaré.	45
38.	Trayectoria suavizada.	46
39.	Mapa de retorno de Poincaré.	46
40.	Mapa de retorno de Poincaré.	47
41.	Ciclo de trabajo.	47
42.	Trayectoria de posición vs velocidad angular en la junta E.	48
43.	Simulación de 150 segundos.	49
44.	Simulación de 200 segundos.	49
45.	Control por capas.	51
46.	Coordenada (2,-1.5) posición angular $\pi/2$	54
47.	Coordenada (2,-1.5) posición angular $\pi/2$	54
48.	Seguimiento de trayectoria circular con posición angular variable en el tiempo.	54
49.	Seguimiento de trayectoria circular con posición angular variable en el tiempo.	55
50.	Seguimiento de trayectoria circular con posición angular constante.	55
51.	Seguimiento de trayectoria circular con posición angular constante.	55
52.	Moviente combinado con traslación y rotación.	56
53.	Movimiento combinado con traslación y rotación.	56
54.	Movimiento de traslación en línea recta.	57
55.	Movimiento de traslación en línea recta.	57
56.	Circuito de alimentación y potencia.	60
57.	Modulo de comunicación NRF24L01.	60
58.	Tiva C Launchpad C.	60
59.	Drivers para motores DC.	61
60.	Conexión de motores.	61
61.	PCB.	63
62.	Placa soldada con todos los módulos	71
63.	Diseño en Altium	72

Lista de cuadros

1.	Ciclo de trabajo del mecanismo.	48
2.	Velocidad angular límite.	49
3.	Fuerza y torque producida por el efector final.	50
4.	Consumo energético del BART.	61
5.	Ancho de pistas.	62
6.	Listado de componentes.	62

Los escarabajos de la familia *Gyridae* han inspirado los mecanismos del proyecto BART, que es un robot acuático de nado superficial. En la primera fase del proyecto se desarrollaron los mecanismos impulsores que imitaban la forma en que nadaba el escarabajo. Durante la segunda fase se logró desarrollar una plataforma con movimiento omnidireccional. Esto aumento en gran manera la capacidad de maniobrabilidad y una característica que pocos robots acuáticos poseen. Para alcanzar este tipo de movimiento no fue necesario cambiar el mecanismo subactuado, solamente se colocaron cuatro mecanismos distribuidos como las ruedas de un automóvil.

El objetivo primordial de este trabajo es desarrollar un controlador por capas para el manejo de la plataforma BART. Con este fin, se planteará un modelo dinámico para el mecanismo impulsor. El cual será de gran utilidad para validar resultados obtenidos en las fases anteriores. Para alcanzar los objetivos se planteó el uso de la plataforma Python y MATLAB, donde se realizarán simulación y análisis con métodos numéricos como Poincaré. Además, se tomarán mediciones de las pruebas experimentales utilizando el programa de KINOVEA.

Se dedujo el modelo dinámico del mecanismo impulsor del BART con el cual se pudo comprobar que por sí mismo es capaz de llegar a un ciclo limite en la trayectoria del efector final. También se comprobó que dicho mecanismo durante su fase de empuje se mueve a una velocidad angular constante de -0.76 rad/s la cual produce una fuerza constante $51.7 \frac{\text{mm}g}{\text{s}^2}$ y un torque constante $2666.06 \frac{\text{mm}^2g}{\text{s}^2}$, esto indica que el supuesto del BART como un vehículo de cuatro ruedas es válido.

The beetles of the *Gyridae* family have inspired the mechanisms of the BART project, which is a shallow-swimming aquatic robot. In the first phase of the project, drives were developed that mimicked the way the beetle swam. During the second phase, a platform with omnidirectional movement was developed. This greatly increased maneuverability and a feature that few aquatic robots possess. To achieve this type of movement it was not necessary to change the under-actuated mechanism, only four mechanisms distributed like the wheels of a car will be placed.

The primary objective of this work is to develop a layered controller for managing the BART platform. To this end, a dynamic model for the drive mechanism will be proposed. Which will be very useful to validate results obtained in the previous phases. To achieve the objectives, the use of the Python and MATLAB platform was proposed, where simulation and analysis with numerical methods such as Poincaré will be carried out. In addition, measurements will be taken from the experimental tests using the KINOVEA program.

The dynamic model of the driving mechanism of the BART was deduced with which it was possible to verify that by itself it is capable of reaching a limited cycle in the path of the end effector. It was also verified that said mechanism during its thrust phase moves at a constant angular velocity of -0.76 rad / s which produces a constant force $51.7 \frac{mmg}{s^2}$ and a constant torque $2666.06 \frac{mm^2g}{s^2}$, this indicates that the assumption of BART as a four-wheeler is valid.

Durante mucho tiempo dentro del mundo de la robotica se trabajó en el uso y fabricación de brazos robóticos o máquinas autónomas, sin embargo, durante las últimas tres décadas se empezaron a realizar análisis y estudios sobre la robótica bio-inspirada. Esta rama de la robótica se centra en replicar comportamiento o movimientos de los animales y aplicarlos para realizar una tarea específica. Esto se debe a que la evolución ha permitido que los animales se especialicen en diversas tareas la cuales en su mayoría requieren de un bajo costo energético.

La robótica bio-inspirada está tomando cada vez más fuerza en el ámbito de las investigaciones, además requiere de una gran integración interdisciplinaria. Los temas de interés de esta rama son el diseño de robots más eficientes, sensores, actuadores y la creación de nuevos algoritmos para robótica swarm.

Con el fin de incrementar el conocimiento dentro de este campo es necesario implementar varias plataformas robóticas que creen una nueva frontera en la investigación de robótica bio-inspirada. Durante la última década las universidades en Guatemala han comenzado a sumergirse dentro de este campo. Por lo que han surgido varios prototipos que se han ido mejorando con el pasar de los años con el fin que un día pueda ser de gran ayuda para la industria, investigadores y personas en general. Todas estas tecnologías que han surgido de la robótica bio-inspirada han hecho un gran aporte en la robótica swarm, nuevas formas de desplazamiento para los robots, prótesis de piernas o brazos para los seres humanos.

A. Proyecto BART

Fase I

BART (Bioinspired Aquatic Robot) es, como su nombre lo indica, una plataforma robótica que incorpora un mecanismo innovador de propulsión inspirado en las extremidades de los escarabajos de la familia Gyrinidae. Este robot puede efectuar movimientos sobre la superficie del agua de forma similar a un robot de dos ruedas terrestre. Este utiliza un mecanismo de manivela-balancín, durante el desarrollo se buscaba el control de dicha plataforma de una manera autónoma por medio de radio frecuencia.[1]



(a) Primer prototipo

(b) Prototipo con mejor coeficiente de arrastre

Figura 1: Prototipos de plataforma BART.[1]

El escarabajo de la familia Gyrinidae es uno de los insectos más rápidos y eficientes de nado superficial.[2] El mecanismo bio-inspirado propuesto al igual que el escarabajo, poseen una mayor área al final de sus extremidades como un remo lo que les permite moverse con mayor velocidad en el agua.

El mecanismo manivela balancín diseñado durante esta fase logró replicar el movimiento de la pata de un escarabajo al nadar, además, se extendió el eslabón de salida para agregar la aleta. La cual es responsable de producir un mayor impulso en el agua. Para ello se utilizaron motores Polulu Micro Metal Geared motor w/Encoder con una relación de 75:1, siendo estos la única entra en el mecanismo. En la Figura 2 se muestra el análisis de trayectoria generada por el punto E.[2]



Figura 2: Simulación de mecanismo manivela-balancín.[1]

Para obtener el comportamiento dinámico del mecanismo de propulsión dentro del agua, se construyó una plataforma de pruebas. Desde esta plataforma se buscaba determinar el modelo a partir de diferentes ciclos de trabajo del motor DC seleccionado. Los resultados en la *Figura 3*, muestran dos funciones lineales que describen el comportamiento. La primera función representa los ciclos de trabajo que no tiene suficiente torque para mover el sistema, y la segunda donde el sistema ya puede moverse fácilmente. Es decir, que el ciclo de trabajo de los motores debe ser superior al 40% para vencer la inercia del robot.

Además, se variaron las áreas de los remos con el fin de identificar qué diseño alcanzará la mayor velocidad. Como se observa en la *Figura 4*, el remo con área de $30 \times 40 \text{ mm}^2$ obtuvo el mejor desempeño. De este modo, se llegó a la conclusión que dependiendo de los ciclos de trabajo el sistema presentaba movimiento a velocidad constante.

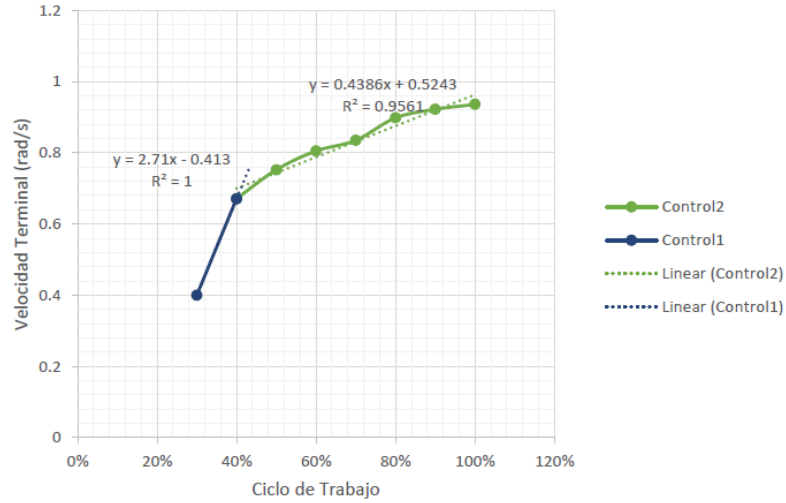


Figura 3: Funciones lineales propuestas para describir el comportamiento.[1]

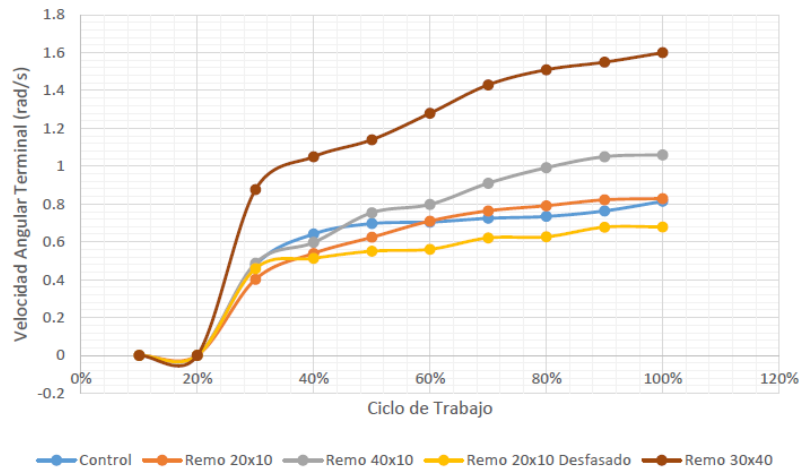


Figura 4: Resultados para todos los remos que generaron un comportamiento deseado.[1]

En esta fase, se logró determinar un modelo matemático que representara el movimiento del robot sobre el agua para el cual se partió de un modelo cinemático de una plataforma robótica con cuatro ruedas. Al ser este simplificado se analizaron las dos ruedas delanteras como se muestra en la Figura 5 esto debido a que BART solo poseía mecanismos de impulso delanteros y no traseros.

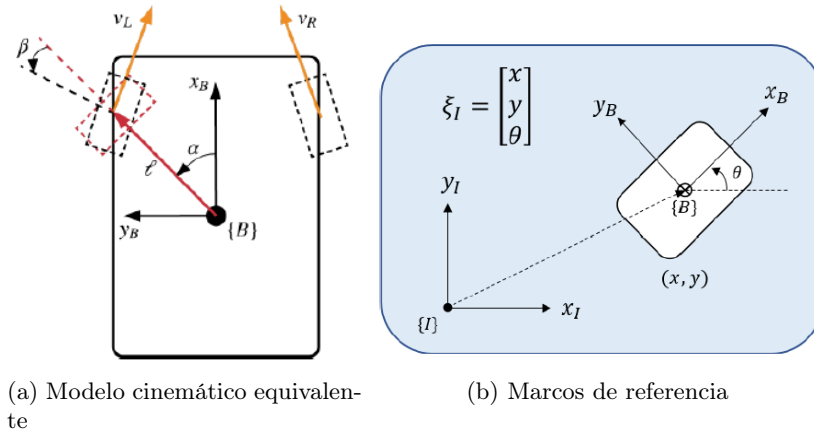


Figura 5: Modelo cinemático BART.[1]

Según los resultados obtenidos del modelo matemático de la cinemática del robot, y las simplificaciones de este, se llegó a la conclusión que se trataba de un modelo unicyclo. Esto permitió realizar pruebas de trayectorias en simulación y en la implementación, tal como se muestra en las Figuras 6 y 7.

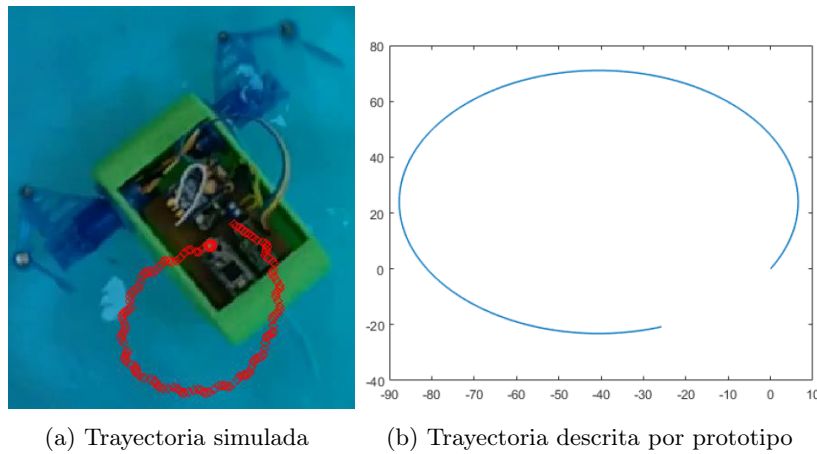


Figura 6: Trayectoria de simulación y prototipo.[1]

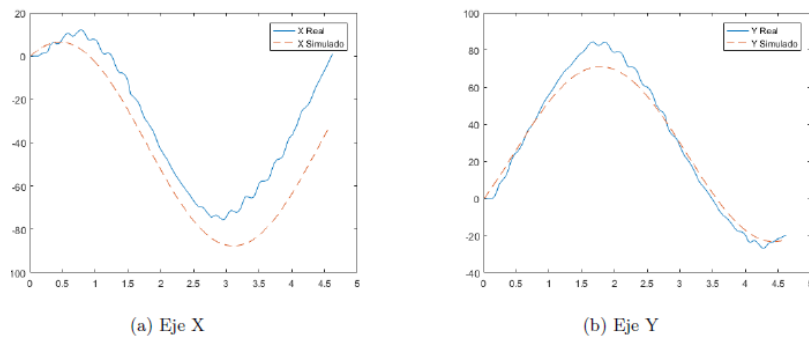


Figura 7: Comparación de trayectorias.[1]

Una vez que se obtuvo el modelo matemático, se diseñó e implementó un sistema de control cinemático para el nado superficial. Por consiguiente, se realizaron pruebas con controladores LQR, PI lineal y exponencial y de pose no lineal. Al analizar los resultados se determinó que el controlador PI con un acercamiento exponencial era el indicado para la plataforma. La trayectoria descrita por este controlador se ve en la *Figura 8*.

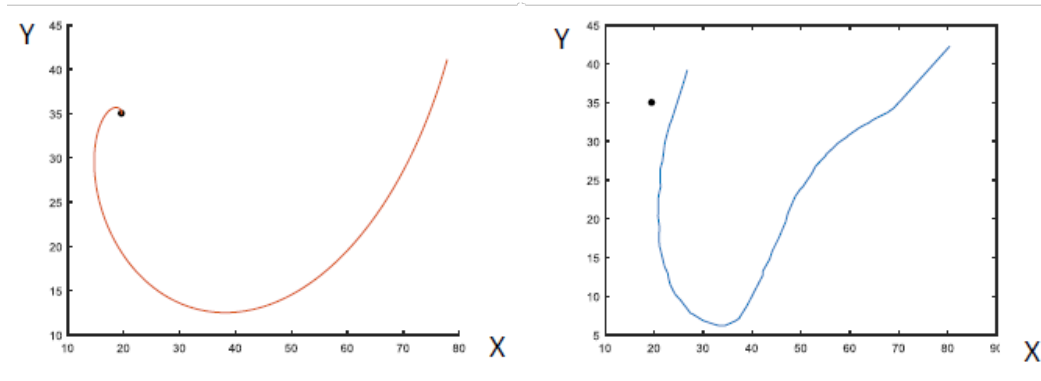


Figura 8: Comportamiento real y simulado.[1]

Asimismo, se diseñó y fabricó una placa electrónica para obtener información, procesarla y poder ser transmitida a otro sistema. En este caso se logró enviar la información de forma inalámbrica mediante ZigBee a un sistema de cómputo. En esta parte se implementaron dos algoritmos. El primero contenía los codificadores rotatorios, el nado en círculos, orientación de la unidad de medición inercial (IMU) y estimación de pose. El segundo obtuvo la orientación del robot, con procesamiento de imágenes.

Fase II

El prototipo presentado para nado superficial en esta fase presentó cuatro mecanismos impulsores, instalados en el casco de forma que permitió un desplazamiento omnidireccional como se observa en la Figura 9. Además, en cada una de las fases se buscaba mejorar o desarrollar nuevas formas de impermeabilizar la plataforma, por lo que el casco fue impreso en resina impermeable. [3]

El primer modelo obtenido de BART en la fase anterior solo tenía dos ruedas. Esto permitía que la plataforma solo se pudiera mover hacia adelante, ya que no podía desplazarse de forma lateral al menos que los motores funcionaran a distintos ciclos de trabajo o solo esté funcionando uno. A partir de esto se planteó la extensión del modelo matemático a 4 ruedas las cuales permitan un movimiento omnidireccional, en la Figura 10 se aprecia el modelo cinemático equivalente.

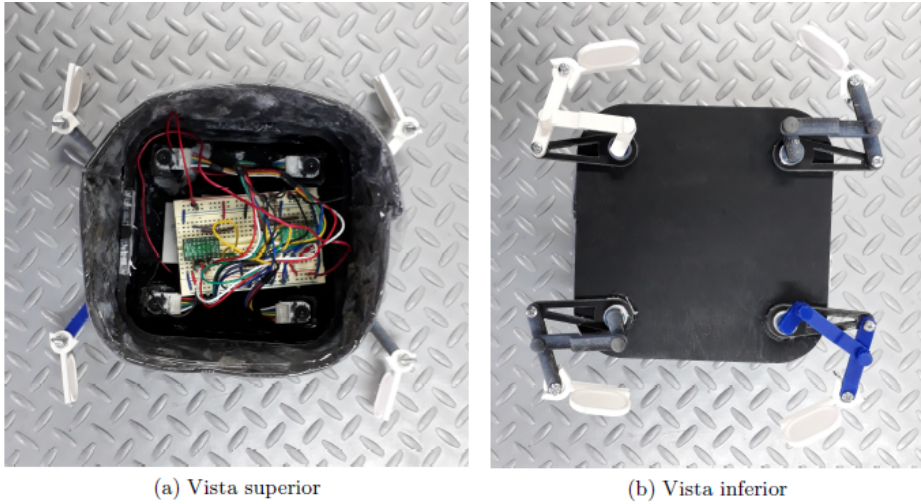


Figura 9: BART con adaptación de cuatro extremidades.[3]

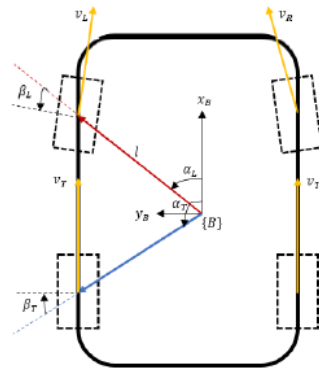


Figura 10: Modelo cinemático equivalente[3]

Después de determinar el nuevo modelo matemático, se procedió a realizar pruebas de movimiento. Esto consistía en que la plataforma robótica se desplazará de forma lateral sin tener que girar previamente sobre su eje.

