



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ETSI Aeroespacial y Diseño Industrial

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Aeroespacial
y Diseño Industrial

Diseño, implementación y control de un prototipo de robot
autoestable BallBot de tres ruedas omnidireccionales

Trabajo Fin de Máster

Máster Universitario en Ingeniería Mecatrónica

AUTOR/A: Marco Arnal, Antonio

Tutor/a: Casanova Calvo, Vicente Fermín

CURSO ACADÉMICO: 2023/2024



ÍNDICE GENERAL

1. INTRODUCCIÓN	5
1.1. MODELOS PREVIOS.....	5
1.2. POSIBLES APLICACIONES	8
1.3. OBJETIVOS	8
2. MARCO TEÓRICO.....	10
2.1. RELACIÓN ENTRE EL MOVIMIENTO DE LAS RUEDAS Y EL DEL ROBOT	11
2.1.1. MOVIMIENTO EN EL EJE X	11
2.1.2. MOVIMIENTO EN EL EJE Y	12
2.1.3. GIRO EN EL EJE Z.....	12
2.1.4. MOVIMIENTO COMPUESTO	13
2.2. CONTROL DEL BALLBOT.....	13
2.2.1. CONTROL DE ESTABILIDAD	13
2.2.2. CONTROL DE POSICIÓN	14
3. SIMULACIÓN.....	15
3.1. SIMULACIÓN DE LA REALIDAD.....	15
3.2. ESQUEMA DE CONTROL	17
3.3. RESULTADOS Y CONCLUSIONES	19
4. IMPLEMENTACIÓN REAL	25
4.1. COMPONENTES MECÁNICOS.....	25
4.1.1. ESFERA.....	26
4.1.2. RUEDAS OMNIDIRECCIONALES.....	27
4.1.3. UNIONES ENTRE LOS MOTORES Y LAS RUEDAS	27
4.1.4. SOPORTES DE LOS MOTORES	29
4.1.5. PLATAFORMA SUPERIOR	30
4.1.6. PLATAFORMA INFERIOR	31
4.2. COMPONENTES ELÉCTRICOS Y ELECTRÓNICOS	31
4.2.1. MOTORES	32
4.2.2. CONTROLADORES DE LOS MOTORES	33
4.2.3. MICROCONTROLADOR	33
4.2.4. IMU.....	34
4.2.5. FUENTE DE ALIMENTACIÓN.....	34
4.3. MONTAJE DEL BALLBOT Y VARIACIONES	35
4.3.1. CAMBIO EN LOS ACOPLÉS DE LOS MOTORES.....	35



4.3.2.	PESO EXTRA	36
4.3.3.	CAMBIO DE ESFERA.....	37
4.3.4.	SISTEMA ANTICAIDA	38
4.4.	ESQUEMA DE CONTROL	39
4.4.1.	PRUEBA DE LOS MOTORES Y LA IMU	39
4.4.2.	ESQUEMA DE ESTABILIDAD ESTANDAR.....	40
4.4.3.	COMPENSACIÓN CON EJE Z.....	40
4.4.4.	FILTRO COMPLEMENTARIO.....	41
4.4.5.	FILTRO COMPLEMENTARIO CON GANANCIA POLINOMIAL.....	42
4.5.	RESULTADOS Y CONCLUSIONES	44
5.	PLIEGO DE CONDICIONES	46
5.1.	SIMULACIÓN	46
5.2.	IMPLEMENTACIÓN REAL	46
6.	PRESUPUESTO	48
6.1.	COMPONENTES DISEÑADOS	48
6.2.	TOTAL	49
7.	CONCLUSIONES Y POSIBLES MEJORAS.....	50
8.	BIBLIOGRAFÍA.....	52
9.	ANEXOS.....	53
9.1.	PLANOS.....	53
9.2.	OBJETIVOS DE DESARROLLO SOSTENIBLE.....	56



ÍNDICE DE FIGURAS

Figura 1. CMU BallBot	6
Figura 2. BallIP	6
Figura 3. Rezero	7
Figura 4. Kugle BallBot	7
Figura 5. Ejes del sistema de referencia	10
Figura 6. Sentido de giro positivo	10
Figura 7. Numeración de las ruedas y motores	11
Figura 8. Simulación 3D del BallBot	15
Figura 9. Simulación de la conexión entre la esfera y el mundo.....	15
Figura 10. Simulación de la conexión entre el BallBot y el mundo.....	16
Figura 11. Simulación de la conexión entre el cuerpo del BallBot y sus ruedas.....	16
Figura 12. Simulación de la conexión de las ruedas con el BallBot y sus rodillos.....	16
Figura 13. Esquema de control general de la simulación	17
Figura 14. Esquema de control de posición de los ejes X e Y de la simulación	18
Figura 15. Esquema de control de posición del eje Z de la simulación	18
Figura 16. Esquema de control de estabilidad de los ejes X e Y de la simulación	18
Figura 17. Seguimiento de referencia de la simulación	19
Figura 18. Seguimiento de trayectoria de la simulación	20
Figura 19. Estabilidad de la simulación.....	21
Figura 20. Sumatorio de los 3 encoders y rotación del eje Z con poca fricción.....	22
Figura 21. Sumatorio de los 3 encoders y rotación del eje Z con mucha fricción	22
Figura 22. Velocidades y aceleraciones de la simulación de las 3 ruedas sin filtro.....	23
Figura 23. Velocidades y aceleraciones de la simulación de las 3 ruedas con filtro.....	23
Figura 24. BallBot	25
Figura 25. Pelota de baloncesto utilizada como esfera.....	26
Figura 26. Ruedas omnidireccionales.....	27
Figura 27. Acople V3 de los motores a las ruedas	28
Figura 28. Acople V3 colocado en el motor.....	28
Figura 29. Soporte de los motores	29
Figura 30. Bridas de los motores	29
Figura 31. Plataforma superior	30
Figura 32. Plataforma inferior	31
Figura 33. Motor Pololu	32
Figura 34. Controlador del motor	33
Figura 35. Arduino Due.....	33
Figura 36. IMU MPU6050	34
Figura 37. Fuente de alimentación	34



Figura 38. Posición original del soporte.....	35
Figura 39. Posición modificada del soporte	36
Figura 40. Saco de peso extra.....	36
Figura 41. BallBot con peso extra	37
Figura 42. BallBot sobre la pelota de yoga	38
Figura 43. Sistema anticaída original	38
Figura 44. Esquema de prueba de los motores y la IMU	39
Figura 45. Esquema de estabilidad.....	40
Figura 46. Esquema de estabilidad con compensación del eje Z	41
Figura 47. Esquema de estabilidad con filtro complementario	42
Figura 48. Esquema de pruebas de polinomios	43
Figura 49. Resultado de pruebas de polinomios.....	43
Figura 50. Esquema de estabilidad con filtro complementario y ganancia polinomial..	44



1. INTRODUCCIÓN

Este Trabajo de Fin de Master trata del diseño, la implementación y el control de un prototipo de robot tipo BallBot movido por tres ruedas omnidireccionales. Este tipo de robot se caracteriza por estar diseñado para mantener el equilibrio sobre una esfera, la cual es el único contacto con el suelo, debido a esto es capaz de ir en cualquier dirección sin necesidad de rotar el robot, en consecuencia tiene un movimiento muy natural además de ser muy ágil y maniobrable comparado a otro tipo de vehículos terrestres.

Con el fin de controlar el BallBot para que se mantenga estable y se mueva al lugar deseado este dispone de tres ruedas omnidireccionales con desplazamiento lateral cada una de las cuales cuenta con un motor independiente. Dichos motores recibirá una acción de control calculada por un microcontrolador en base a los datos obtenidos por la IMU.

El principio de funcionamiento de un robot tipo BallBot es como el de un péndulo invertido, con la diferencia de que el robot puede caer en cualquier dirección en lugar de una, lo que añade un grado extra de dificultad al control.

En primer lugar se hará una simulación del comportamiento del BallBot hasta obtener los resultados esperados, una vez esto se haya conseguido se montará el robot y se intentará replicar el mismo comportamiento de la simulación en el mundo real.

1.1. MODELOS PREVIOS

En el año 2005 el profesor Ralph Hollis del Instituto de Robótica de la Universidad Carnegie Mellon desarrolló el primer modelo de BallBot funcional. Este fue diseñado para ser de tamaño humano tanto en altura como en anchura y demostró una gran capacidad a la hora de soportar perturbaciones tales como empujones, patadas o colisiones contra obstáculos robustos, además de contar con un movimiento ágil y elegante capaz de circular por entornos humanos.



Figura 1. CMU BallBot

El siguiente paso en la historia de los robots tipo BallBot sería en el año 2008 cuando el profesor Masaaki Kumagai de la universidad de Tohoku Gakuin, situada en Japón, desarrolló el BallIP con el que demostraron la capacidad de los BallBots para transportar cargas. Además estos BallBots eran más pequeños y capaces de trabajar en conjunto.

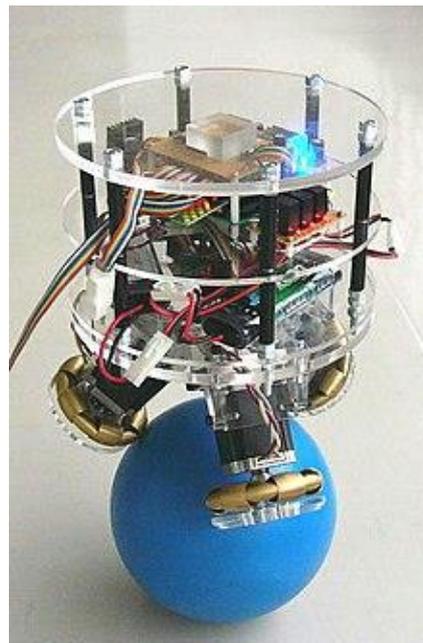


Figura 2. BallIP

En el año 2010 un grupo de estudiantes de ingeniería mecánica de la Escuela Politécnica Federal de Zúrich, en Suiza, crearon Rezero con el cual mejoraron la capacidad de realizar movimientos ágiles de un BallBot añadiendo unos soportes que evitan que el robot se separe de la esfera.