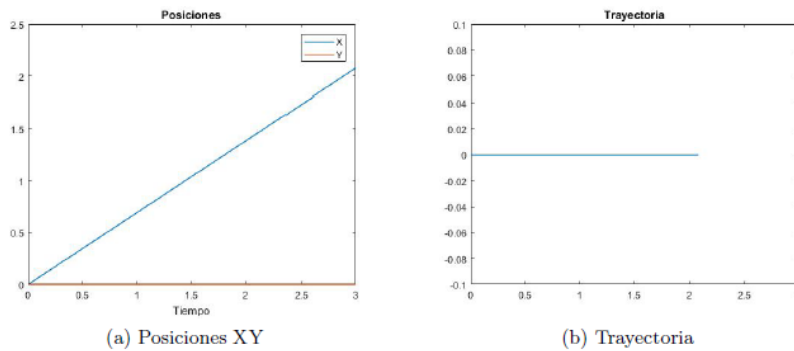


Figura 11: Trayectoria simulada del robot[3]

Estas pruebas de movimiento lineal de BART reflejaron que para realizar estos movimien-

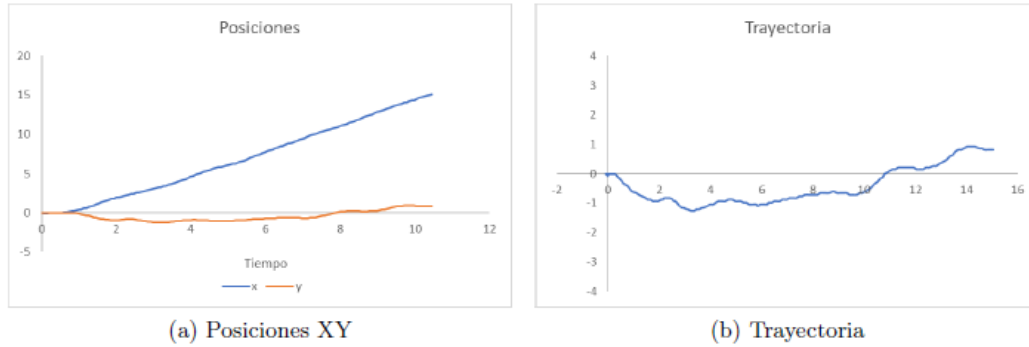


Figura 14: Trayectoria real de BART usando controlador PID[1]

1. Manta ray robot

Investigadores de la Universidad Nacional de Singapur (NUS) crearon un robot manta raya que nada más rápido y durante más tiempo que los vehículos submarinos autónomos tradicionales. El robot acuático se mueve a la velocidad del doble de la longitud de su cuerpo por segundo, lo que equivale a 0,7 metros por segundo, y funciona durante hasta 10 horas. Se deslizan sin esfuerzo por el agua agitando sus aletas pectorales. A diferencia de otros robots submarinos basados en aleteo que replican la cinemática de aleteo de las mantarrayas mediante el uso de múltiples motores para lograr accionamientos activos en todas las aletas, este funciona con un solo motor eléctrico en cada aleta. [4]



Figura 15: Manta ray Robot[4]

Con una longitud de 35 cm, 63 cm de ancho y 0,7 kg de peso, las aletas del robot manta ray están hechas de láminas de PVC. Estos reemplazan a los propulsores basados en hélices están conectados a un cuerpo plano y ancho que puede acomodar sensores para estudios marinos, datos hidrográficos y operaciones de búsqueda. Con sus características de rendimiento y capacidad reforzadas, tiene como objetivo mostrar cómo podría ser el futuro para futuros vehículos autónomos submarinos.

BART es una plataforma robótica bioinspirada que incorpora un mecanismo de propulsión innovador, inspirado en las extremidades inferiores de los escarabajos *Gyrinidae*. Este robot de nado superficial en el futuro se podrá utilizar para realizar tareas multiagente, recolección de basura o monitoreo y medición de distintas propiedades fisicoquímicas en cuerpos de agua. También puede ser utilizado para el estudio de la biología y ecosistemas acuáticos. Esto ayudara a la universidad e investigaciones en estas áreas ya que existen pocas plataformas para utilizarse en el agua.

Al igual que en muchos sistemas electromecánicos, el control es fundamental para un correcto funcionamiento. Por lo tanto, es necesario obtener y analizar el modelo dinámico del mecanismo, ya que sin este modelo no se podrá realizar un sistema de control funcional. Con un sistema de control eficaz es posible obtener diferentes modos de funcionamiento en los cuales será posible controlar la posición del robot, la orientación y trayectorias de desplazamiento.

Un control por capas nos permitirá relacionar el modelo dinámico del mecanismo con el control del motor. Así mismo, relacionar la velocidad que imparte el mecanismo sobre el modelo cinemático del robot y este modelo podrá ser colocado en un control de generación o planificación de trayectorias. Basado en que el robot presenta movimiento omnidireccional, pero sin forma de orientarlo y de establecer una posición objetivo. Se tratará de proponer un control por capas que integre el control de velocidad para los mecanismos y otro para desplazamiento del robot. Para ello se requerirá restablecer la plataforma de comunicación realizada durante la fase I.

A. Objetivo general

Implementar y diseñar una metodología de control por capas, para lograr el control de la plataforma robótica BART.

B. Objetivos específicos

- Determinar y simular el modelo dinámico de los mecanismos impulsores utilizados en las patas del BART.
- Diseñar e implementar un controlador cinemático para el desplazamiento del BART.
- Desarrollar una placa electrónica para un microcontrolador, donde se ejecutarán los algoritmos de control y la comunicación inalámbrica de la plataforma robótica.

En la fase II de BART se extendió el modelo de la fase I a un robot de cuatro patas, bajo ciertos supuestos y permitiendo un movimiento omnidireccional al desplazarse sobre el plano x,y . El presente trabajo busca desarrollar los controladores para el control de movimiento, de modo que se logre modelar analíticamente la dinámica del mecanismo impulsor.

Durante la últimas fases se realizaron análisis de trayectoria del mecanismo y pruebas de impulso a distintas revoluciones. Al final se obtuvieron controladores de desplazamiento para el modelo de dos patas y solo un controlador de velocidad para el motor del modelo de cuatro patas. En este proyecto no se incluirá el control por visión para el manejo de trayectorias o interpretación de su entorno, debido a las limitantes de desarrollo que presentó la pandemia de COVID-19 del año 2020. Por lo tanto, las pruebas realizadas no se pudieron realizar en un entorno de laboratorio. Los controladores utilizados se escogerán en base a los resultados y recomendaciones de las fases anteriores.

En la segunda fase se trabajó la electrónica a nivel de protoboard, por lo que para este trabajo se diseñara una placa electrónica en donde todos los módulos se puedan ensamblar de una forma fácil y ordenada. No se planteará la realización de un PCB en donde se integren todos los componentes, ya que no se cuenta con la disponibilidad de herramientas necesarias para llevar a cabo este tipo de trabajo.

A. Mecanismos

Un mecanismo es un dispositivo que permite transformar un movimiento arbitrario en un patrón deseado, por medio de la restricción del movimiento relativo. Por lo general estos dispositivos generan fuerzas muy bajas y no transmiten grandes cantidades de potencia. Para su estudio los mecanismos se pueden tratar como dispositivos cinemáticos (sin considerar las fuerzas) y funcionan a velocidades bajas. Por otro lado, si los niveles de energía o de fuerzas son considerables se consideran como máquinas, y es necesario un análisis dinámico.[5]

1. Grados de libertad (GDL)

Es el número de entradas que se necesita proporcionar para crear una salida predecible por lo que, es fundamental tanto para la síntesis como para el análisis de mecanismos. Para determinar los GDL en algún mecanismo es necesario conocer la cantidad de eslabones y juntas que este posee. Usando la ecuación de Gruebler¹ es posible determinar los GDL de cualquier mecanismo.[5]

$$M = 3(L - 1) - 2J \tag{1}$$

Donde:

- $M = \text{GDL}$
- $L = \text{número de eslabones}$
- $J = \text{número de juntas}$
- $G = \text{número de eslabones conectados a tierra}$

Por lo que para un mecanismo de 4 barras cerrado se necesitara solo 1 GDL para definir su posición, velocidad y aceleración.

2. Análisis de posición

Para cualquier mecanismo con un GDL, tal como uno de cuatro barras, se requiere solo un parámetro para definir por completo las posiciones de todos los eslabones. El parámetro usualmente elegido es el ángulo θ_2 del eslabón de entrada. Se requiere hallar θ_3 que corresponde al acoplador y θ_4 es el ángulo del eslabón de salida.

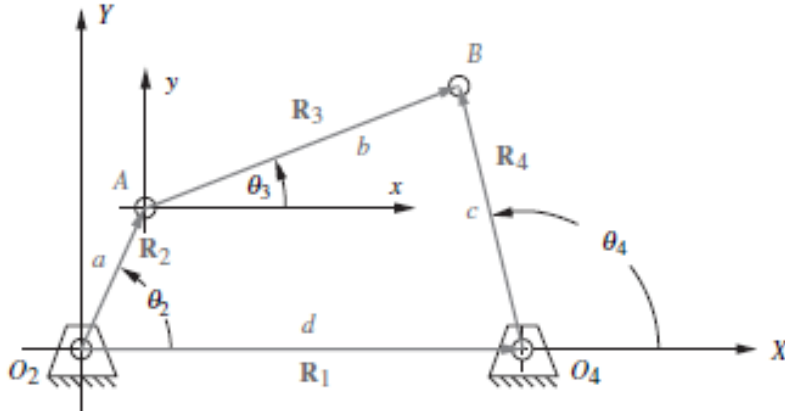


Figura 16: Análisis de posición de un mecanismo de 4 barras[5]

Las direcciones de los vectores de posición en la Figura 16 se eligen de modo que definan sus ángulos donde se desea medirlos. Por definición, el ángulo de un vector siempre se mide en su raíz, no en su cabeza. Se quiere medir el ángulo θ_4 en el pivote fijo O_4 , de modo que el vector R_4 se acomoda para que su raíz quede en ese punto. Una lógica similar dicta el arreglo de los vectores R_3 , R_2 y R_1 . Estas elecciones de las direcciones y sentidos de los vectores, indicados por sus puntas de flecha, conducen a la ecuación de lazo vectorial:[5]

$$R_2 + R_3 - R_4 - R_1 = 0$$

Esta ecuación deja ver que la variable independiente es θ_2 , la cual puede controlarse con un motor u otro dispositivo motriz. Esto deja a los ángulos del eslabón 3 y 4 por ser encontrados. Se necesitan expresiones algebraicas que definan θ_3 y θ_4 como funciones solo de las longitudes constantes de los eslabones y del ángulo de entrada, θ_2 . Al realizar el análisis respectivo para θ_4 y θ_3 se obtiene lo siguiente:[5]

$$K_1 = \frac{d}{a} \quad K_2 = \frac{d}{c} \quad K_3 = \frac{a^2 - b^2 + c^2 + d^2}{2ac} \quad K_4 = \frac{d}{b} \quad K_5 = \frac{c^2 - b^2 - a^2 - d^2}{2ab}$$

Por motivos de simplificación se realiza lo siguiente:

$$A = \cos(\theta_2) - K_1 - K_2 \cos(\theta_2) + K_3 \quad C = K_1 - (K_2 - 1) \cos(\theta_2) + K_3$$

$$D = \cos(\theta_2) - K_1 - K_4 \cos(\theta_2) + K_5 \quad F = K_1 - (K_4 - 1) \cos(\theta_2) + K_5$$

Obteniendo

$$\theta_{4,2} = 2 * \tan^{-1}\left(\frac{\sin(\theta_2) \pm \sqrt{\sin^2(\theta_2) - AC}}{A}\right) \quad (2)$$

$$\theta_{3,2} = 2 * \tan^{-1}\left(\frac{\sin(\theta_2) \pm \sqrt{\sin^2(\theta_2) - DF}}{D}\right) \quad (3)$$

Las ecuaciones 2 y 3 tiene dos posibles soluciones, obtenidas a partir de las condiciones \pm en el radical. La solución por lo general será real y desigual, lo cual significa que existen dos valores para θ_4 y θ_3 que corresponden a θ_2 . Estas se conocen como las configuraciones cruzada y abierta del mecanismo.

3. Análisis de velocidad

Luego de realizar el análisis de posición para un mecanismo es posible determinar la velocidad de cada eslabón en ciertos puntos de interés. La velocidad es la tasa de cambio de la posición con respecto al tiempo. Se denominará \mathbf{r} a la posición y es una cantidad vectorial. La velocidad puede ser calculada como angular o lineal. La velocidad angular se denota ω y la velocidad lineal como V , entonces [5]

$$\omega = \frac{d\theta}{dt} \quad V = \frac{d\mathbf{r}}{dt} \quad (4)$$

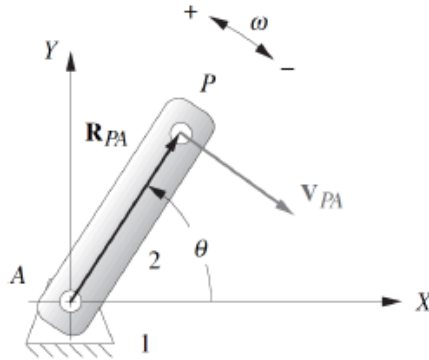


Figura 17: Velocidad para un eslabón.[5]

La velocidad V_{PA} que se observa en la *Figura 17* se conoce como velocidad relativa y esta se define como:

$$V_{PA} = V_P - V_A \quad (5)$$

De esta ecuación se puede obtener la velocidad en el punto P al despejar para V_P . Esto también se puede realizar a partir de la velocidad angular para cualquier punto de interés, de modo que, si se conoce el centro de rotación del cuerpo se conoce la componente de la velocidad rotacional y el sentido será el mismo que la velocidad angular ω del cuerpo. El método de solución es válido para cualquier punto en el eslabón, basta con conocer las

longitudes de todos los eslabones, posiciones angulares de los eslabones y la velocidad de entrada instantánea de cualquier punto motriz. [5]

4. Fuerza de arrastre

En dinámica de fluidos, el arrastre (a veces llamado resistencia del aire, un tipo de fricción, o resistencia a fluidos, otro tipo de fricción o fricción de fluidos) es una fuerza que actúa opuesta al movimiento relativo de cualquier objeto que se mueve con respecto a un fluido circundante. Esto puede existir entre dos capas fluidas (o superficies) o un fluido y una superficie sólida. A diferencia de otras fuerzas resistivas, como la fricción seca, que son casi independientes de la velocidad, las fuerzas de arrastre dependen de la velocidad. La fuerza de arrastre es proporcional a la velocidad de un flujo laminar y la velocidad al cuadrado de un flujo turbulento. Aunque la causa última de un arrastre es la fricción viscosa, el arrastre turbulento es independiente de la viscosidad. Esta fuerza está dada por la siguiente ecuación:

$$F = \frac{1}{2}\rho v^2 AC_D \quad (6)$$

Donde,

- ρ es la densidad del fluido
- A es el área de contacto del cuerpo con el fluido
- v es la velocidad lineal del cuerpo
- C_D es el coeficiente de arrastre

B. Robótica móvil

Todo robot móvil puede trasladarse en el entorno para el cual fue diseñado mediante mecanismos. A diferencia con los robots en base a mecanismos de eslabonamiento, en la locomoción de un robot móvil, los entornos están fijos y el robot genera fuerza a dicho ambiente para generar movimiento. Esta interacción con el entorno genera una cinemática y dinámica que pueden llegar a ser modelados.[6]

1. Cinemática de robots móviles

La posición de un cuerpo rígido en un plano bidimensional se obtiene a partir de un marco de referencia inercial I correspondiendo este a las coordenadas globales. Además, se pueden definir la posición y orientación de otros objetos a partir del marco de referencia del robot R siendo las coordenadas locales, tal como se muestra en la Figura 18. Ya que se considerará al robot como un cuerpo rígido, las coordenadas locales R se miden a partir de un punto P el cual normalmente este dado por el centro de masa, aunque se puede escoger otro punto a conveniencia.[6]

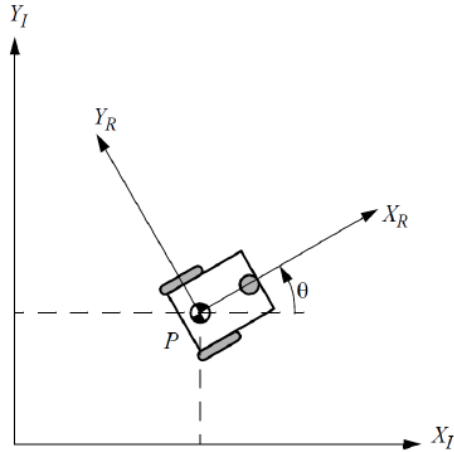


Figura 18: Marco de referencia global y local.[6]

Si se tiene el desplazamiento del robot, siendo este el punto p y su orientación θ es posible describir su pose con el vector:

$$I_{\xi} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (7)$$

Para poder describir el movimiento del robot y su entorno en términos de su propio marco de referencia, es necesario emplear una transformación homogénea, la cual mapea de un marco de referencia a otro.[6]

$$T = \begin{bmatrix} \cos(\theta) & \sin(\theta) & x \\ -\sin(\theta) & \cos(\theta) & y \\ 0 & 0 & 1 \end{bmatrix} \quad (8)$$

2. Modelos cinemáticos

La rueda desde sus inicios ha sido uno de los mecanismos de locomoción más simples y eficientes de implementar. Ya que estas siempre permanecen en contacto con la superficie, el diseñador debe preocuparse por la estabilidad, maniobrabilidad y control del sistema. Para ello, se necesita entender el comportamiento mecánico del sistema y las tareas del robot móvil, por lo que la vía más fácil para su comprensión es la cinemática.[6]

Además, el modelo del robot está fuertemente relacionado con la clase de ruedas que este posee, ya que estas implican distintas restricciones para el sistema. Dicho lo anterior es indispensable definir los marcos de referencia global y local y al final se tiene que encontrar una relación entre ellos y obtener como resultado la cinemática de cada rueda o del robot como tal.[6]

3. Restricciones cinemáticas de las ruedas

Rueda estándar fija

Como se aprecia en la Figura 19 este tipo de rueda posee un único grado de libertad, el cual está dado por su velocidad lineal v . Como resultado su movimiento se limita hacia atrás y adelante. La posición de la rueda A está dada por la distancia l y un ángulo α . También se observa β como el ángulo del plano de la rueda respecto al casco del robot que será un valor fijo ya que la rueda no puede cambiar la orientación.

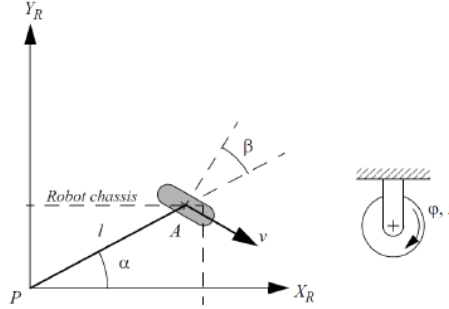


Figura 19: Rueda fija y restricciones.[6]

Teniendo identificados los parámetros de la posición de la rueda y cinemática del robot, se pueden dar las siguientes restricciones:

- De rodamiento

$$\begin{bmatrix} \sin(\alpha + \beta) & -\cos(\alpha + \beta) & -l \cos(\beta) \end{bmatrix} R(\theta) \dot{\xi}_I - r \dot{\varphi} = 0 \quad (9)$$

- De deslizamiento

$$\begin{bmatrix} \cos(\alpha + \beta) & \sin(\alpha + \beta) & l \sin(\beta) \end{bmatrix} R(\theta) \dot{\xi}_I = 0 \quad (10)$$

En la condición de deslizamiento no debe existir un movimiento en el plano ortogonal al de la rueda y por tal razón se iguala a cero. Para la condición de rodamiento, este término debe ser igual a $r \dot{\varphi}$ que es el movimiento realizado por el giro de la rueda.[6]

Rueda orientable

Esta es parecida a la rueda fija, con la diferencia de que el ángulo de rotación puede variar, tal como se observa en la *Figura 20*.