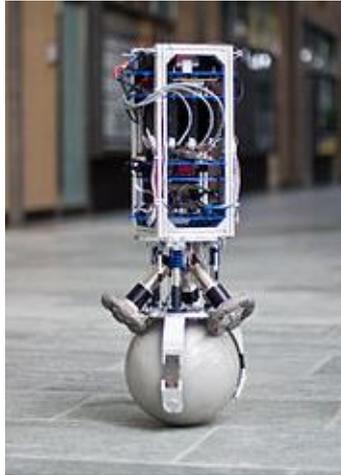


Figura 3. Rezero

La última de las innovaciones en los robots tipo BallBot fue en el año 2019, año en el que Thomas Kølbæk Jespersen perteneciente a la Universidad de Aalborg, en Dinamarca, desarrollo el Kugle BallBot, el cual es capaz de maniobrar de manera autónoma en interiores y guiar a las personas.

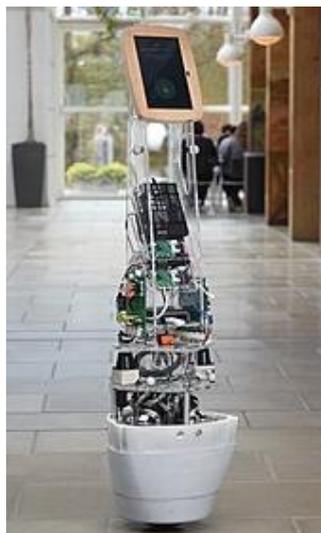


Figura 4. Kugle BallBot



1.2. POSIBLES APLICACIONES

Teniendo en cuenta las características que diferencia a los BallBot de otro tipo de robots, estos pueden ser ideales para hacer asistentes robóticos personales capaces de seguir a personas para ayudarles en sus tareas diarias y llevar cargas por ellos. En la misma línea de interacción con personas podrían servir de guía en museos y aeropuertos o por la calle para personas con discapacidad visual.

Otro enfoque podría ser desde el punto de vista de la seguridad, podría utilizarse este tipo de robot, acoplándole una cámara, para realizar rondas de vigilancia que suplementen la visión que pueden aportar las cámaras estáticas y dar visión de las zonas deseadas o realizar comprobaciones visuales sin poner en riesgo a seres humanos.

También se podría utilizar la tecnología BallBot para sustituir las ruedas de las sillas de ruedas, aportándoles una mayor maniobrabilidad para que sean capaces de ir en todas las direcciones en cualquier momento, de esta manera podrían circular con mayor naturalidad entre los demás transeúntes.

A la hora de buscar en qué aplicaciones específicas se pueden utilizar los BallBot no hay que tener en cuenta únicamente sus virtudes, también hay que tener en cuenta sus puntos más débiles. Uno de los mayores limitantes es que, al menos los modelos existentes hoy en día, no son capaces de circular por superficies muy irregulares, por lo que no sería la opción más adecuada para, por ejemplo, realizar las tareas de acompañamiento, vigilancia móvil o guía en el entorno agrícola. Las escaleras o desniveles también complican la aplicación de este tipo de robot, necesitando en dicho caso un ascensor inteligente coordinado con el BallBot, la instalación de rampas o al menos un BallBot en cada nivel.

1.3. OBJETIVOS

El objetivo final es el desarrollo de un BallBot funcional, la primer parte del proyecto se basa en hacer una simulación, lo más parecida posible al mundo real, y se ajustará un controlador para conseguir que el BallBot sea capaz de mantenerse en pie y seguir trayectorias.

Una vez la simulación esté completa se empezará con la segunda parte del proyecto, que es la implementación real, para lo cual se hará la impresión en 3D de las partes necesarias, la selección y calibración de componentes, las pruebas y el montaje



del BallBot, cuyo objetivo será que tenga un comportamiento lo más similar posible al obtenido en la simulación.

2. MARCO TEÓRICO

El primer paso para desarrollar un marco teórico es definir un sistema de referencia, en este caso los ejes X e Y se sitúan en la plataforma inferior de la siguiente manera:

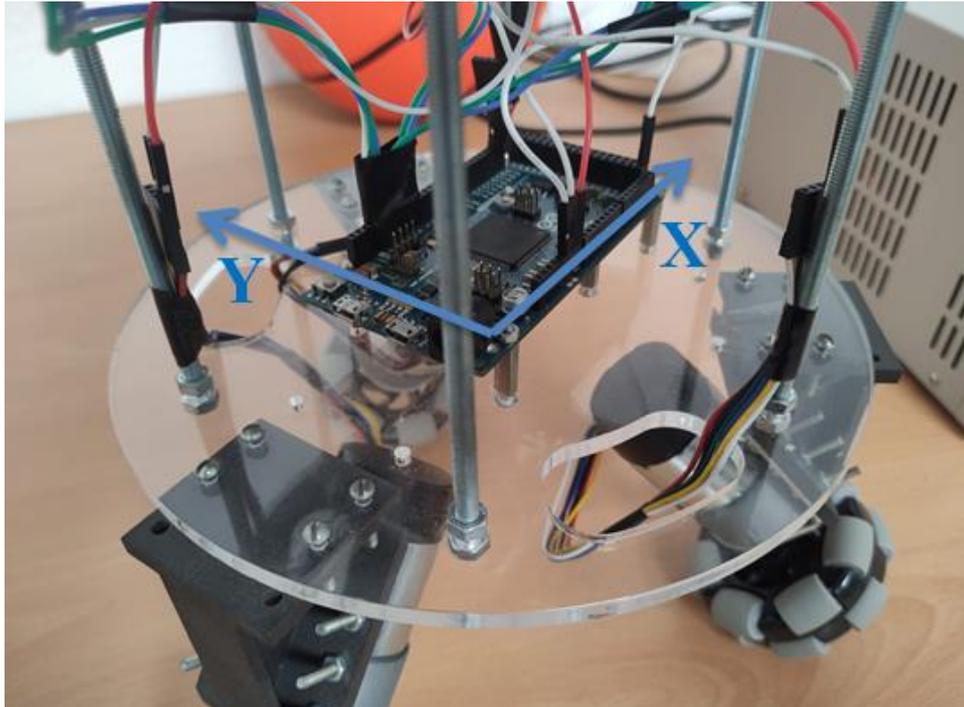


Figura 5. Ejes del sistema de referencia

En cuanto al movimiento de giro de las 3 ruedas tomaremos el sentido antihorario como positivo.



Figura 6. Sentido de giro positivo

2.1. RELACIÓN ENTRE EL MOVIMIENTO DE LAS RUEDAS Y EL DEL ROBOT

Para establecer una relación entre el giro de las ruedas y el movimiento del BallBot es necesario establecer un número a cada rueda y su correspondiente motor.

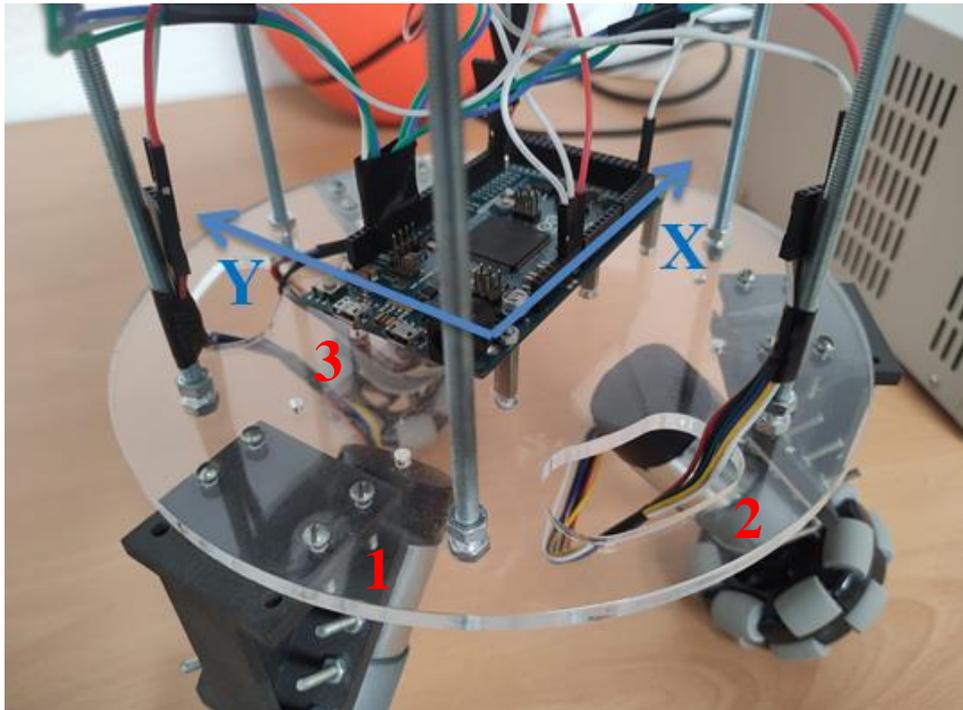


Figura 7. Numeración de las ruedas y motores

El movimiento se analizará el movimiento en los ejes X e Y y el giro en el eje Z por separado para después unirlos y crear el movimiento compuesto.

2.1.1. MOVIMIENTO EN EL EJE X

Puesto que el giro de la rueda 1 es perpendicular a la trayectoria de del eje X esta no influye en el movimiento en dicho eje, los rodillos que posee la rueda omnidireccional sí que la hará deslizar pero el motor 1 no debería girar.

En lo que refiere a las ruedas 2 y 3 tienen una orientación igual y opuesta respecto al movimiento en el eje X, por lo que sus velocidades tendrán la misma magnitud pero en sentido opuesto.



Siendo r el radio de las ruedas, d la distancia del eje Z el área de contacto de las ruedas con la esfera, v la velocidad, bien de las ruedas, cuando tienen un subíndice numérico, o bien del BallBot en los ejes de referencia, cuando tienen un subíndice alfabético y w para la velocidad angular, los subíndices son iguales que los de las velocidades. Considerando el desplazamiento en el eje X se obtienen las siguientes ecuaciones para cada una de las ruedas:

$$v_1 = 0 \quad (1) \quad v_2 = -v_X * \sin 60 \quad (2) \quad v_3 = v_X * \sin 60 \quad (3)$$

$$w_1 = 0 \quad (4) \quad w_2 = \frac{-v_X * \sin 60}{r} = \frac{-v_X * \sqrt{3}}{2 * r} \quad (5) \quad w_3 = \frac{v_X * \sin 60}{r} = \frac{v_X * \sqrt{3}}{2 * r} \quad (6)$$

2.1.2. MOVIMIENTO EN EL EJE Y

El dirección en la que gira la rueda 1 es paralela a la del eje Y por lo que la distancia recorrida por la rueda es igual a la avanzada en el eje Y. Las ruedas 2 y 3 también participan en el desplazamiento en el eje Y. Las ecuaciones del movimiento en el eje Y para cada rueda quedan de la siguiente manera:

$$v_1 = v_Y \quad (7) \quad v_2 = -v_Y * \sin 30 \quad (8) \quad v_3 = -v_Y * \sin 30 \quad (9)$$

$$w_1 = \frac{v_Y}{r} \quad (10) \quad w_2 = \frac{-v_Y * \sin 30}{r} = \frac{-v_Y}{2 * r} \quad (11) \quad w_3 = \frac{-v_Y * \sin 30}{r} = \frac{-v_Y}{2 * r} \quad (12)$$

2.1.3. GIRO EN EL EJE Z

El giro en el eje Z es el único de los 3 movimientos que es igual para las 3 ruedas por lo que sus ecuaciones también serán iguales:

$$v_1 = v_2 = v_3 = -w_Z * d \quad (13)$$

$$w_1 = w_2 = w_3 = \frac{-w_Z * d}{r} \quad (14)$$



2.1.4. MOVIMIENTO COMPUESTO

El movimiento compuesto para cada una de las ruedas será la suma de sus movimientos en los ejes X e Y más el giro en el eje Z, lo que queda de la siguiente manera:

$$w_1 = \frac{v_Y - w_Z * d}{r} \quad (15)$$

$$w_2 = \frac{-v_X * \sqrt{3} - v_Y - 2 * w_Z * d}{2 * r} \quad (16)$$

$$w_3 = \frac{v_X * \sqrt{3} - v_Y - 2 * w_Z * d}{2 * r} \quad (17)$$

2.2. CONTROL DEL BALLBOT

El control del BallBot tiene dos partes, una parte imprescindible de control de estabilidad, la cual se encarga de mantener el robot sobre la esfera y evitar que se caiga, y otra parte que se encarga de controlar la posición en la que se encuentra el BallBot, también es importante mantener un correcto control de este parámetro pero nunca debería comprometer la estabilidad del sistema.

2.2.1. CONTROL DE ESTABILIDAD

El control de estabilidad se puede dividir en los ejes X e Y, de manera que los ejes de la IMU coincidan con los del control de estabilidad y el sistema de referencia.

Puesto que la posición en la que deseamos que se mantenga el BallBot es una posición de equilibrio inestable, este no se puede mantener totalmente quieto en esa posición y precisará constantes pequeñas correcciones para mantenerse cerca del punto de equilibrio.

Dado que la acción integral tiene como propósito eliminar el error en estado estacionario y sabiendo que debido a las características del sistema no será capaz de lograrlo, la mejor opción es no incluirlo en el control puesto aumentaría la inestabilidad del sistema, por eso se opta por utilizar un controlador PD para cada eje.



2.2.2. CONTROL DE POSICIÓN

El tener un control de posición no solo posibilitará que el BallBot vaya a una posición deseada, también hace que el robot se pueda mantener en una posición y no se desplace de manera involuntaria. La acción integral en el control de estabilidad también podría haber ayudado a mantenerse en la posición inicial pero no es necesario al añadir este control.

El control de posición utiliza los mismos ejes que el control de estabilidad para seguir coincidiendo con los de la IMU y el sistema de referencia.

Ya que el control de posición no debería desestabilizar el sistema hasta el punto de que el BallBot se caiga de la esfera su acción estará limitada. Puesto que no es necesario que la posición del BallBot sea exacta y con el fin de reducir la inestabilidad el controlador para la posición también será un PD.

3. SIMULACIÓN

La simulación se realizará en Simulink con la librería de Simscape Multibody y los modelos 3D del BallBot, la esfera y las ruedas omnidireccionales.