Las restricciones de deslizamiento y rodamiento son las mismas que para la rueda fija, esto se debe a que la velocidad de $\dot{\beta}$ no tiene un impacto directo en el movimiento a comparación de $\dot{\varphi}$. Este cambio de dirección únicamente afecta el movimiento del robot.

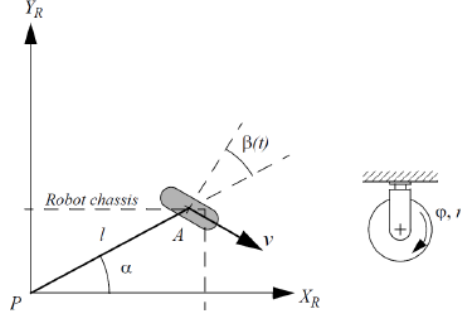


Figura 20: Rueda orientable y restricciones.[6]

Rueda orientable descentrada

Esta posee las mismas restricciones que la rueda anterior, sin embargo, el ángulo β gira con respecto a un eje que se encuentra a una distancia d de su centro.[6]

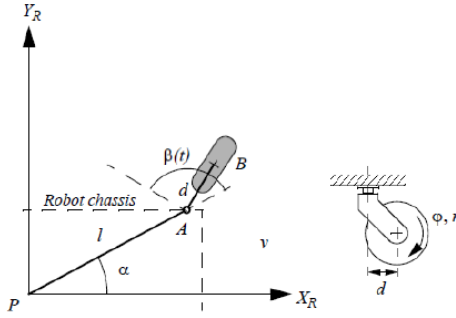


Figura 21: Rueda orientable descentrada y restricciones.[6]

La restricción de rodamiento será igual a la rueda fija. Ahora bien, para la restricción de deslizamiento se debe realizar cambios. Esto es consecuencia de una fuerza lateral aplicada en A , ya que este es el punto de acople entre la rueda y el vehículo.

$$[\cos(\alpha + \beta) \quad \sin(\alpha + \beta) \quad d + l \sin(\beta)] R(\theta) \dot{\xi}_I + d \dot{\beta} = 0 \quad (11)$$

Esta ecuación dice que todo movimiento ortogonal al plano de la rueda debe ser balanceado por una cantidad equivalente y opuesta de movimiento de rotación descentrada. En otras palabras, un robot cuyo movimiento dependa exclusivamente de ruedas orientables descentradas puede moverse a cualquier velocidad en el espacio posible de movimientos robóticos, por lo que se consideraría omnidireccional.

C. Teoría de control

Esta disciplina su objetivo principal es el controlar sistemas para que realicen una tarea. La teoría de control se puede agrupar en dos grandes grupos, de control clásico y moderno.

El control clásico fue la primera que utilizo, desarrollo y aun se puede ver en muchos sistemas de la actualidad, especialmente dentro de la industria. La característica principal del control clásico es su uso de técnicas en el dominio de frecuencia.

Una de las deficiencias de este tipo de control son las condiciones iniciales, ya que este utiliza la función de transferencia, en donde se descartan estas condiciones. Otra limitante es que al diseñar un controlador se obtiene a partir del dominio de la frecuencia, esto implica que el sistema controlado ejecutara siempre la misma tarea. Además, en el control de sistemas multivariados acoplados, donde una variable depende de otra es casi imposible controlar ambas variables. [7]

Utilizando la teoría de control moderno se busca eliminar las limitantes del control clásico. Esto es posible utilizando técnicas de algebra lineal y acercamientos a sistemas dinámicos. Esta teoría utiliza las ecuaciones diferenciales que describen el sistema, las cuales son colocadas de forma vectorial. [8]

1. Estabilidad

En la teoría de control la estabilidad es un aspecto que siempre se toma en cuenta al momento de aplicar control a un sistema. La estabilidad permite conocer el comportamiento a largo plazo del sistema sin necesidad de resolver las ecuaciones diferenciales que describen su dinámica. [7]

Punto de equilibrio

Dado un sistema representado en variables de estado: [9]

$$\begin{aligned}x &\in \mathbb{R}^n & u &\in \mathbb{R}^k & y &\in \mathbb{R}^m \\ \dot{x} &= f(x, u) \\ y &= h(x, u)\end{aligned}\tag{12}$$

Se tiene que $(x_{ss}, u_{ss}) \in \mathbb{R}^n \times \mathbb{R}^k$ es un punto de equilibrio si

$$f(x_{ss}, u_{ss}) = 0.\tag{13}$$

Estabilidad en el sentido de Lyapunov

El teorema de estabilidad de Lyapunov permite el análisis de los sistemas de control no lineales. La estabilidad en el sentido de Lyapunov establece que un estado de equilibrio x_{ss} es estable si dado un número real $\rho > 0$ existe otro número real $\delta > 0$ de forma que todo estado inicial $x(t_0)$ para el que se cumpla

- implica que:

$$\|x(t_0) - x_{ss}\| < \delta \quad (14)$$

$$\|x(t) - x_{ss}\| < \rho, \quad \forall t > t_0 \quad (15)$$

Debido a que por sí solo Lyapunov no es una condición suficiente para los sistemas de control, esta tiene que ser complementada con el concepto de estabilidad asintótica. Llegados a este punto, un estado de equilibrio x_{ss} es asintóticamente estable, si es estable según el teorema de Lyapunov y la trayectoria que describa el sistema permanezca dentro de la región $S(\delta)$ converge hacia x_{ss} sin abandonar $S(\rho)$ en ningún instante.

Este estado de equilibrio se dice que es global, si toda solución converge hacia x_{ss} . Además, el punto x_{ss} puede ser inestable si $\rho > 0$ existe otro número $\delta > 0$ exista un estado $x(t_0)$ para el cual las trayectorias cumplen. [10]

$$\|x(t) - x_{ss}\| > \rho \quad (16)$$

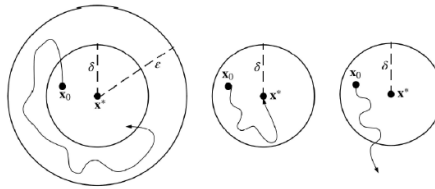


Figura 22: Estabilidad de Lyapunov.[10]

Teorema de estabilidad de Lyapunov

Sea $x_{ss} = 0$ un punto de equilibrio de $\dot{x} = f(x)$, $D \subset \mathbb{R}^n$ con $\mathbf{0} \in D$ y $V(x)$ un candidato de función de Lyapunov. [7]

$$V(x) > 0 \quad \forall x \in D - \{0\} \quad (17)$$

$$\dot{V}(x) \leq 0 \quad \forall x \in D \Rightarrow x_{ss} \text{ es estable} \quad (18)$$

Si $\dot{V}(x)$ es menor que 0, entonces x_{ss} es localmente asintóticamente estable.

Estabilidad en sistemas LTI

En los sistemas LTI los puntos de equilibrio pueden ser:

- Globalmente asintóticamente estables (G.A.S):

$$\text{si } \text{Re}(\lambda) \leq 0 \quad \forall \lambda \in \sigma(A).$$

- Estables:

$$\text{si } \text{Re}(\lambda) < 0 \quad \forall \lambda \in \sigma(A) \text{ y } \text{Re}(\lambda).$$

- Inestables:

$$\text{si } \text{Re}(\lambda) > 0 \quad \forall \lambda \in \sigma(A).$$

La estabilidad del sistema está estrechamente relacionada con los eigenvalores de la matriz A . Por lo tanto, los polos del sistema LTI, los cuales caracterizan completamente su estabilidad, son los eigenvalores de la matriz A . [7]

Método indirecto de Lyapunov

Por medio del teorema de Hartman-Grobman[7], el cual garantiza que el comportamiento local del sistema dinámico no lineal es cualitativamente el mismo que el de su linealización en un punto de equilibrio hiperbólico, es posible determinar si un sistema dinámico no lineal es localmente asintóticamente estable o inestable. Esto se obtiene al evaluar la estabilidad de la linealización alrededor de tal punto de equilibrio. Cabe mencionar que este análisis tiene la limitante de no poder dar información si el sistema linealizado no es ni G.A.S. ni inestable.[7]

2. Python Dynamics (PyDy)

Es una plataforma que utiliza una variedad de herramientas científicas escritas en el lenguaje de programación Python para estudiar la dinámica de múltiples cuerpos y un conjunto de paquetes de software que ayudan a automatizar y mejorar el flujo de trabajo. El núcleo de este conjunto de herramientas es el paquete de mecánica SymPy que genera ecuaciones simbólicas de movimiento para sistemas multicuerpo complejos.[11]

Guía de programación

Para comenzar a trabajar con vectores, necesita crear un marco de referencia, ya que todos los vectores deben definirse con respecto a un marco de referencia.

```
N = ReferenceFrame( 'N' )
```

N es el nombre de la variable del que se almacena el marco de referencia llamado 'N'. Es importante tener en cuenta que N es un objeto y tiene propiedades y funciones asociadas. Tres de las propiedades son x , y y z . Estos son los vectores unitarios ortonormales asociados con el marco de referencia y son los bloques de construcción para crear vectores. Podrá crear un vector simplemente construyendo una combinación lineal de los vectores unitarios.

```
v = 1 * N.x + 2 * N.y + 3 * N.z
```

Los vectores no tienen que estar definidos con respecto a un solo marco de referencia. Podrá crear un nuevo marco de referencia y orientarlo con respecto al marco N que ya se ha creado. El método de orientación del nuevo marco para hacer una rotación simple a través de α sobre el eje N_x .

```
A = ReferenceFrame( 'A' )
alpha = symbols( 'alpha' )
A.orient( N, 'Axis', [alpha, N.x])
```

En los sistemas dinámicos, al menos parte de la orientación relativa de los marcos y vectores de referencia varía en el tiempo. El módulo de mecánica proporciona una forma de especificar cantidades que varían en el tiempo. Se define dos variables beta y d como variables que son funciones del tiempo.

```
beta, d = dynamicsymbols('beta_d')
```

Se calcula la velocidad angular del marco de referencia A con respecto a otros marcos de referencia:

```
B.ang_vel_in(N)
```

Esto nos permite ahora diferenciar el vector vec , con respecto al tiempo y un marco de referencia:

```
vecdot = vec.dt(N)
vecdot.express(N)
```

La función de símbolos dinámicos también permite almacenar las derivadas de variables variables en el tiempo

```
theta = dynamicsymbols('theta')
omega = dynamicsymbols('theta', 1)
```

También puede definir un punto en el origen de una rotación que se fija en N:

```
no = Point('no')
no.set_vel(N,0)
```

Se genera movimiento (velocidad y aceleración) a lo largo de cualquier eje de un marco de referencia:

```
p = Point('p')
x = dynamicsymbols('x')
r = x * A.y
p.set_pos(no, r)
p.set_vel(N, r.dt(N))
p.vel(N)
p.acc(N)
```

Se puede generar un cuerpo rígido idealizado. Este es esencialmente un objeto que contiene los diversos componentes que describen un cuerpo rígido: un nombre, masa, centro de masa, marco de referencia e inercia. Todos estos deben ser suministrados en la creación, pero se pueden cambiar después.

```
# RigidBody(name, masscenter, frame, mass, inertia)
m = Symbol('m')
A = ReferenceFrame('A')
P = Point('P')
I = outer(A.x, A.x)
inertia_tuple = (I, P)
B = RigidBody('B', P, A, m, inertia_tuple)
# Or you could change them afterwards
m2 = Symbol('m2')
B.mass = m2
```

Así como a los puntos también se les puede asignar velocidades y aceleraciones a los cuerpos rígidos, para lo cual se debe seguir el siguiente procedimiento:

```
M, v, r, omega = dynamicsymbols('M_v_r_omega')
N = ReferenceFrame('N')
b = ReferenceFrame('b')
b.set_ang_vel(N, omega * b.x)
P = Point('P')
P.set_vel(N, 1 * N.x)
I = outer(b.x, b.x)
B = RigidBody('B', P, b, M, (I, P))
```

3. Método de Kane

Bajo el método de Kane existen cinco ecuaciones básicas las cuales nos ayudaran a describir el mecanismo. Estas, son: restricciones holonómicas, restricciones no holonómicas, ecuaciones diferenciales cinemáticas, ecuaciones dinámicas y ecuaciones no holonómicas diferenciadas.

$$\begin{aligned}
 \mathbf{f}_h(q, t) &= 0 \\
 \mathbf{k}_{nh}(q, t)u + \mathbf{f}_{nh}(q, t) &= 0 \\
 \mathbf{k}_{k\dot{q}}(q, t)\dot{q} + \mathbf{k}_{ku}(q, t)u + \mathbf{f}_k(q, t) &= 0 \\
 \mathbf{k}_d(q, t)\dot{u} + \mathbf{f}_d(q, \dot{q}, u, t) &= 0 \\
 \mathbf{k}_{dnh}(q, t)\dot{u} + \mathbf{f}_{dnh}(q, \dot{q}, u, t) &= 0
 \end{aligned} \tag{19}$$

En este método se identifican que las variables de estados son q y \dot{q} , es decir, la posición y velocidad angular. De estas velocidades generalizadas se puede obtener las ecuaciones diferenciales cinemáticas al relacionarlas con u .

$$\{\dot{q}_1 : u_1, \dot{q}_2 : u_2, \dot{q}_3 : u_3, \dot{q}_4 : u_4\} \tag{20}$$

Para el planteamiento de las últimas ecuaciones es necesario obtener las fuerzas activas generalizadas(F_r) y las fuerzas inerciales generalizadas(F_r^*) del sistema. Las fuerzas F_r están definidas como producto punto de las velocidades parciales y las fuerzas activas y el producto punto de las velocidades angulares parciales con los torques activos. Luego, para cada junta P_j sometido a fuerzas aplicadas y para eslabón B_k sometido a pares aplicados:

$$(F_r)_{P_j} \triangleq \mathbf{v}_r^{P_j} \cdot \mathbf{R}_{P_j} \quad (F_r)_{B_k} \triangleq \omega_r^{B_k} \cdot \mathbf{T}_{B_k} \quad (r = 1, \dots, p) \tag{21}$$

$$F_r = \sum_{j=1}^{\mu} (F_r)_{P_j} + \sum_{k=1}^v (F_r)_{B_k} \quad (r = 1, \dots, p) \tag{22}$$

Las fuerzas de inercia generalizadas se obtendrán a partir del producto de punto de velocidades parciales y fuerzas de inercia y producto de punto de velocidades angulares par y pares de inercia. Luego, para cada junta P_j que posee masa y para cada eslabón B_k que posee inercia:

$$(F_r^*) p_j \triangleq \mathbf{v}_r^{P_j} \cdot \mathbf{R}_{P_j}^* \quad (F_r^*)_{B_k} \triangleq \omega_r^{B_k} \cdot \mathbf{T}_{B_k}^* \quad (r = 1, \dots, p) \quad (23)$$

$$F_r^* = \sum_{j=1}^{\mu} (F_r^*) p_j + \sum_{k=1}^v (F_r^*)_{B_k} \quad (r = 1, \dots, p) \quad (24)$$

Al sumar las fuerzas F_r y F_r^* el resultado es 0, teniendo esto en mente se pueden reescribir de la siguiente forma:

$$\mathbf{M}(q, t) \dot{u} = \mathbf{f}(q, \dot{q}, u, t) \quad (25)$$

donde,

$$\mathbf{M}(q, t) = \begin{bmatrix} \mathbf{k}_d(q, t) \\ \mathbf{k}_{dnh}(q, t) \end{bmatrix} \quad (26)$$

$$\mathbf{f}(q, \dot{q}, u, t) = \begin{bmatrix} -\mathbf{f}_d(q, \dot{q}, u, t) \\ -\mathbf{f}_{dnh}(q, \dot{q}, u, t) \end{bmatrix}$$

Al despejar para \dot{u} , se obtendrán las velocidades y aceleraciones para q_j de nuestro sistema. Es decir, que se lograría determinar las ecuaciones que describen la dinámica del mecanismo.

$$\dot{u} = \mathbf{M}(q, t)^{-1} \mathbf{f}(q, \dot{q}, u, t) \quad (27)$$

D. Secciones de Poincaré

Se considera un ciclo límite una solución $x(t) = p(t)$ de $\dot{x} = F(x, y)$ y $\dot{y} = G(x, y)$ de periodo $T > 0$, es decir, $p(t + T) = p(t)$. Para estudiar la dinámica cerca de dicho ciclo límite, se construye una sección de Poincaré, que es una superficie Π de dimensión $n - 1$ respecto del sistema, y que contiene un punto $x_p = p(t^*)$ en el ciclo límite que es transversal al flujo en x_p .

Se considera una solución $\Phi(x, t)$ en el espacio de fase n -dimensional y una hipersuperficie $\Pi = \{x \in \mathbb{R}^n : \pi(x) = 0\}$, de tal forma que corte transversalmente la trayectoria correspondiente a $\Phi(x, t)$, como se muestra en la Figura 23. La condición de transversalidad es que el vector normal $\pi_x(x_p)$ de Π en x_p tenga una componente distinta de cero en la dirección de $\Phi_t(x_0, t_0) = f(x_p)$. (Aquí el subíndice x o t significa diferenciación respecto de esa variable). Es decir, se requiere

$$\pi_x(x_p)f(x_p) \neq 0,$$

donde el subíndice x significa diferenciación parcial respecto de esa variable, es decir, que $\pi_x(x_p)$ es el vector normal de Π en $x = x_p$. [12]

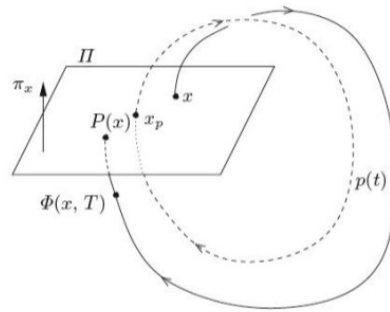


Figura 23: Sección de Poincaré. [12]

A continuación, se puede usar Φ para definir una sección de Poincaré, la cual está definida para x suficientemente cerca de x_p a través

$$P(x) = \Phi(x, \tau(x)) \tag{28}$$

donde $\tau(x)$ se define implícitamente como el tiempo más cercano a T para el cual $\pi(\Phi(x, \tau(x))) = 0$.

Uno de los beneficios de estudiar las secciones de Poincaré es que permiten reducir los sistemas de ecuaciones diferenciales de dimensiones n a un sistema discreto de dimensión $n - 1$. [12]

A. Modelo dinámico

Para el desarrollo de este módulo y alcanzar los objetivos planteados, el modelo dinámico del mecanismo de propulsión es de gran importancia. Por lo tanto, primero se determinará el modelo dinámico utilizando la plataforma de Python y su librería PyDy. Este modelo es imprescindible al momento de desarrollar y plantear el controlador para los mecanismos. Además, estas ecuaciones permiten observar la evolución del sistema a través del tiempo al ser perturbado por fuerzas externas. El método que se escogió para determinar estas ecuaciones es el método de Kane.

El modelo dinámico obtenido será migrado a lenguaje de MATLAB, donde será simulado y posteriormente empleado para el controlador por capas. Adicionalmente, en MATLAB, se extraerá información de posición y velocidad del extremo libre del remo y de las juntas.

Para su validación se procederá a extraer la misma información que se obtuvo en MATLAB utilizando el software de análisis cinemático de video Kinovea. En Kinovea se analizarán los videos del mecanismo impulsor en movimiento. Se graficarán los datos obtenidos de MATLAB y Kinovea para un análisis cualitativo. Si todas las trayectorias presentan el mismo comportamiento y se encuentran dentro de rango similares, el modelo dinámico será válido para su uso en el desarrollo de los controladores.

B. Control de velocidad de mecanismos

Se planteará un controlador utilizando teorías de control clásico, para mantener una velocidad constante en el motor. Para ello será necesario determinar la planta del sistema, que en cuyo caso es la función de transferencia que describe al motor. Con ello se podrá calcular las diversas constantes que utilice el controlador para estabilizar el sistema en el punto deseado.

Una vez determinadas las ecuaciones que describen la dinámica del mecanismo de propulsión del BART, se tendrá la opción de plantear controlador que dé una velocidad lineal constante en la salida. De ser así, dicho controlador daría las velocidades a las que debe girar el motor.

Se generarán gráficos de velocidad en la salida de la etapa de potencia del mecanismo impulsor del BART. Para ello, se utilizará Kinovea con el fin de obtener los datos necesarios para los gráficos. Se realizarán varias pruebas a diferentes velocidades con el propósito de corroborar el correcto funcionamiento del controlador. Por lo tanto, los gráficos deben de mostrar la misma magnitud que la señal de referencia del controlador.

C. Control de desplazamiento

Se planteará un modelo unicycle del BART con el cual se planteará un controlador cinemático de cuerpo completo. Este controlador puede ser determinado utilizando métodos de control moderno o clásico. Este controlador o conjunto de controladores dará tres velocidades de las cuales dos son velocidades lineales y una angular. Estas velocidades se encuentran representadas ecuaciones encontradas en la fase II. Con estas ecuaciones se podrá despejar para cada una de las velocidades lineales de los mecanismos de propulsión. [3]

$$\begin{bmatrix} v_x \\ v_y \\ w \end{bmatrix} = \begin{bmatrix} -\frac{v_{LD}-v_{LU}+v_{RD}-v_{RU}}{4} \sin(\alpha + \beta) \\ -\frac{v_{LD}+v_{LU}-v_{RD}-v_{RU}}{4} \cos(\alpha + \beta) \\ \frac{v_{LD}-v_{LU}-v_{RD}+v_{RU}}{4l} \cos(\beta) \end{bmatrix} \quad (29)$$

Debido a que el mecanismo del BART es incapaz de producir velocidades negativas[1] y el sistema se encuentra sobreactuado, se plantea cuatro condiciones iniciales a partir de las cuales se buscará despejar cada v_j . Estas condiciones serán instantes en donde el BART se desplaza sobre el eje $+X$, $-X$, $+Y$ y $-Y$. En cada una de estas condiciones se anulan dos velocidades que actúan sobre el robot, como resultado de las características mencionadas.

Una vez que se tengan despejadas las velocidades para cada caso. Se procederá a sumar cada ecuación que contenga la misma v_j . Por tanto, será posible relacionar la salida del controlador de trayectoria con cada uno de los motores del BART.

Para su validación, se compararán las trayectorias descritas en las simulaciones realizadas en MATLAB y la trayectoria descrita por el BART. Para obtener la trayectoria del BART se utilizará Kinovea para el análisis de su trayectoria. Por otro lado, también se planteará el uso de comunicación RF BART-PC para obtener los datos de posición de BART y enviar la información del controlador al BART. Por lo tanto, las trayectorias descritas por el BART y la simulación deberán de ser parecidas para la validación del controlador.

D. Diseño de placa electrónica

Se diseñará una placa electrónica en donde se encuentren todos los módulos del BART. Estos módulos están divididos por alimentación, controladores, drivers de motores y comunicaciones. Las dimensiones de esta se verán limitadas por el espacio y forma interna del casco del BART.

Se utilizará el software de desarrollo Altium, en donde se podrá simular las corrientes y voltajes de cada pista que poseerá la placa. Con el fin de determinar las áreas transversales de cada pista para soportar la corriente que pasa a través de ellos. También se podrá generar los diagramas de cada PCB para su futura fabricación. Una vez que se tenga cortada la placa electrónica se procederá a realizar la soldadura de los componentes electrónicos y la instalación de módulos que ya estén fabricados.

Para su validación se realizarán pruebas para cada uno de los módulos. Para el módulo de comunicación se enviará información y se leerá la información recibida en el microcontrolador. De ser la información recibida idéntica a la enviada, se validará el módulo de comunicación. Para los drivers de motores, se configurará el microcontrolador para que un PWM haga girar los motores. Si los motores giran a la velocidad establecida se validarán los drivers de motores. En el caso de la alimentación se verificará que los voltajes suministrado para cada uno de los módulos sea el establecido. Además, se realizarán pruebas de continuidad para verificar que los pistas este correctamente maquinados.

E. Desarrollo de simulaciones

Con el propósito de mostrar de una forma clara y sencilla el comportamiento de los controladores o la dinámica del mecanismo de propulsión, se hará uso del software de programación MATLAB, en el que se definirán cada una de las ecuaciones de estado de la dinámica del sistema o del controlador, según sea el caso. Para los casos que no hagan uso de ningún controlador se utilizara el método numérico Runge-Kutta de orden cuatro[7], para estimar los distintos valores del sistema.

Dinámica de mecanismo impulsor

Durante las fases anteriores se modelaron los mecanismos como una rueda de velocidad constante, sobre la cual se plantearon controladores PID. Sin embargo, no hubo un modelado dinámico formal. Al plantear el modelo dinámico se espera que brinde una relación clara entre el mecanismo de cuerpos rígidos y la aproximación realizada como una rueda de velocidad constante. Para poder determinar el modelo dinámico del mecanismo de propulsión bioinspirado del BART, se empleó el método que el método de Kane.

A pesar que las ecuaciones obtenidas utilizando los diversos enfoques (Lagrange, Newton-Euler, etc.) son equivalentes en el sentido de que generan los mismos resultados numéricos, el método de Kane tiene algunas ventajas únicas, incorpora restricciones de configuración, también usa velocidades generalizadas lo que permite incrustar restricciones de movimiento. El uso de velocidades generalizadas en el método de Kane permite la selección de un conjunto independiente de variables de movimiento para sistemas restringidos no holonómicos que permite obtener un modelo en forma de ecuaciones diferenciales de primer orden.

A. Mecanismo

El mecanismo cuenta con 5 juntas, 5 eslabones y uno de ellos con 3 juntas. Para su análisis, el eslabón con 3 juntas se dividirá en dos eslabones de dos juntas con el fin de facilitar la programación del sistema de ecuaciones. Ya que se utilizó coordenadas generalizadas, el eje X ira alineado con el eslabón. Por lo tanto, q_1 se medió respecto al marco de referencia $\{N\}$, q_2 respecto al marco $\{A\}$, q_3 con el marco $\{B\}$ y q_4 con el marco $\{C\}$. Tal como se muestra en la Figura 24.

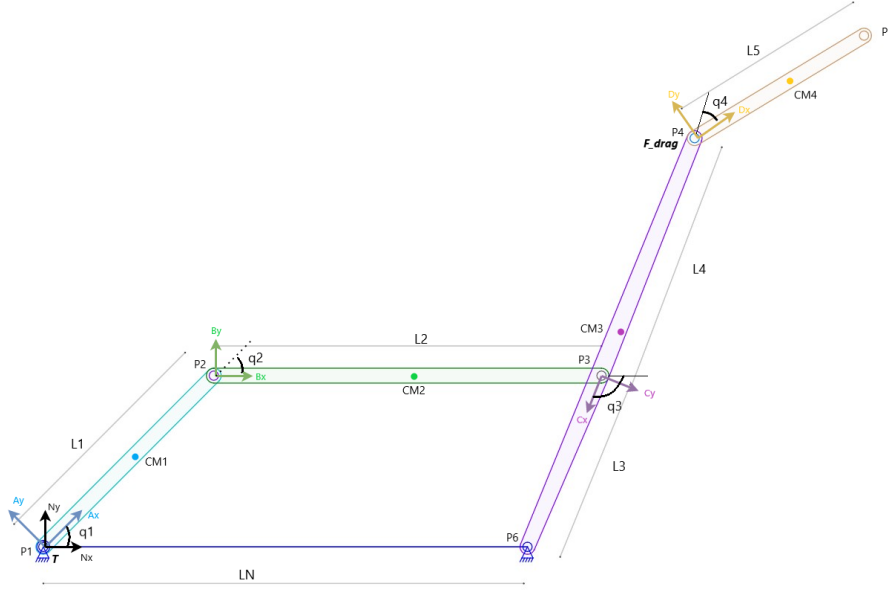


Figura 24: Mecanismo impulsor.

Al analizar el mecanismo de la Figura 24 para obtener las ecuaciones de restricción de movimiento para el eje X y Y se consigue lo siguiente:

- Para el eje X:

$$L_1 \cos(q_1) + L_2 \cos(q_1 - q_2) + L_3 \cos(-q_1 + q_2 + q_3) - L_N \quad (30)$$

- Para el eje Y:

$$L_1 \sin(q_1) + L_2 \sin(q_1 - q_2) - L_3 \sin(-q_1 + q_2 + q_3) \quad (31)$$

Si se derivan las ecuaciones de restricción mecánica respecto al tiempo, se obtiene como resultado dos ecuaciones de restricciones holonómicas. Las velocidades generalizadas se represento como en la ecuación 19.

$$-L_1(u_1) \sin(q_1) - L_2(u_1 - u_2) \sin(q_1 - q_2) - L_3(-u_1 + u_2 + u_3) \sin(-q_1 + q_2 + q_3) \quad (32)$$

$$L_1(u_1) \cos(q_1) + L_2(u_1 - u_2) \cos(q_1 - q_2) - L_3(-u_1 + u_2 + u_3) \cos(-q_1 + q_2 + q_3) \quad (33)$$

Al observar el mecanismo y conocer que la entrada de este se encuentra en q_1 se puede determinar que los ángulos y velocidades independientes son q_1 , q_4 , u_1 y u_4 y los ángulos y velocidades dependientes son q_2 , q_3 , u_2 y u_3 . El ángulo q_4 es independiente ya que el eslabón L_5 puede rotar aun cuando la cadena cerrada de eslabones esta estática.

En el punto P_4 de la Figura 24 se tiene una fuerza de arrastre, debido a que el mecanismo se mueve dentro de un fluido. Esta fuerza F_{drag} permite que el eslabón se mueva en dirección contraria del eslabón 3. El eslabón de remo posee topes mecánicos que ayudan a variar esta fuerza de arrastre dependiendo en la dirección a la que se mueva el balancín. Si el balancín se mueve hacia la izquierda, esta fuerza será mayor ya que hay una mayor área de contacto

con el fluido. En el caso de que el balancín se mueva hacia la derecha, el remo se retrae y tiene una menor área de contacto con el fluido. Esta característica del mecanismo permite que el BART se mueva en el agua.

A pesar de que existe una fuerza de empuje en ambas direcciones en una revolución, la fuerza resultante empuja el BART hacia adelante. En consecuencia, un solo mecanismo es incapaz de poder movilizar al BART hacia adelante y atrás cambiando el sentido de giro de su eslabón de entrada.

Se calcularon las inercias I_{yy} y I_{zz} que poseen cada uno de los eslabones 1, 2 y 3. En este caso particular las inercias I_{yy} y I_{zz} son iguales.

$$I_{yy} = I_{zz} = \frac{mL^2}{12} \quad (34)$$

En el caso particular del remo se uso el teorema de ejes paralelos para el cálculo de sus inercias.

$$I_{yy} = I_{zz} = \frac{mL^2}{12} + mD^2 \quad (35)$$

Es necesario calcular el centro de masa de cada uno de los eslabones antes de utilizar el método de Kane. Estos se encuentran a una distancia de $L/2$, es decir en el centro del eslabón. Para el caso del eslabón de salida, donde se consideró previamente como dos eslabones en vez de un único eslabón. Se requirió de ubicar un único centro de masa

$$CM_3 = \frac{L_3 + L_4}{2} \quad (36)$$

Es importante resaltar que cada centro de masa es referenciado al marco de referencia de su eslabón. Así mismo, se debe de asignar una velocidad y aceleración a los centros de masas. Esto represento la velocidad y aceleración de los eslabones al resolver el sistema de ecuaciones del método de Kane. Todos estos cálculos de inercia y centro de masa son necesarios ya que los eslabones se planteo como cuerpos rígidos y no como partículas. Al modelar el sistema como un conjunto de cuerpos rígidos se espera tener un mejor modelo de la dinámica del sistema.

Debido a la complejidad que presento las ecuaciones del método de Kane, se escribió un código en Python utilizando la librería de Python Dynamics. En dicho programa se plantearán todas las ecuaciones expuestas con anterioridad buscando tener como resultado final la dinamica del mecanismo u según la ecuación 27.

B. Python

La estructura del programa en python siguió la estructura del siguiente pseudocódigo:

- Definición de variables tipo *dynamicsymbols* para q_n, u_n y fuerzas
- Declarar de marcos de referencia para cada eslabón
- Crear *Points* en las coordenadas de cada junta.
- Colocar los centros de masa (CM) para cada eslabón.
- Declarar la *ecuación* 19 de restricción de movimiento respecto a la bancada.
- Introducir una velocidad generalizada para cada coordenada generalizada (tanto para coordenadas generalizadas independientes como dependientes)
- Colocar las velocidades de las juntas y CM en términos de las velocidades generalizadas.
- Declarar las *ecuaciones* 23 y 24 de restricción de movimiento respecto a la bancada.
- Colocar valor de 0 a la velocidad de la junta que une bancada y eslabón de salida.
- Asegúrese de que todas las aceleraciones relevantes para formar las ecuaciones de Kane no tengan q 's.
- Calcular la inercia I_{yy} y I_{zz} para cada eslabón.
- Declarar estructura de datos *RigidBody* para los eslabones.
- Introducir parámetros a *KaneMethod* (frame, q_{ind} , u_{ind} , ecuaciones cinemáticas diferenciales, $q_{dependent}$, restricciones de configuración, $u_{dependent}$, restricciones de velocidad, $u_{auxiliary}$)
- Declarar las fuerzas del sistema.
- Introducir los cuerpos rígidos y cargas a las ecuaciones de Kane utilizando *kanes_ - equations*
- Obtener \dot{u} mediante la ecuación 31.
- Exportar ecuaciones de \dot{u} a MATLAB.

Para unir los eslabones en las animaciones, se debe pasar cada junta del marco de referencia de su eslabón al marco de referencia global $\{N\}$, por lo que su posición estará dada como $P_n = (x, y)$

$$\begin{aligned} P_2 &= (L_1 \cos(q_1), L_1 \sin(q_1)), \\ P_3 &= P_2 + (L_2 \cos(q_1 - q_2), L_2 \sin(q_1 - q_2)), \\ P_4 &= P_3 + (-L_4 \cos(-q_1 + q_2 + q_3), L_4 \sin(-q_1 + q_2 + q_3)), \\ P_5 &= P_4 + (L_5 \sin(-q_1 + q_2 + q_3 + q_4), L_5 \cos(-q_1 + q_2 + q_3 + q_4)), \\ P_6 &= (L_N, 0). \end{aligned} \tag{37}$$

C. Resultados

Para la animación del mecanismo se plantearon cuatro segmentos de rectas, donde su punto inicial y final se conectaron mediante las ecuaciones 37. El eslabón de entrada está representado de color azul, el acoplador es de color rojo, eslabón de salida de color amarillo y el remo de color morado. Los eslabones poseen las dimensiones reales del mecanismo.

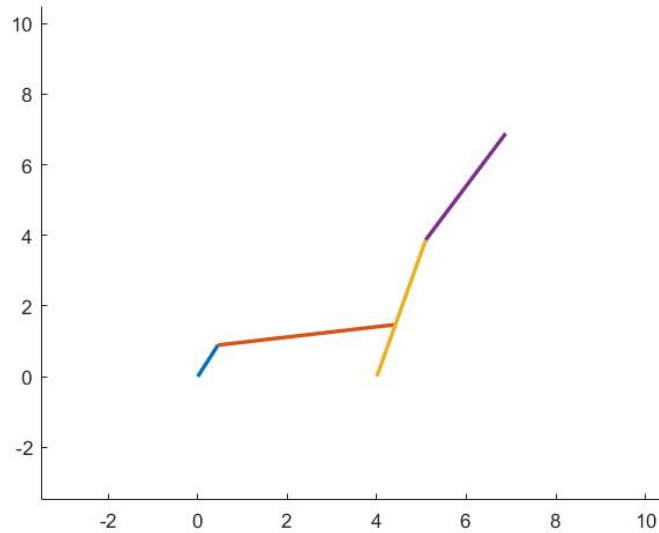


Figura 25: Animación en MATLAB.

Se realizaron dos simulaciones del sistema dinámico. La primera simulación se realizó con un torque constante de $0.5 \text{ kg} \cdot \text{cm}$. La segunda simulación utilizó un controlador PD para la velocidad del eslabón de entrada, colocando como velocidad de referencia $\frac{\pi}{2}$.

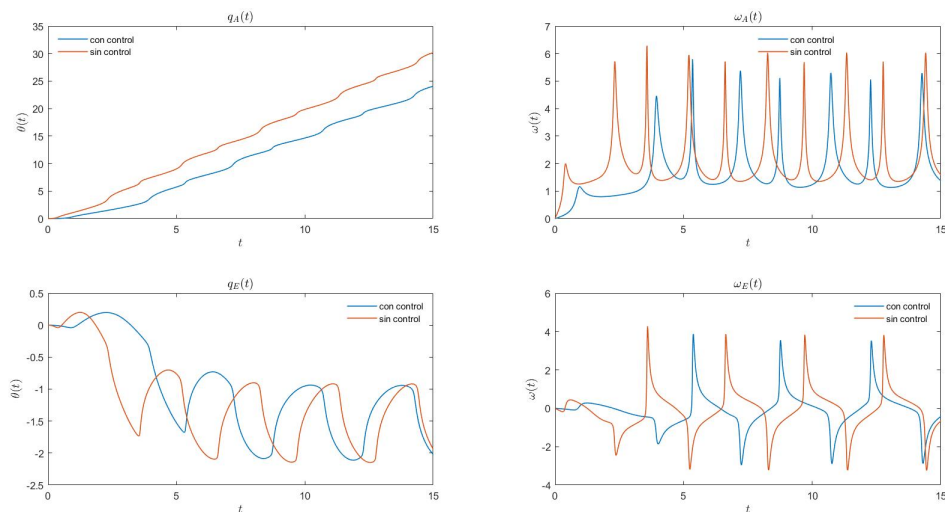


Figura 26: Posición y velocidad angular de juntas independientes.

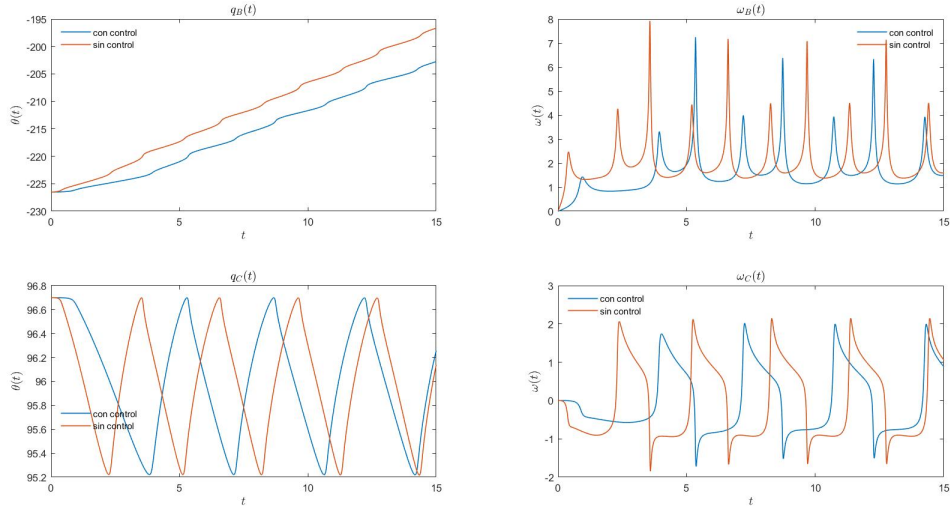
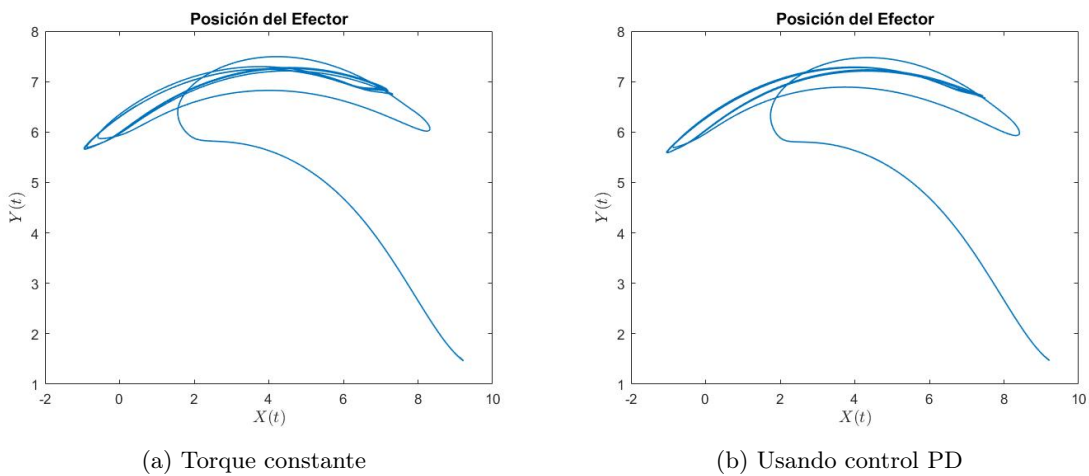


Figura 27: Posición y velocidad angular de juntas dependientes.