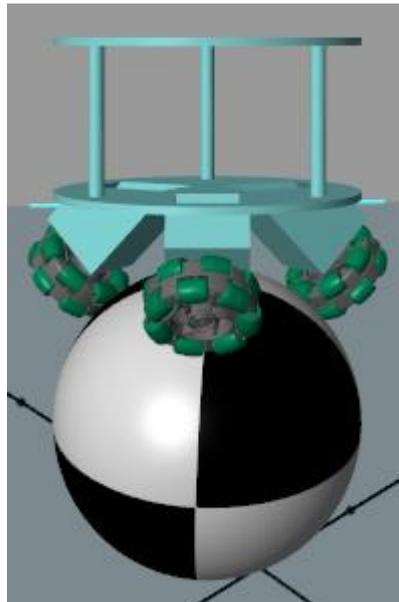


Figura 8. Simulación 3D del BallBot

3.1. SIMULACIÓN DE LA REALIDAD

Para simular la realidad debemos conectar correctamente cada uno de los modelos que componen el sistema entre ellos así como al mundo. Las conexiones son las siguientes:

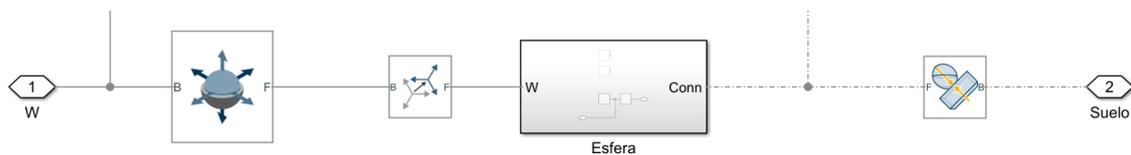


Figura 9. Simulación de la conexión entre la esfera y el mundo

La esfera tiene libre movimiento dentro del mundo y una fuerza normal en contacto con el suelo.

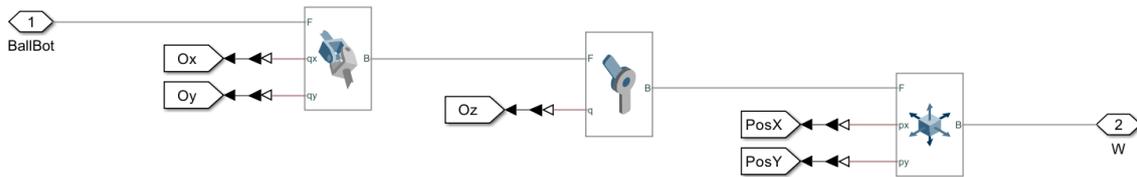


Figura 10. Simulación de la conexión entre el BallBot y el mundo

En este caso en lugar de dar total libertad al BallBot con un solo bloque, se ha hecho por pasos para de esa manera obtener cada uno de los valores deseados por separado.

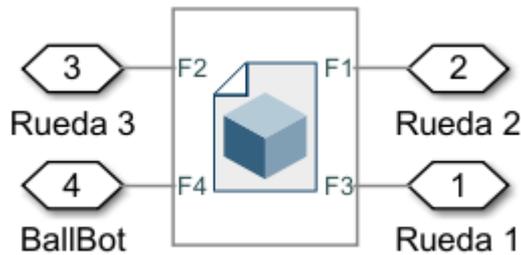


Figura 11. Simulación de la conexión entre el cuerpo del BallBot y sus ruedas

Se le aplican las libertades de movimiento observadas en la Figura 10 al cuerpo del BallBot y se unen las ruedas en sus correspondientes puntos del chasis.

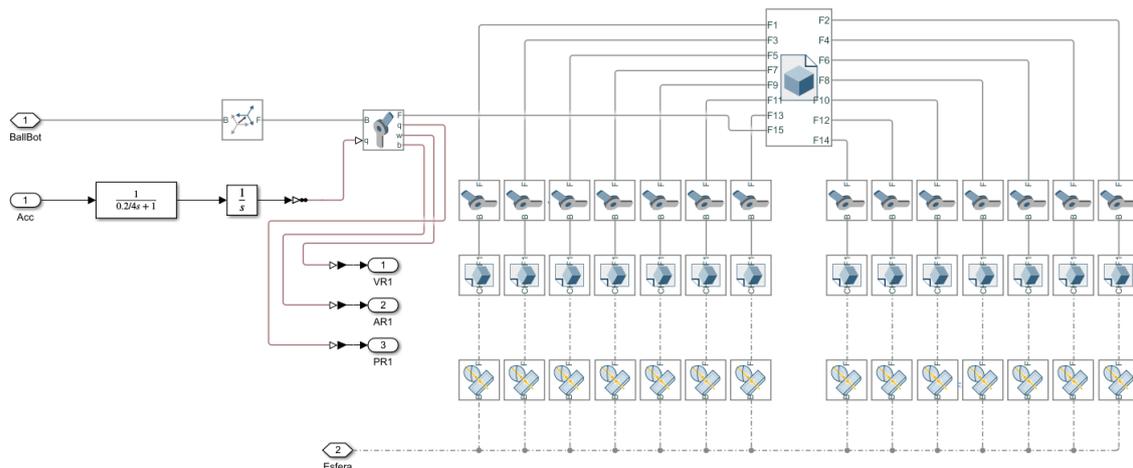


Figura 12. Simulación de la conexión de las ruedas con el BallBot y sus rodillos

La línea de arriba de las dos de la izquierda une el BallBot con el modelo de la rueda, que es el rectángulo de más arriba y algo a la derecha de la imagen, otorgándole libertad de rotación con el segundo bloque que atraviesa dicha línea. A ese bloque también se le une la otra línea que hay a la izquierda la cual lleva la acción de control y atraviesa las transformadas correspondientes al motor.