Como se observa en las gráficas de q_1 , la magnitud de los picos de velocidad se vio reducidos al utilizar el controlador PD entre un 7.69% a 22.82%. Estos picos de velocidad que poseen ambas simulaciones se debio a la inercia que lleva el eslabón del remo al llegar a ambos extremos de su trayectoria. Esta energía se transfiere al resto del mecanismo y produce un impulso extra antes de comenzar la carrera de ida o de retorno.

En las velocidades angulares de las cuatro juntas y en el desplazamiento angular de las juntas q_3 y q_4 se pudieron apreciar ciclos periódicos. Esto es un buen indicador de usar un modelo de rueda con velocidad constante. Para reforzar este argumento se tiene que observar la trayectoria del punto final del remo. Al utilizar control se ve una trayectoria periódica más definida que al momento de usar un torque constante a la entrada.



(a) Torque constante

(b) Usando control PD

Figura 28: Trayectoria del efector final.

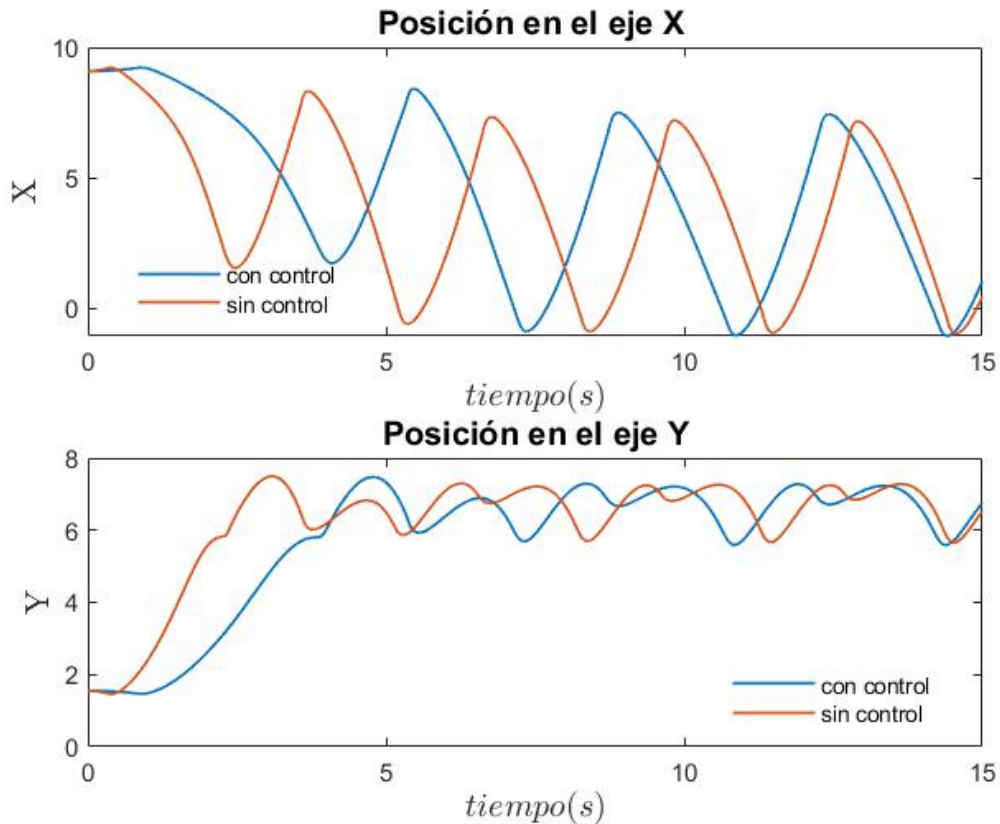


Figura 29: Trayectoria del efector en X y Y.

El sistema al llegar a un estado estacionario se mantiene oscilando en la trayectoria descrita por el efector final. Esto llevó a plantear una evaluación de estabilidad del modelo, con el fin de determinar cómo ayuda el control dentro del sistema. Al mantenerse oscilando en una trayectoria, la forma de evaluar estabilidad no puede ser mediante el uso de polo o la noción de un punto de equilibrio, sino hay que utilizar algún método para evaluar la estabilidad de un Ciclo límite.

Este mecanismo es complejo debido a las restricciones mecánicas del mecanismo de cuatro barras, los topes mecánicos que existen en la junta P_4 , el modelo de doble péndulo que forma el eslabón de salida junto con el remo y las dos entradas de potencia del mecanismo, siendo esta la del motor en la junta q_1 y la fuerza de lastre en la junta q_4 . Se evaluó el Ciclo límite mediante el método numérico de Poincaré. Se aplicó el método de Poincaré para ambos casos de simulación.

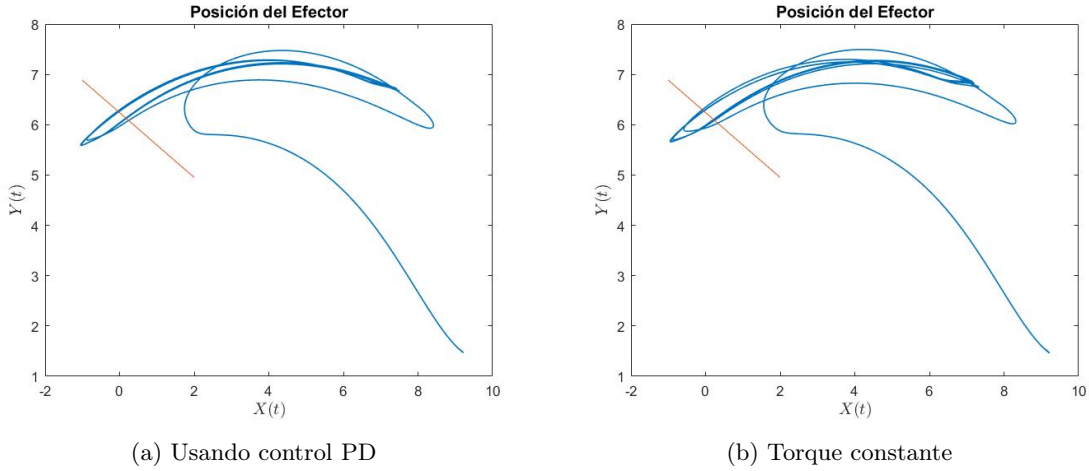


Figura 30: Corte de línea de Poincare.

Para ambos casos de simulación las trayectorias que cruzaron la línea S eventualmente retornaron a S . Se puede reducir el análisis de estabilidad para un ciclo límite en las coordenadas originales al análisis de estabilidad de un punto fijo en el mapa discreto. Para poder afirmar que y_p^* o x_p^* es un punto fijo estable del mapa P , debe de existir una órbita periódica única $x(t)^*$ que pase por x_p^* que es un Ciclo límite. Este comportamiento se observa en las Figuras 32 y 31.

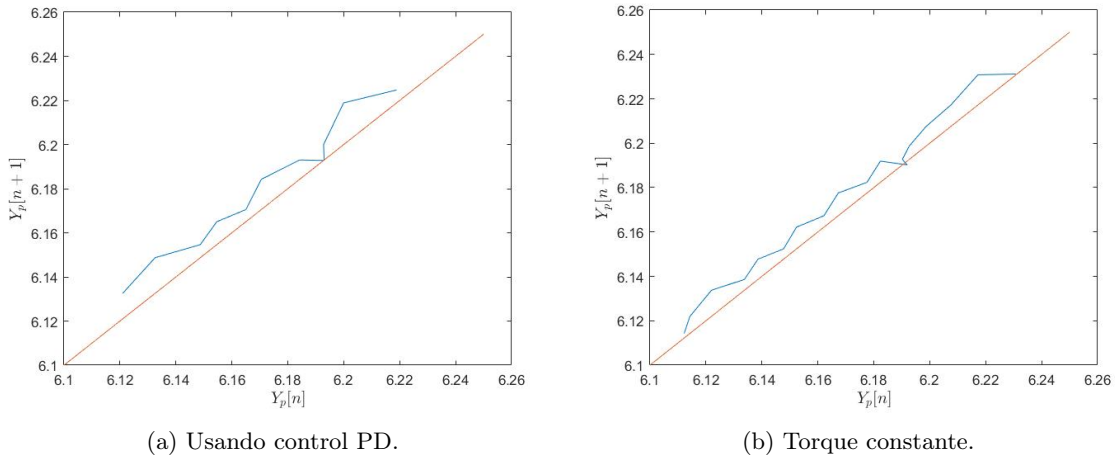


Figura 31: Mapa de retorno de Poincare con coordenadas en Y.

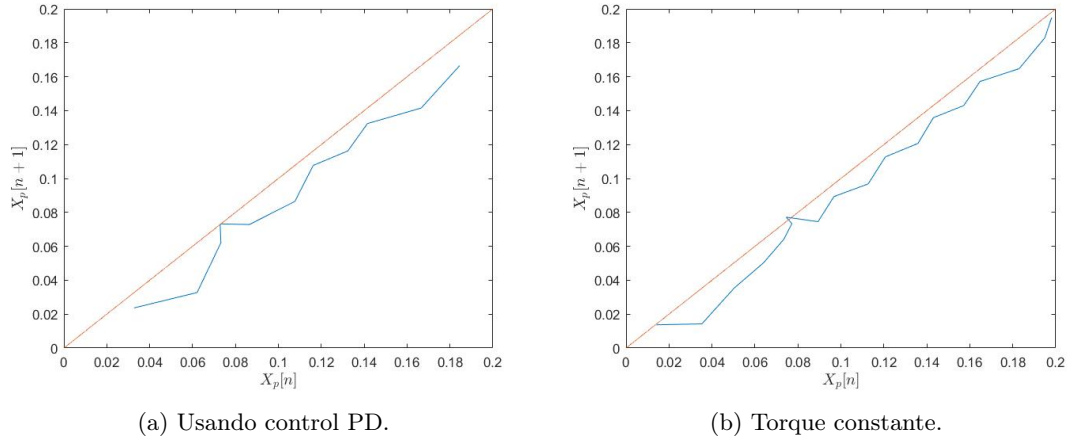


Figura 32: Mapa de retorno de Poincare con coordenadas en X.

Los puntos fijos estables para la simulación con control PD fueron $y_p = 6.193$ y $X_p = 0.0732$. Para el caso de torque constante los puntos fijos estables fueron $x_p = 0.0768$ y $y_p = 6.190$. Debido al orden de magnitud de la diferencia que existe del punto fijo estable de ambas simulaciones, se pudo concluir que era el mismo.

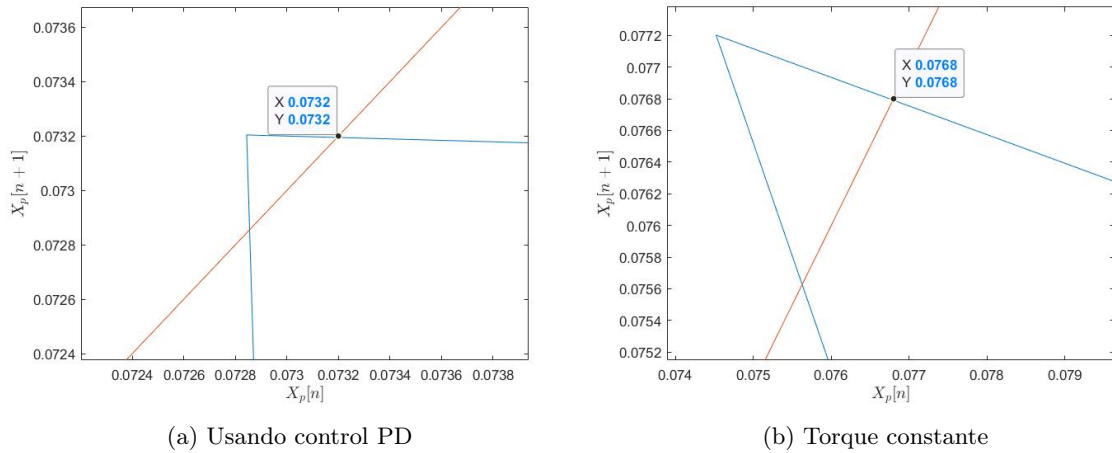
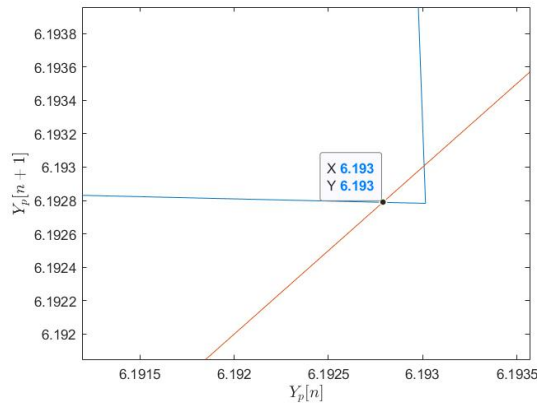
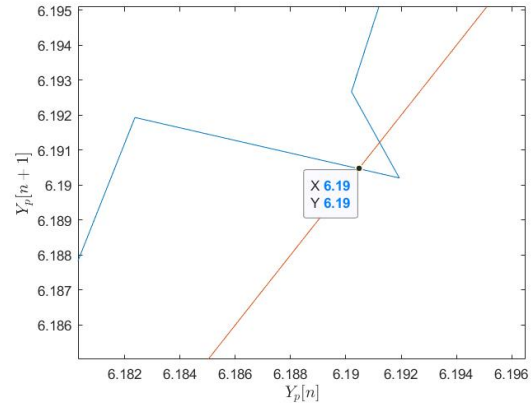


Figura 33: Punto fijo estable X.



(a) Usando control PD



(b) Torque constante

Figura 34: Punto fijo estable en Y.

Para validar el comportamiento del sistema dinámico, se compararon con mediciones reales tomadas del mecanismo impulsor del BART. Para ello se utilizó un controlador PD en el motor con una referencia de 30 rpm, en términos de velocidad angular $\frac{\pi}{2} rad/s$. Debido a la pandemia las pruebas no pudieron realizarse con equipo de laboratorio, por lo que se utilizó un recipiente plástico el cual fue llenado con agua hasta cubrir por completo el mecanismo. La cámara se colocó a una altura de 40 cm sobre el nivel del suelo y a una distancia horizontal de 10 cm del borde más cercano del recipiente. Además, se colocó una regla en el extremo opuesto al BART para una posterior calibración en KINOVEA y se pintó de rojo el extremo del remo para realizar un análisis de seguimiento de trayectoria.

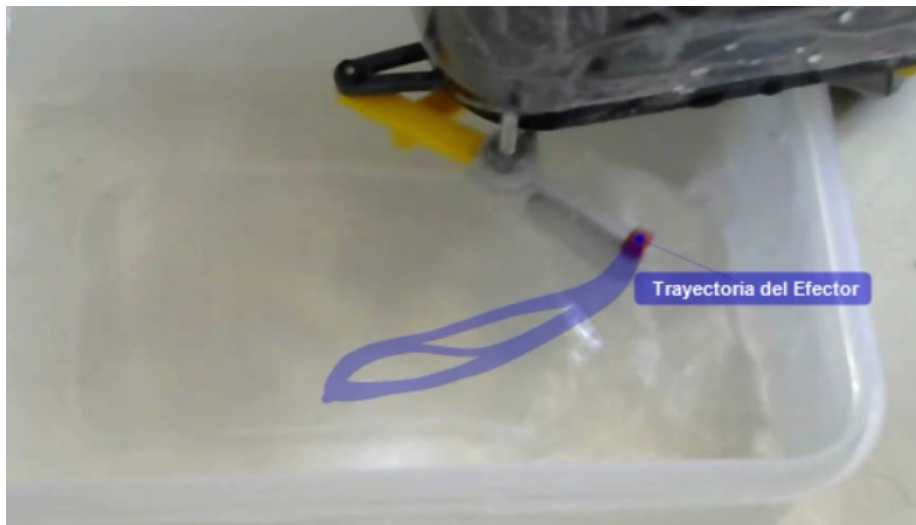


Figura 35: Análisis de trayectoria.

Una vez se obtuvieron los puntos de la trayectoria descrita por el mecanismo del BART se analizará su estabilidad mediante Poincaré, como se hizo anteriormente con las estabilidades, con el fin de encontrar el Ciclo límite de mecanismo impulsor.

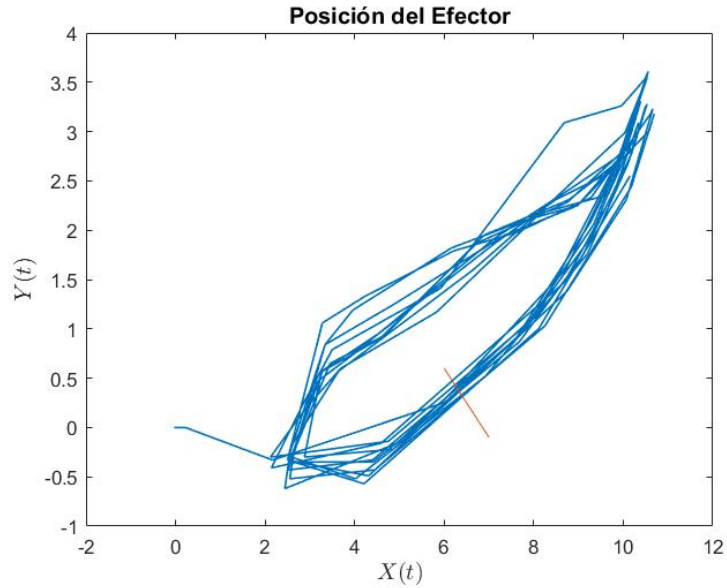


Figura 36: Trayectoria experimental en MATLAB.

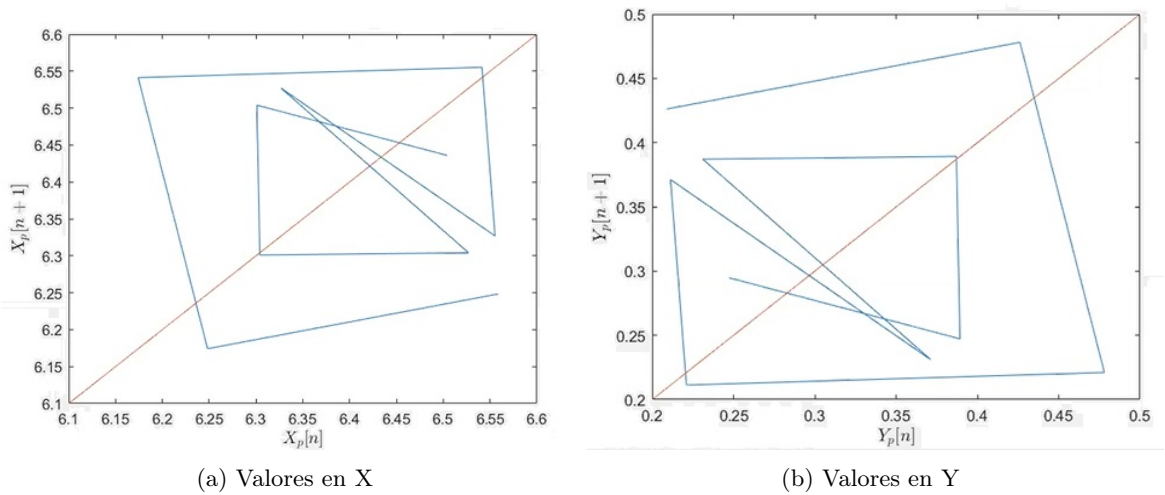


Figura 37: Mapa de retorno de Poincaré.

Debido a la velocidad de la mejor cámara que se pudo obtener, que fue de 30 fps, se decidió realizar una interpolación con polinomios de tercer grado y suavizar para tener una mejor visualización del comportamiento del sistema. Para el mapa de Poincaré se procesó con un filtro pasa bajas (promediador móvil), con el fin de reducir errores que existen en el mundo real y así poder determinar el intervalo en el cual estaba comprendido el punto fijo estable.

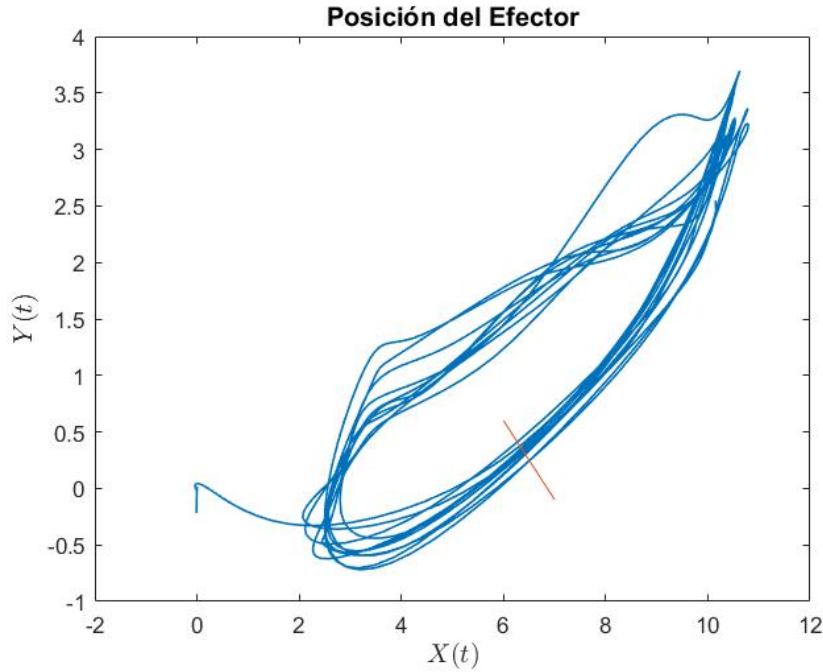


Figura 38: Trayectoria suavizada.

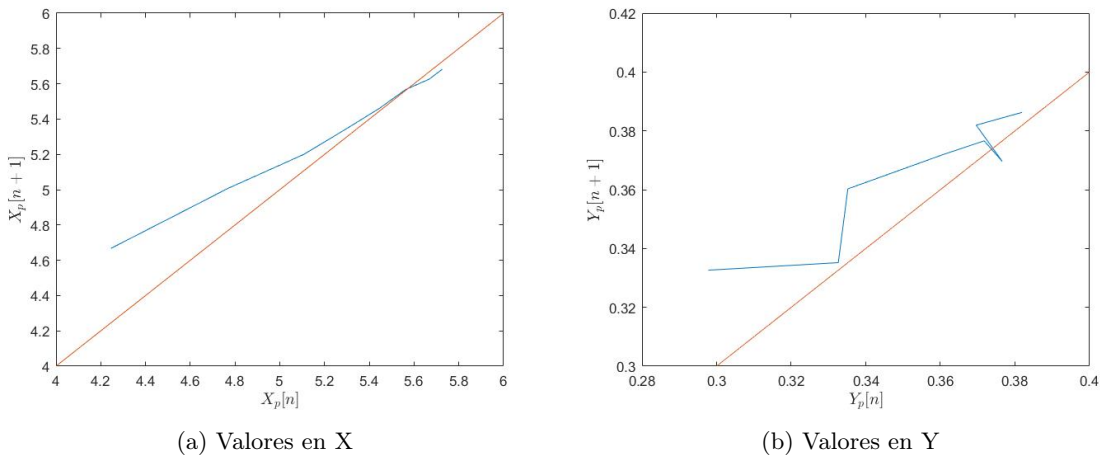
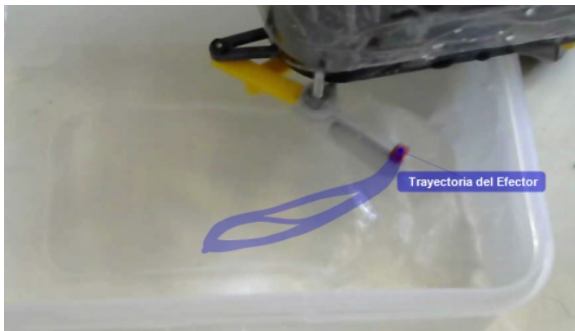


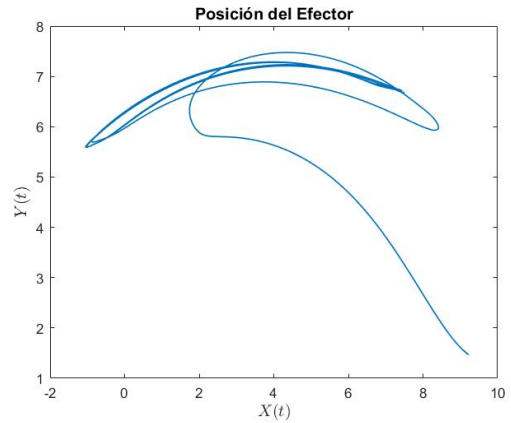
Figura 39: Mapa de retorno de Poincaré.

Los puntos fijos estables para la trayectoria experimental del mecanismo son $y_p = 5.57$ y $x_p = 0.373$. Para validar el modelo dinámico se comparó de forma cualitativa la trayectoria del mecanismo del BART versus la trayectoria simulada con control de PD.

Al observar ambas graficas de trayectoria se pudo apreciar que ambas presentan la forma de lágrima que caracteriza al efector final. En el modelo dinámico al igual que en el experimental se cuenta con una sección donde se describe una elipse y después se aplanan completamente mientras continúa describiendo un arco. Por lo tanto, el modelo dinámico encontrado fue válido para el mecanismo impulsor del BART.



(a) Trayectoria de BART



(b) Trayectoria de MATLAB

Figura 40: Mapa de retorno de Poincaré.

D. Fuerza ejercida por el mecanismo

Como punto de partida se analizó el ciclo de trabajo del mecanismo. Esto proveyó información de que porcentaje por ciclo está en fase de empuje(azul), transición(morado y rojo) o retroceso(amarillo). En la Figura 41 se observa la trayectoria segmentada en cada uno de los estados. En el Cuadro 1 se muestra que en el estado de empuje corresponde a un 52 % del tiempo, por lo que se puede hacer la analogía del efector final y el agua como un mecanismo de retorno rápido.

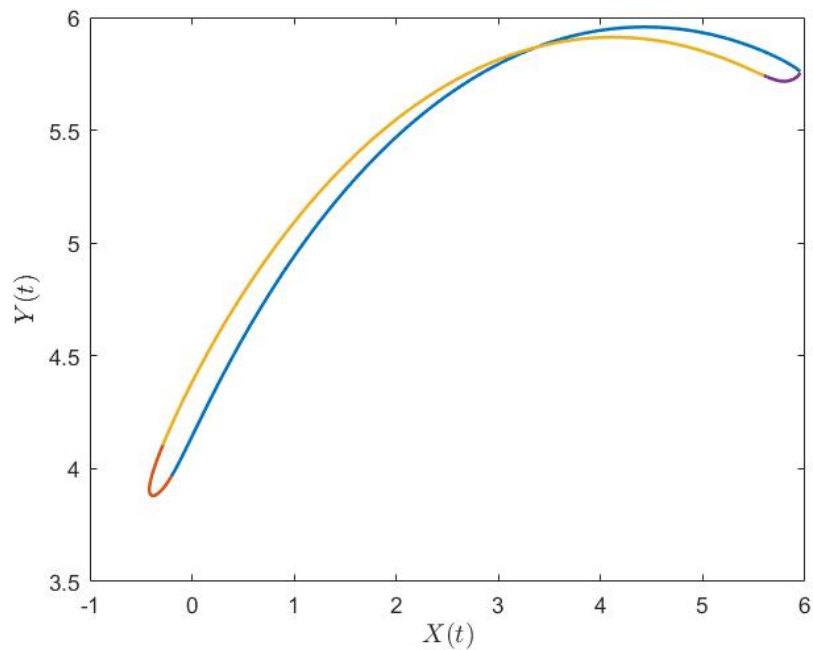


Figura 41: Ciclo de trabajo.

Estado	Ciclo de trabajo
Empuje	52 %
Transición	14 %
Retroceso	34 %

Cuadro 1: Ciclo de trabajo del mecanismo.

Durante la primera fase se observó que el mecanismo producía una fuerza constante sobre el BART, lo cual fue validado experimentalmente. Utilizando mapas de Poincaré se le dió fundamento teórico a esta conclusión. Para ello se graficó el ángulo q_E contra la velocidad angular ω_E

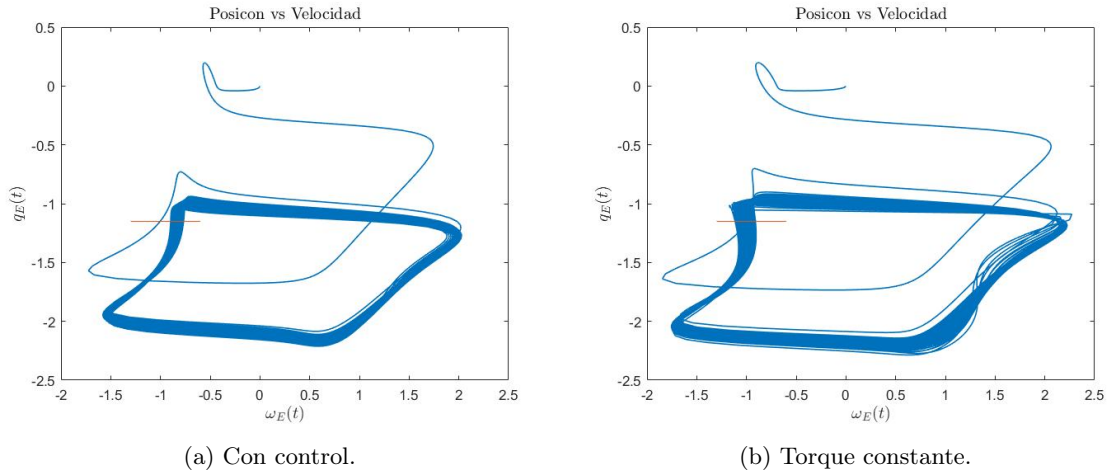
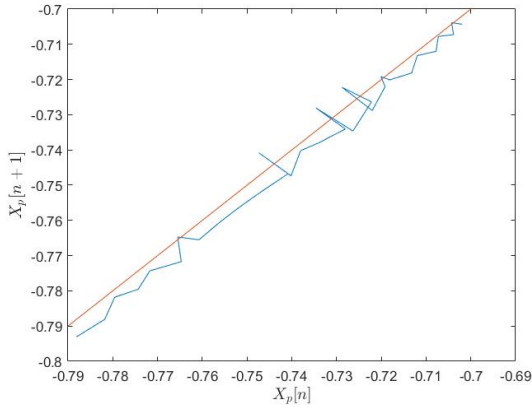
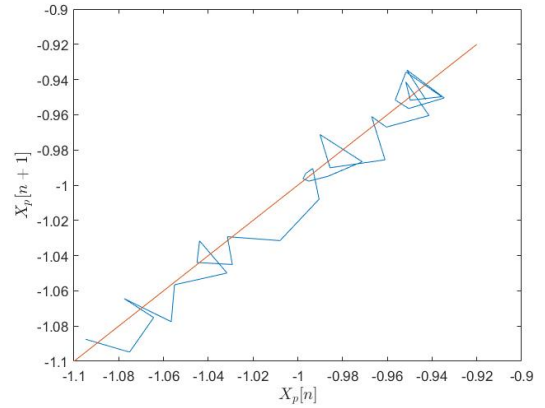


Figura 42: Trayectoria de posición vs velocidad angular en la junta E.

Como se muestra en la Figura 42 la línea de Poincaré se posiciono en el ángulo -1.15 rad , el ángulo donde se alcanza la máxima velocidad angular. Debido a la parte caótica que presenta el mecanismo se realizaron dos simulaciones con duración de 150 y 200 segundos para cada uno de los casos. Como se observa en las Figura 43 existen múltiples puntos fijos estables para la trayectoria, como consecuencia se encontró la velocidad angular limite mediante la media de todos los puntos que atravesaron la línea de Poincaré.

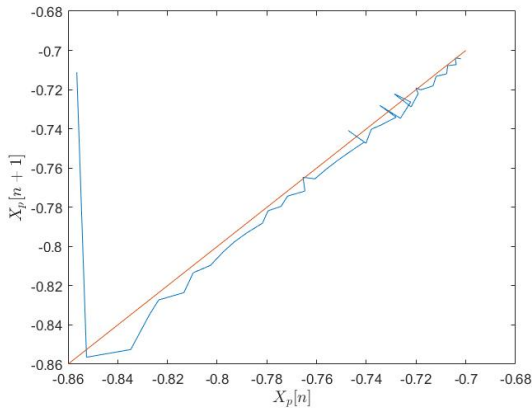


(a) Con control.

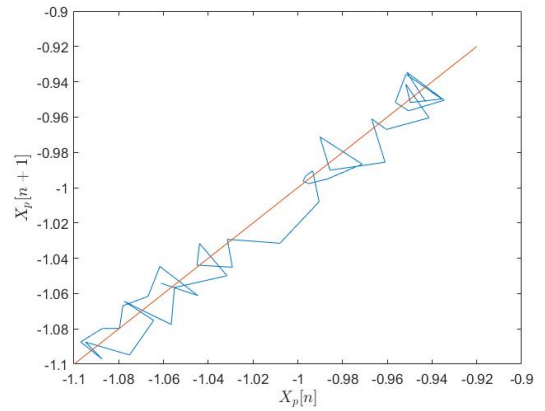


(b) Torque constante.

Figura 43: Simulación de 150 segundos.



(a) Con control.



(b) Torque constante.

Figura 44: Simulación de 200 segundos.

	Tiempo de simulación (s)	Media (rad/s)	Desviación estándar
Con control	200	-0.7576	0.048
	150	-0.7493	0.026
Torque constante	200	-1.015	0.063
	150	-0.9989	0.058

Cuadro 2: Velocidad angular límite.

Como se muestra en el Cuadro 2 al utilizar control se obtuvo una menor desviación estándar. La diferencia de velocidad límite al utilizar control fue de 0.0083 y para el caso sin control fue de 0.016. Esto indicó que el sistema es estable fue por sí mismo, pero al usar control mejoro su rendimiento. Teniendo la velocidad angular límite se pudo calcular la fuerza de arrastre. Esta fuerza se calculó para tres puntos en específico. Tomando como referencia la Figura 41 los puntos para calcular la fuerza estuvieron al inicio, final y a la mitad del segmento azul.

El coeficiente de arrastre para el efector final fue de 1.05 ya que es una pieza prismática. El área de contacto fue de 560 mm^2 o $5.6 \times 10^{-4} \text{ m}^2$. La densidad del fluido, que en este caso fue agua, será de 1000 kg/m^3 . Utilizando el método anterior para la velocidad angular límite, encontró la velocidad para el punto inicial y final siendo estas de -0.7421 rad/s y -0.7572 rad/s respectivamente. Utilizando la ecuación 6 se obtuvo los siguientes resultados,

Punto	Velocidad (rad/s)	Fuerza ($\frac{mmg}{s^2}$)	Distancia a punto de anclaje (mm)	Angulo entre fuerza y distancia (rad)	Torque ($\frac{mm^2g}{s^2}$)
Inicio	-0.7572	51.6	57.4	1.18	2738.54
Medio	-0.7576	51.7	53.1	1.33	2666.06
Final	-0.7421	50.5	56.2	1.12	2578.79

Cuadro 3: Fuerza y torque producida por el efector final.

Como se observa en el Cuadro 3 tanto la fuerza como el torque son aproximadamente constante a lo largo de la trayectoria de empuje. A pesar de que la fuerza fue relativamente pequeña, al estar alejada del punto de pivote el torque es dos órdenes de magnitud más grande que la misma. Esto indicó que al moverse existe un mayor desplazamiento angular que lineal. Lo que se observó experimentalmente durante la fase I (Figura 7). Esto es, que el torque que genera el mecanismo se ve contrarrestado por el torque de arrastre que genera el cuerpo del BART en el agua, por lo que se mantiene en equilibrio angular y por ende el BART gira entorno a su centro geométrico a una velocidad angular aproximadamente constante. Esto implica tener una aceleración angular aproximadamente igual a cero, lleva a un torque neto igual a cero. Esto validó el uso de un modelo unicycle de robot móvil para la cinemática del BART.

Control por capas

Se propuso un control por capas basado en tres capas principales como se muestra en la Figura 45. La primera capa fue de sensado y análisis del entorno. En esta capa se consideran dos tipos de sensado dependiendo el tamaño del entorno donde se encuentre el BART. Para el caso de un entorno pequeño se puede utilizar una cámara la cual tenga visión completa del ambiente donde se encuentra el BART, la información de la cámara puede ser procesada por algoritmos de visión de computadora como visual servoing[13]. Para el caso de un entorno de mediano y gran tamaño se consideró el uso de una IMU en conjunto con un módulo de GPS donde puede plantearse un problema de estimación de pose. La información de estos dos módulos podría combinarse para conocer la posición exacta del BART, además se deberá de agregar un sistema LIDAR o una cámara para conocer la posición de los objetos que lo rodean.

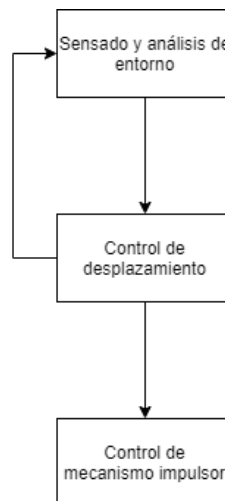


Figura 45: Control por capas.

La primera capa presentó como subcapa la planificación de trayectoria hacia la meta en caso de que el objetivo sea llegar a una posición determinada en el espacio. En el caso contrario donde se esté explorando se deberán enviar coordenadas periódicamente a la capa inferior.

En la segunda capa de control se colocó el control de desplazamiento. Esta utiliza un controlador LQI, que se seleccionó debido a los buenos resultados que se obtuvieron durante la fase I. Este controlador mide el error que tenga en cuanto posición y orientación el cual penalizará la velocidad requerida por cada mecanismo para llegar a la meta. Esta capa envía las velocidades lineales requeridas por cada mecanismo a la capa inferior.

En la última capa se planteó inicialmente el uso de dos subcapas de control, en las cuales se pensó utilizar un controlador óptimo para trayectorias periódicas en el mecanismo con el fin de mantener una velocidad constante en el remo. Este sub-controlador se descartó debido a los resultados que se obtuvieron en el capítulo anterior, donde se determinó que con un controlador simple como el PID se puede mantener una velocidad constante en la salida. Por esta razón, esta capa pasará esa velocidad lineal a rpm del motor, y se coloca como señal de referencia al controlador PID.

A. Controlador de desplazamiento

Para esta capa se utilizó un controlador LQI donde se controlaron tres parámetros, la posición en X, Y y la posición angular θ respecto al eje X. Se utilizaron las siguientes matrices extendidas:

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Los valores de las matrices Q y K dependieron de si se estaba tratando de llegar a un punto en específico o si se estaba siguiendo una trayectoria. A continuación, se muestran los valores utilizados para ambos casos.

- Punto fijo:

$$Q = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad K = \begin{pmatrix} 1.7321 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1.7321 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1.7321 & 0 & 0 & 1 \end{pmatrix}.$$

- Trayectoria:

$$Q = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 15 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad K = \begin{pmatrix} 2.9574 & 0 & 0 & 3.873 & 0 & 0 \\ 0 & 2.9574 & 0 & 0 & 3.873 & 0 \\ 0 & 0 & 1.7321 & 0 & 0 & 1 \end{pmatrix}.$$

También se planteó el modelo unicyclo del BART que al ser este omnidireccional, su difeomorfismo dio como resultado una matriz de rotación para dos dimensiones:

$$Dif = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

El campo vectorial de la dinámica del sistema está dado por

$$f(x, u) = \begin{pmatrix} v_x \cos(\theta) - v_y \sin(\theta) \\ v_x \sin(\theta) + v_y \cos(\theta) \\ \omega \end{pmatrix},$$

$$u = [v_x \ v_y \ \omega], \quad x = [x_I \ y_I \ \theta].$$

Donde u es la salida del controlador LQI y x representa el estado del sistema. De la salida de control se pudo deducir las velocidades lineales para cada uno de los mecanismos despejando de la ecuación (29). Identifico cada velocidad según su posición en el BART, por ejemplo, si el mecanismo está en la esquina inferior izquierda V_{LD} donde L es Left y D es Down. Para hacer referencia de derecha se usará la letra R y arriba la letra U. Durante la fase dos se determinó que los ángulos α y β eran 0.85 y 0.44 radianes respectivamente. La distancia l del centro del BART al remo en promedio fue de 0.068m.

$$V_{LU} = v_x \csc(\alpha + \beta) - v_y \sec(\alpha + \beta) - 2\omega l \sec(\beta) \quad (38)$$

$$V_{LD} = -v_x \csc(\alpha + \beta) - v_y \sec(\alpha + \beta) + 2\omega l \sec(\beta) \quad (39)$$

$$V_{RU} = v_x \csc(\alpha + \beta) + v_y \sec(\alpha + \beta) + 2\omega l \sec(\beta) \quad (40)$$

$$V_{RD} = -v_x \csc(\alpha + \beta) + v_y \sec(\alpha + \beta) - 2\omega l \sec(\beta) \quad (41)$$

Se realizaron varias simulaciones para corroborar el funcionamiento del controlador. La primera prueba fue seguir una trayectoria circular mientras este giraba sobre su propio eje. La segunda prueba fue siguiendo la misma trayectoria, pero esta vez manteniendo posición angular constante y por último se colocó una coordenada fija a la cual debía llegar.

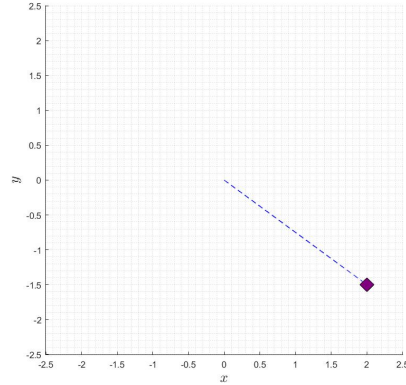
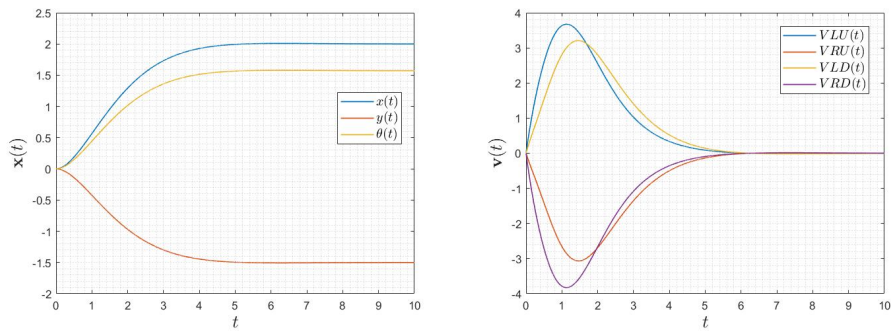


Figura 46: Coordenada (2,-1.5) posición angular $\pi/2$.



(a) Variables de estado.

(b) Velocidades lineales.

Figura 47: Coordenada (2,-1.5) posición angular $\pi/2$.

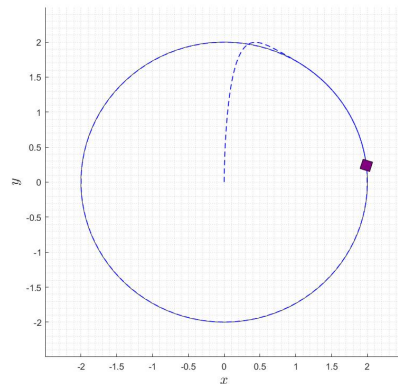


Figura 48: Seguimiento de trayectoria circular con posición angular variable en el tiempo.

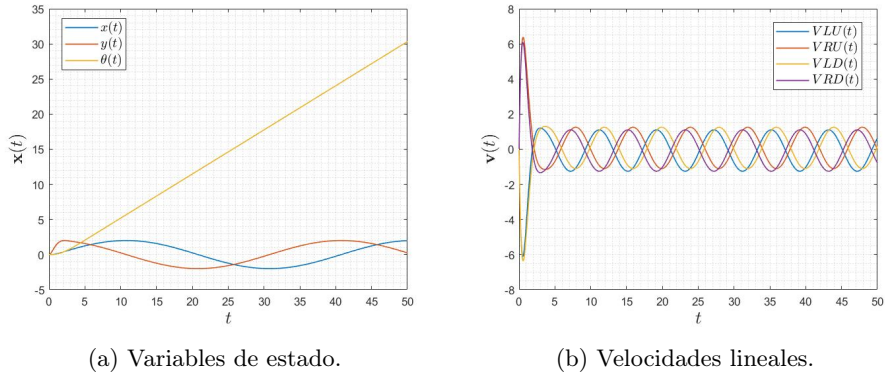


Figura 49: Seguimiento de trayectoria circular con posición angular variable en el tiempo.

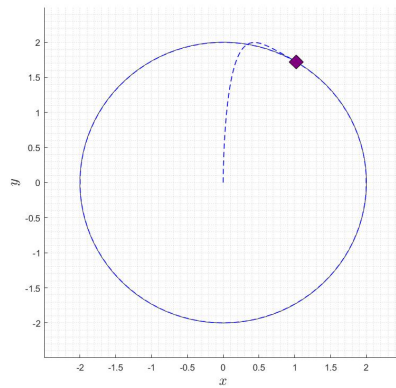


Figura 50: Seguimiento de trayectoria circular con posición angular constante.

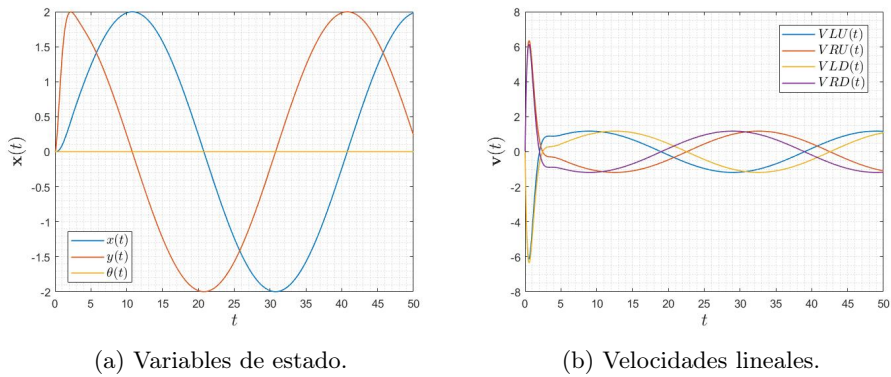


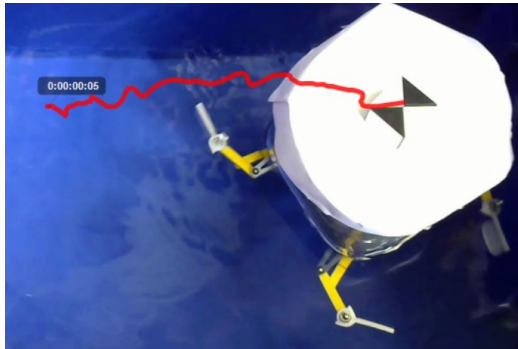
Figura 51: Seguimiento de trayectoria circular con posición angular constante.

Como se muestra para ambos casos el controlador LQI cumplió con el seguimiento de trayectoria a la perfección haciendo uso de la capacidad de movimiento omnidireccional. Para cualquiera de los tres casos mostrados el controlador requirió de tres señales de referencia las cuales consisten en posición x , y y posición angular respecto al marco inercial. Debido a que no se cuenta con un cuerpo de agua lo suficiente mente grande como para poder describir la trayectoria simulada, se validó el controlador mediante simulación. Por lo tanto,

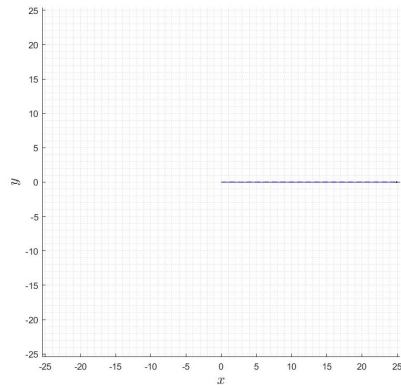
el controlador es válido para el seguimiento de trayectorias y estabilización del BART en un punto arbitrario.

B. Integración de controladores

1. Movimiento lineal con giro

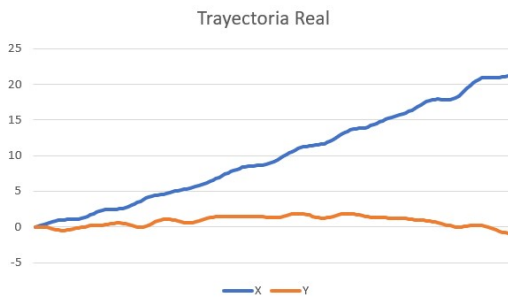


(a) Trayectoria real.

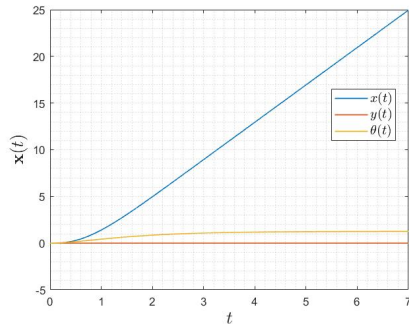


(b) Trayectoria simulada.

Figura 52: Moviente combinado con traslación y rotación.



(a) Trayectoria real.



(b) Trayectoria simulada.

Figura 53: Movimiento combinado con traslación y rotación.

Se analizó ambas pruebas con el software de KINOVEA y estas tuvieron un control a lazo abierto. Para ello se colocó un cuadro en la parte superior del BART con un patrón con el fin de seguir la trayectoria de mejor manera. Para esta prueba se planteó una trayectoria recta mientras se llegaba a una posición angular de 60 grados con respecto a la vertical. La desviación máxima que se obtuvo en el eje Y fue de 1.88 cm, pero conforme avanzó el BART esta desviación se fue reduciendo hasta llegar a 0.83 cm. El ángulo no se pudo medir a cada instante durante el recorrido, por lo que se midió al inicio y al final del recorrido. La posición angular final del BART fue de 55 grados teniendo así un error de 9.3%.

2. Desplazamiento horizontal

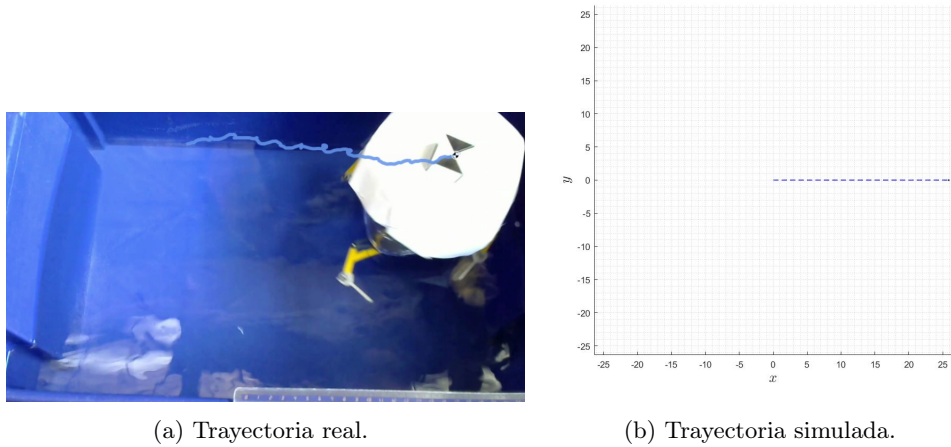


Figura 54: Movimiento de traslación en línea recta.

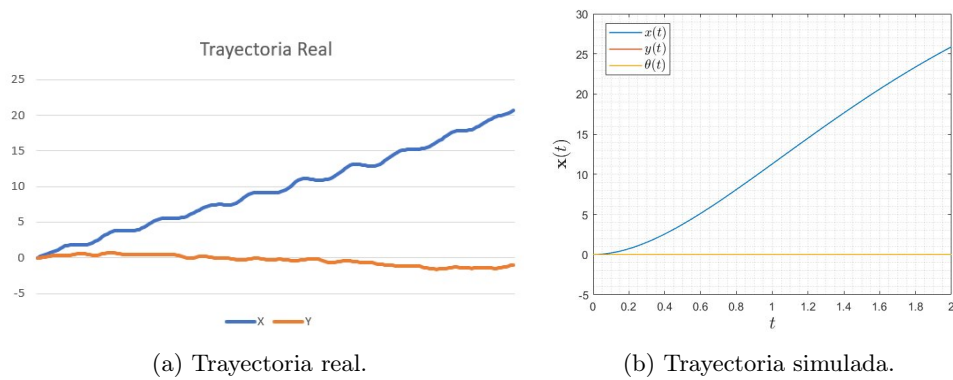


Figura 55: Movimiento de traslación en línea recta.

Se realizó la prueba de desplazarse en línea recta sin girar sobre su propio eje. La desviación máxima fue de 1.62cm, pero al igual que la prueba anterior esta se fue reduciendo. Al finalizar el recorrido se encontró con una posición en Y de 0.76cm, cuando esta tendría que permanecer en 0 durante todo el trayecto. En la Figura 55 de posición en X, se observa que durante las fases de retroceso y transición en los mecanismos permanece casi constante la posición.

A pesar de que el control se realizó a lazo abierto, se obtuvieron buenos resultados. Ya que la posición en Y, se mantuvo cercana a 0 a pesar de que no se contaba con retroalimentación. Por otro lado, al ir variando la posición angular la posición en X no se ve tan pausada como en la segunda prueba. El error que se obtuvo en la posición angular fue de 9.3%. Por lo tanto, el control fue válido para el manejo del BART.

Diseño de placa electrónica

Durante la primera fase se utilizó una plataforma basada en tennsy Arduino, con el cual no se obtuvieron excelentes resultados y para la segunda fase se decidió utilizar un micro-controlador de la familia DSPIC, el cual cumplió con los requerimientos, pero se tenía la limitante de documentación y la cantidad de periféricos disponibles. Tomando estos aspectos en cuenta se decidió el uso de la plataforma Tiva C Launchpad C Launchpad. Esta plataforma posee una gran variedad de periféricos. Además, tiene procesador ARM Cortex-M4 Cortex-M4, los cuales han sido diseñados con características orientadas al procesamiento y control de señales, y pueden ejecutar instrucciones que pertenecen a los DSP. Combinando estas características con la alta velocidad de su CPU permite realizar tareas complejas en tiempos menores. Para la integración de cada uno de los módulos del BART se procedió a diseñar una PCB en forma de shield para la Tiva C Launchpad C.

En la primera fase se utilizó un módulo Xbee para la comunicación con el BART, la gran desventaja que presenta este módulo es su alto precio. Debido a que el BART está pensado para realizar robótica swarm en el futuro, se decidió utilizar un módulo de radiofrecuencia NRF24L01. El precio de este módulo ronda los Q30.00 en comparación con el Xbee que ronda los \$25.00, además este módulo posee un menor tamaño y su velocidad de procesamiento es igual al Xbee ya que los rangos de transmisión de los módulos NRF24L01 y Xbee son de 100m y 150m respectivamente, considerando la relación precio-alcance, se utilizó la versión del NRF24L01, el cual tiene un rango de 1 km y un precio que ronda los Q60.00.

El módulo de alimentación fue diseñado pensando en el uso de sola una celda. Dicha celda corresponde a una batería tipo Li-ion 18650 de 4200mAh. Con el fin de prolongar la vida útil de la batería se usará un TP4056 con protección, que permite un voltaje máximo de carga de 4.2V y un voltaje mínimo de descarga de 2.4V. Para elevar el voltaje de alimentación a 5V, se colocará un MT3608. Este componente puede dar un voltaje máximo de 28V a la salida con una corriente de 2A.

A. Esquemáticos de los circuitos

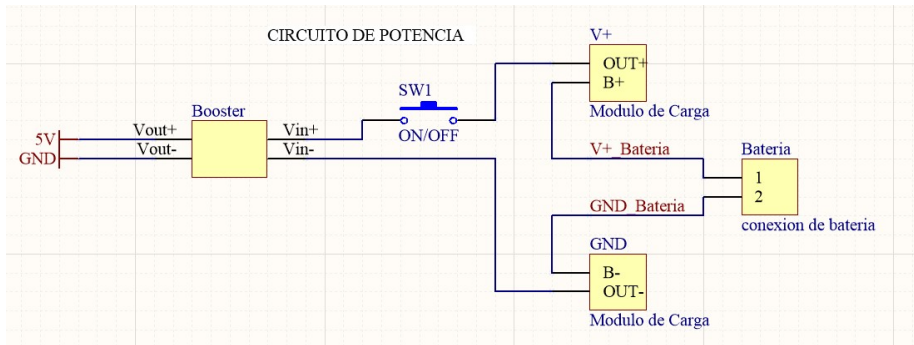


Figura 56: Circuito de alimentación y potencia.

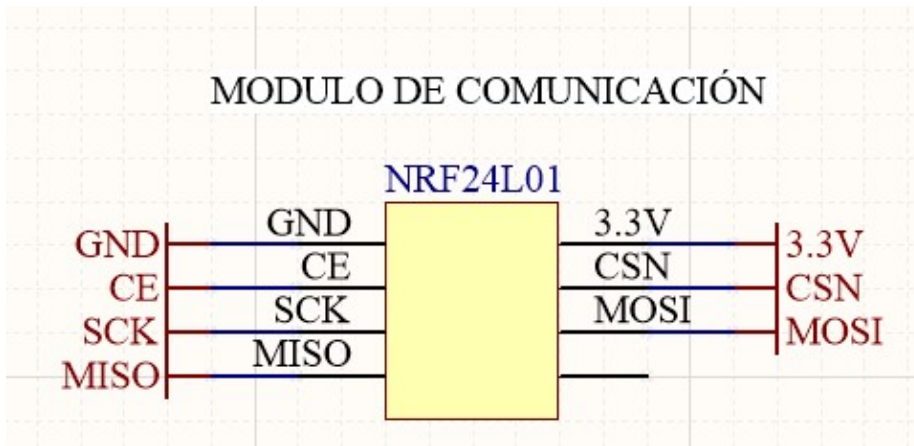


Figura 57: Módulo de comunicación NRF24L01.

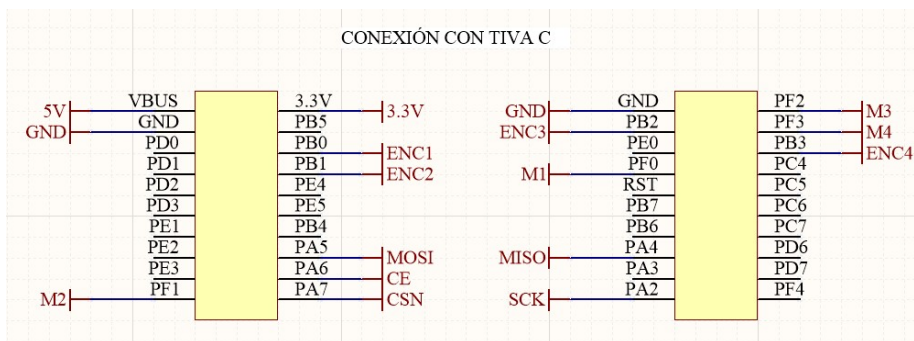


Figura 58: Tiva C Launchpad C.

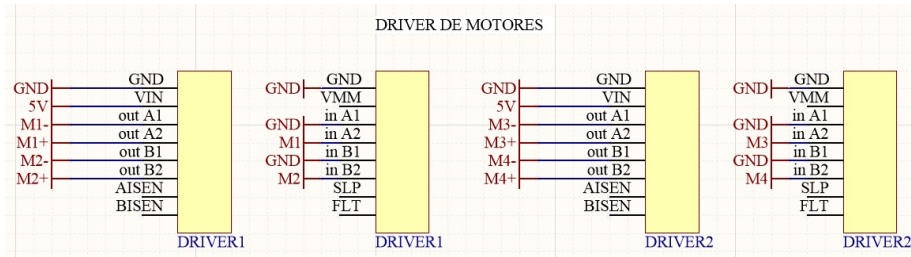


Figura 59: Drivers para motores DC.

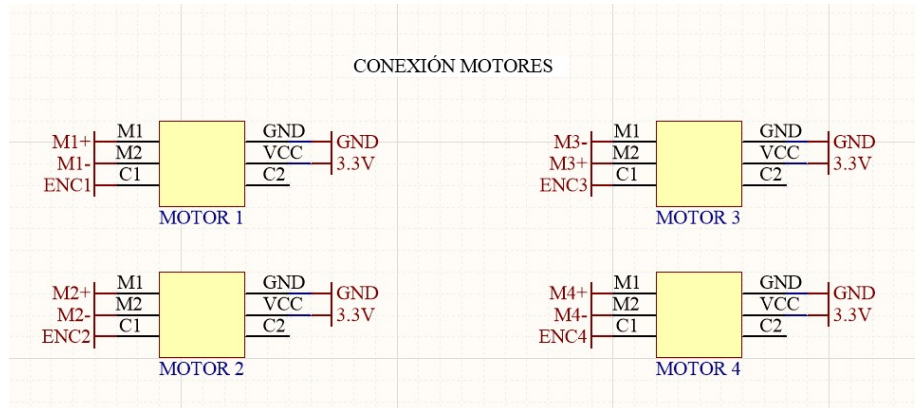


Figura 60: Conexión de motores.

B. Consumo energético

Se calculo el consumo energético del BART tomando el peor de los casos para cada uno de sus componentes. Para ellos se supuso que la Tiva C tendría todos sus módulos encendidos, los motores estaban a carga completa con un ciclo de trabajo del 100% y el peor caso de eficiencia para el MT3608.

Componente	Potencia calculada (W)	Unidades	Total de energía requerida (Wh)
MT3608	0.7	1	0.7
TP4056	0.62	1	0.62
DRV8833	0.15	2	0.3
Motor	0.85	4	3.4
Tiva C	0.193	1	0.193
NRF24L01	0.05	1	0.05
Total de energía consumida por hora			5.26 Wh
Total de energía disponible en la batería			15.54 Wh
Tiempo de operación			2.95 hrs

Cuadro 4: Consumo energético del BART.

C. Cálculos para la PCB

Previo al diseño de la PCB se calcularon los anchos mínimos requeridos para los pistas. Se identificaron cuatro grupos de pistas importantes, los cuales son GND, 5V, 3.3V y pistas en general. Para ello se tomará un aumento de 10°C y un grosor de cobre de 2 Cu/ft^2 .

Pista	Corriente máxima (A)	Ancho (mil)	Ancho utilizado (mil)	Margen (%)
5V	2	15.77	25	58.5
GND	2	15.77	25	58.5
3.3V	1	6.06	15	148
General	0.7	3.71	15	304

Cuadro 5: Ancho de pistas.

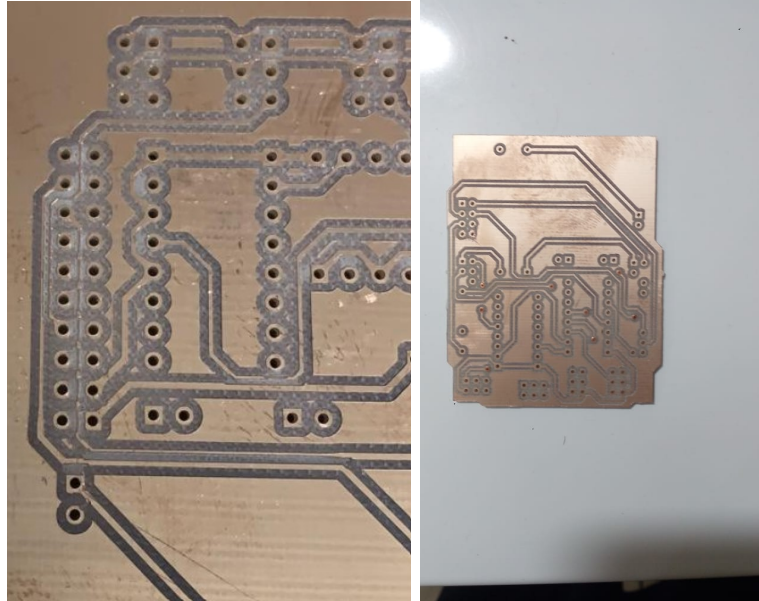
D. Listado de componentes

	Referencia	Descripción	Fabricante	Cantidad	Precio c/u(Q)
1	Driver1, Driver2	DRV8833	Polulu	2	38.00
2	Booster	MT3608	Aerosemi Tech	1	19.00
3	Modulo de carga	TP4056	Nanjing Corp.	1	14.00
4	Motor1, Motor2, Motor3 y Motor4	Header pin 3, dual row		4	1.00
5	Switch	Dip switch	CUI Device	1	1.50
6	NRF24L01	NRF24L01	Nordic Semiconductor	1	27.00
7	P1, P2	Header pin 10, dual row		2	2.00

Cuadro 6: Listado de componentes.

E. Diseño de PCB

Debido a la disponibilidad de las herramientas de fresado que cuenta el MakerLab, se eliminaron varios pads que se conectaban a la Tiva C y no tenían ninguna función de momento. Esto llevo a utilizar headers sueltos para realizar la conexión con la Tiva C Launchpad C - PCB.



(a) Intentos previos.

(b) Diseño final de PCB.

Figura 61: PCB.

El módulo de comunicación, conexión con motores y el boton de encendido quedaron en la parte externa con el fin de evitar posibles interferencias para el caso del módulo NRF24L01 y tener un fácil acceso para encender o apagar el BART.

Se dedujo el modelo dinámico del mecanismo impulsor del BART con el cual se pudo comprobar que por sí mismo es capaz de llegar a un ciclo límite en la trayectoria del efector final. También se comprobó que dicho mecanismo durante su fase de empuje se mueve a una velocidad angular constante de -0.76 rad/s la cual produce una fuerza constante $51.7 \frac{\text{mm}g}{\text{s}^2}$ y un torque constante $2666.06 \frac{\text{mm}^2g}{\text{s}^2}$, esto indicó que el supuesto del BART como un vehículo de cuatro ruedas es válido.

Se implementó un controlador de desplazamiento LQI para el modelo del BART el cual fue capaz de seguir movimientos rectilíneos y circulares con la capacidad de combinar un movimiento traslacional y rotacional simultáneamente. Además, este presentó un gran desempeño al realizar control a lazo abierto en las pruebas físicas, donde se obtuvo un desviación máxima de posición de 1.88cm en el eje Y y un error de orientación de 9.3 %.

Se realizó el diseño y fabricación de una placa PCB para la integración de todos los módulos del BART la cual tuvo un correcto funcionamiento y permitió una mejor organización de las conexiones del hardware. Su fabricación junto con los componentes tuvo un costo total de Q145.50.

Uno de los principales problemas que se dió fue la disponibilidad de un espacio lo suficientemente grande como para replicar las trayectorias de seguimiento que fueron simuladas, por lo que se recomienda realizar futuras pruebas en la piscina de la universidad y probar algoritmos de evasión de obstáculos.

A pesar de las facilidades que ofrece la universidad para fabricar PCBs se recomienda buscar una empresa externa la cual cuente con la capacidad de fabricar PCBs de tres capas o más, con el fin de fabricar una PCB con componentes SMD y THT que ocupe menor espacio y no utilizar módulos prefabricados.

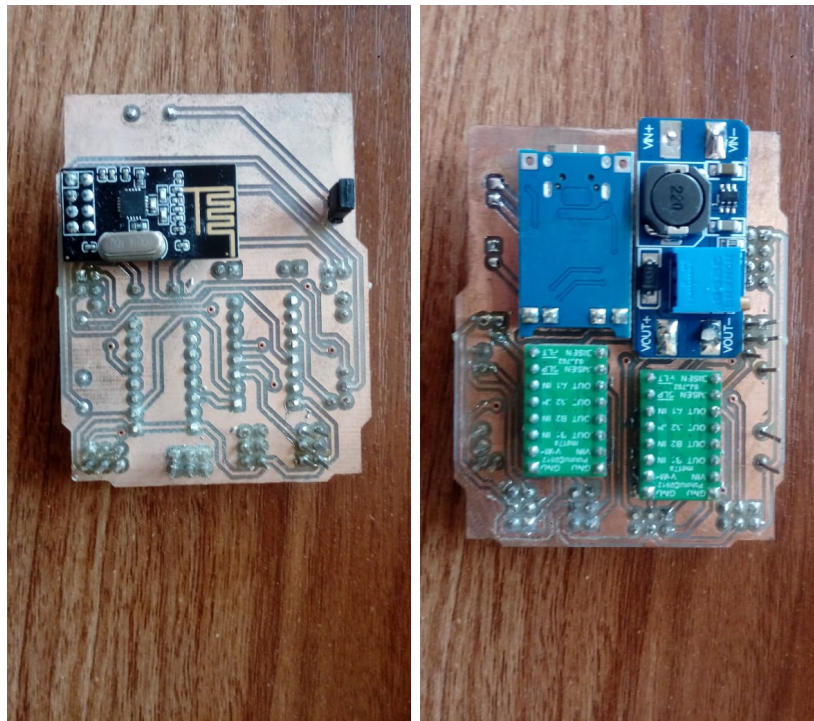
Para futuras investigaciones del mecanismo impulsor se recomienda utilizar una cámara con gran capacidad de fotogramas, ya que muchos de los movimientos del efector final fueron casi imposibles de capturar con precisión con una cámara de 30 fps.

Si se desea utilizar el BART para aplicaciones swarm, se sugiere que se utilice el módulo de radiofrecuencia NRF24L01 +. Este módulo permite la comunicación, permite la creación de 6 canales de comunicación por cada dispositivo NRF24L01. Sería de gran utilidad y simplificaría los algoritmos de comunicación al querer realizar tareas de formaciones o tareas en conjunto.

-
- [1] A. E. Basegoda Curiel, A. S. Gonzalez Noriega, D. Gerendas Obiols, D. Lara Moir, E. A. Chamo Calderon y J. E. Cruz Ubeda, *BART: diseño e implementación de un robot acuático bio-inspirado de nado superficial*. Universidad del Valle de Guatemala, Guatemala, 2017, pág. 264.
 - [2] X. Jia, Z. Chen, A. Riedel, T. Si, W. R. Hamel y M. Zhang, «Energy-efficient surface propulsion inspired by whirligig beetles», *IEEE Transactions on Robotics*, vol. 31, n.º 6, págs. 1432-1443, 2015.
 - [3] Y. N. Meléndez Ovando, *BART: Prototipo de un robot acuático bio-inspirado para nadar omnidireccionalmente*. Universidad del Valle de Guatemala, Guatemala, 2018, pág. 93.
 - [4] Y. Zhang, S. Wang, X. Wang e Y. Geng, «Design and Control of Bionic Manta Ray Robot With Flexible Pectoral Fin», en *2018 IEEE 14th International Conference on Control and Automation (ICCA)*, IEEE, 2018, págs. 1034-1039.
 - [5] R. L. Norton, *Diseño de maquinaria*. Elibro Catedra. McGraw-Hill Interamericana de España S.L., 2009.
 - [6] R. Siegwart, I. R. Nourbakhsh y D. Scaramuzza, *Introduction to autonomous mobile robots*. MIT press, 2011.
 - [7] J. Hespanha, *Linear Systems Theory*. Princeton University Press, New Jersey, United States of America, 2009.
 - [8] K. Ogata, *Ingeniería de Control Moderna*. Tercera Edición, Pearson, 1998, pág. 997.
 - [9] K. J. Åström y R. M. Murray, *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.
 - [10] C.-T. Chen, *Linear system theory and design*. Oxford University Press, Inc., 1998.
 - [11] L. Sandino, M. Bejar y A. Ollero, «Tutorial for the application of Kane's Method to model a small-size helicopter», nov. de 2011. DOI: 10.13140/RG.2.1.4763.4409.
 - [12] I. del Valle Andújar, «Métodos numéricos y analíticos para la determinación de bifurcaciones en sistemas planos discontinuos», 2016.

- [13] R. F. Vassallo, H. J. Schneebeli y J. Santos-Victor, «Visual servoing and appearance for navigation», *Robotics and autonomous systems*, vol. 31, n.º 1-2, págs. 87-97, 2000.

A. PCB



(a) Cara superior

(b) Cara inferior

Figura 62: Placa soldada con todos los módulos

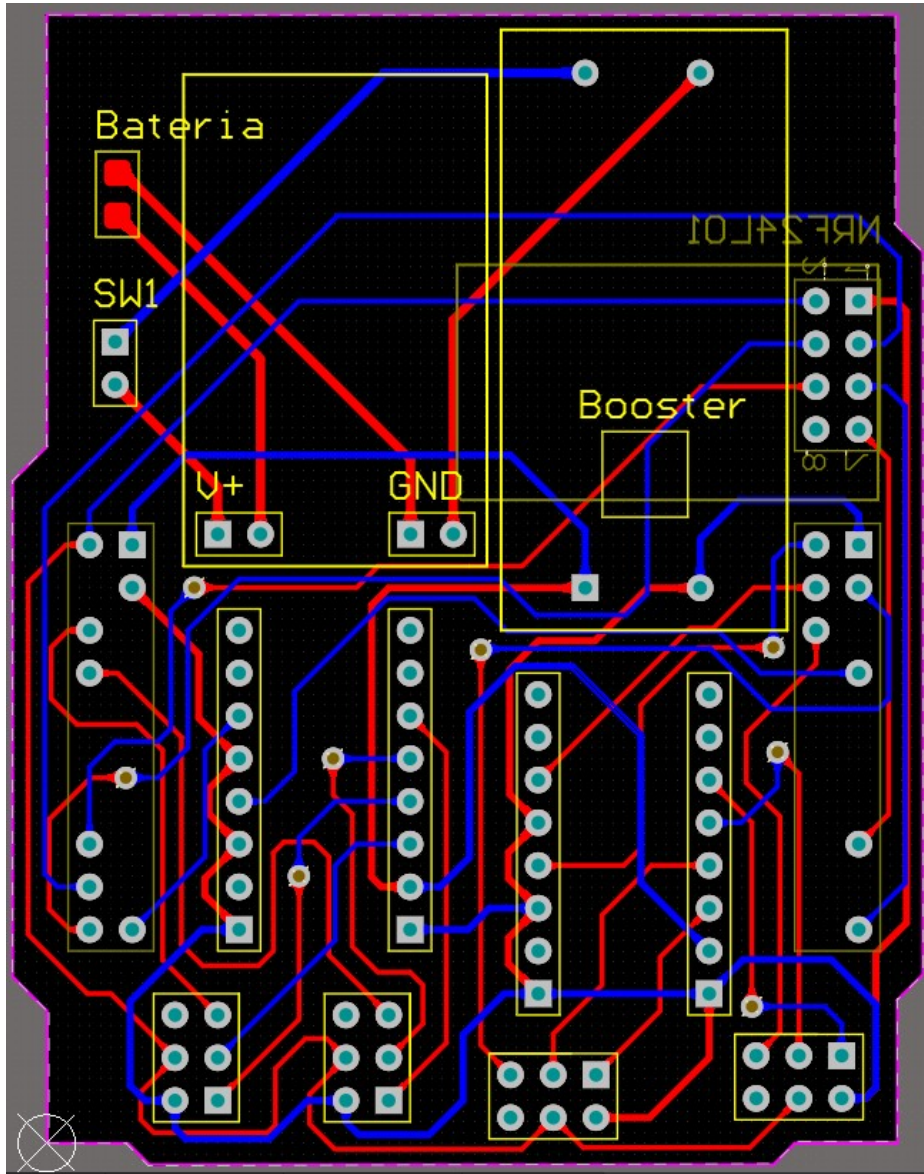


Figura 63: Diseño en Altium

Arduino : Es una plataforma electrónica de código abierto basada en hardware y software fáciles de usar. 59

ARM Cortex-M4 : Microcontroladores que se caracterizan principalmente por proporcionar capacidades para el procesamiento de señal digital. 59

Ciclo límite : Órbita periódica de un sistema. 41, 42, 44

CM : Centro de Masa. 38

CPU : Unidad central de procesamiento. 59

DSP : Procesador de señales digitales. 59

DSPIC : Microcontroladores de 16 bits para procesamiento de señales. 59

GND : (GROUND) Es el punto de referencia en los circuitos donde se encuentra el 0V. 62

IMU : Unidad de medición inercial, el cual cuenta con un giroscopio y acelerómetro. 51

Li-ion : Iones de litio. 59

LIDAR : (Laser Imaging Detection and Ranging) dispositivo que permite determinar la distancia desde un emisor láser a un objeto. 51

LQI : (Linear Quadratic Integral) Tipo de control moderno para varias variables. Es un método basado en LQR. 52, 53, 55

MakerLab : Laboratorio de la Universidad del Valle para la creación de prototipos de piezas. 63

MT3608 : Es un regulador de tipo conmutado elevador con una alta eficiencia de conversión. 59, 61

NRF24L01 : Módulo para comunicaciones por radiofrecuencia. 59, 60, 63

PCB : Una placa de circuito impreso. 33, 59, 62, 63

PD : Proporcional-diferencial. Método de control clásico. 39, 40, 42–44, 46

PID : (Proporcional-Integral-diferencial) Método de control clásico. 52

PWM : (Pulse-width modulation) es una técnica en la que se modifica el ciclo de trabajo de una señal periódica. 33

RF : Radiofrecuencia. 33

Runge-Kutta : Método para resolución numérica de ecuaciones diferenciales. 33

shield : Placas de circuitos modulares que se montan unas encima de otras para dar funcionalidad extra. 59

SMD : (Surface Mounting Device) tecnología para soldar componentes en la superficie de una PCB. 67

THT : (Through Hole Technology) Se usan agujeros pasantes en una PCB para soldar componentes. 67

Tiva C Launchpad : Es una plataforma de prototipos electrónicos basados en una familia de microcontroladores creada por Texas Instruments. 59, 60, 63

TP4056 : Módulo de carga para baterías de Li-ion. 59

Xbee : Módulo para comunicación por radiofrecuencia. 59