La línea de abajo del todo viene de la esfera y a través de fuerzas normales se conecta a cada uno de los rodillos, estos a su vez se conectan mediante un par rotatorio con el cuerpo de la rueda.

Este esquema de conexiones es igual para todas las ruedas, únicamente cambia el lugar del BallBot al que se unen y las acciones de control que se aplican.

3.2. ESQUEMA DE CONTROL

El esquema general de control para la simulación del BallBot final con sus conexiones es el siguiente:

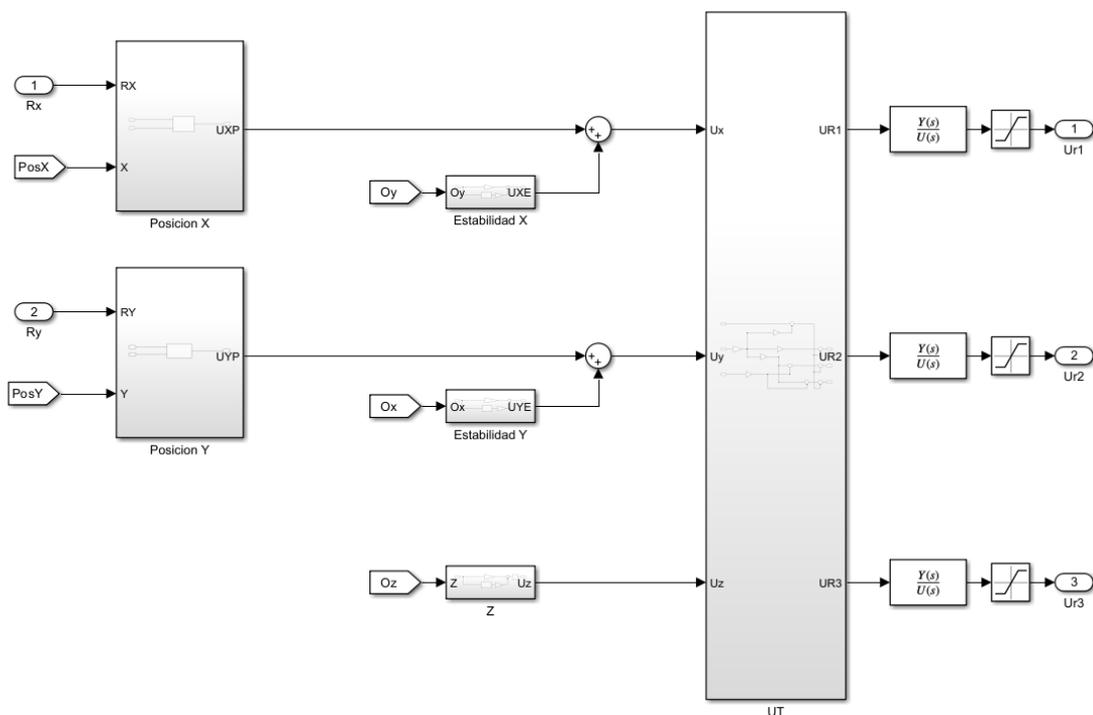


Figura 13. Esquema de control general de la simulación

Para los ejes X e Y se aplican por separado los controles de posición y estabilidad para después sumarse, por otra parte el eje Z solo tiene el control de posición.

Posteriormente se introducen los valores en el bloque UT el cual con dichos valores calcula las acciones de control necesarias que se debe aplicar a cada rueda. Por último las acciones de control de cada rueda se pasa por un filtro paso bajo antes de limitarlas con una saturación y aplicarlas a las ruedas.

Los esquemas de los controladores de posición X e Y son iguales, el esquema normal de un PD con un bloque de saturación para evitar un valor muy alto que pueda llegar a desestabilizar el sistema:

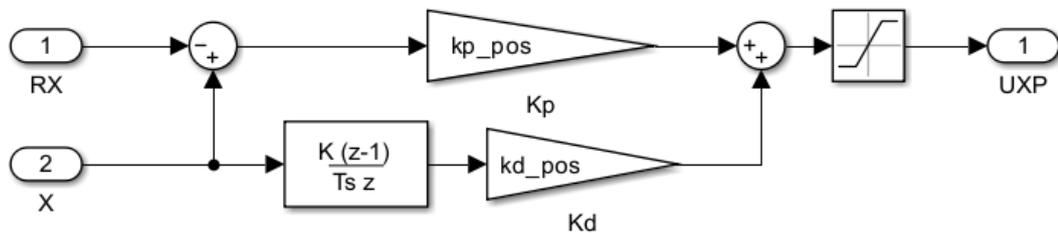


Figura 14. Esquema de control de posición de los ejes X e Y de la simulación

En el caso del controlador de posición del eje Z es muy similar a los otros dos con la excepción de que al ser la referencia constante y en este caso 0 el esquema cambia ligeramente:

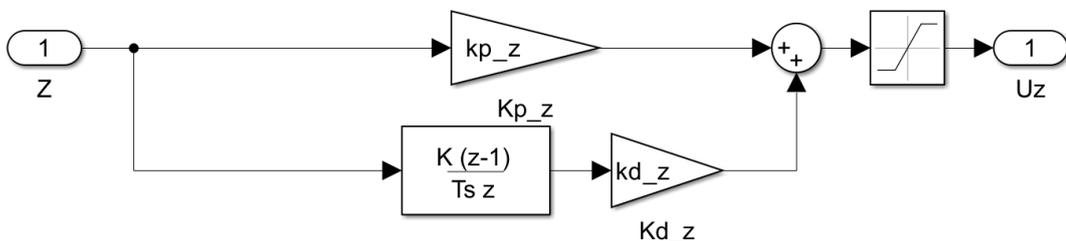


Figura 15. Esquema de control de posición del eje Z de la simulación

Los esquemas de control de estabilidad para los ejes X e Y son iguales entre sí y con el control de posición del eje Z, ya que también es un PD con referencia 0, la diferencia es que al controlar la estabilidad no tiene una saturación ya que es la acción prioritaria.

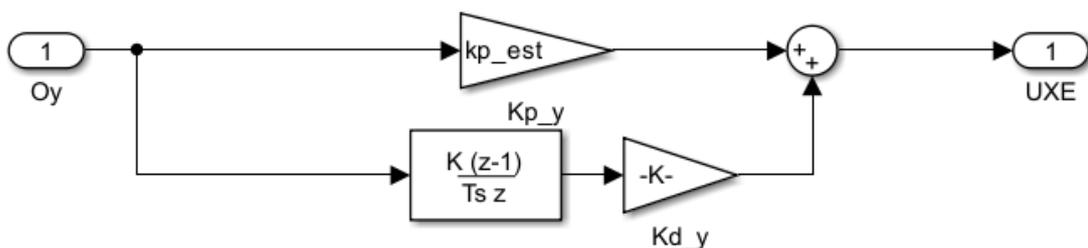


Figura 16. Esquema de control de estabilidad de los ejes X e Y de la simulación

3.3. RESULTADOS Y CONCLUSIONES

Para comprobar el correcto funcionamiento del BallBot en la simulación se ha diseñado una trayectoria de referencia la cual consta de dos senoidales, una en el eje X y otra en el eje Y, este caso en concreto la del eje X consta de dos periodos y la del eje Y un único periodo:

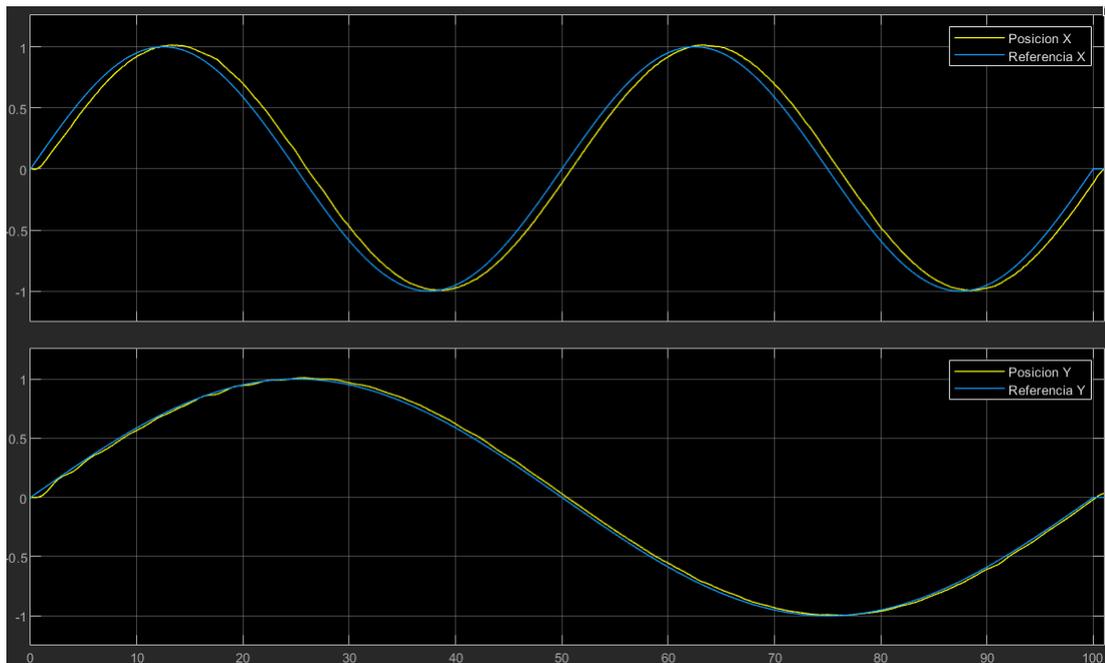


Figura 17. Seguimiento de referencia de la simulación

Podemos observar como el seguimiento de la referencia en la simulación es muy correcto y reproduce las senoidales casi a la perfección. Es cierto que cuenta con un pequeño desfase temporal pero es completamente normal en este caso y no afecta a la funcionalidad del BallBot.

Otra manera de representar esta información es ver la comparativa entre la trayectoria de referencia y la recorrida por el BallBot en la simulación, si bien es cierto que con este otro tipo de representación no es visible el desfase temporal, es mejor para observar el error en el seguimiento de trayectoria.

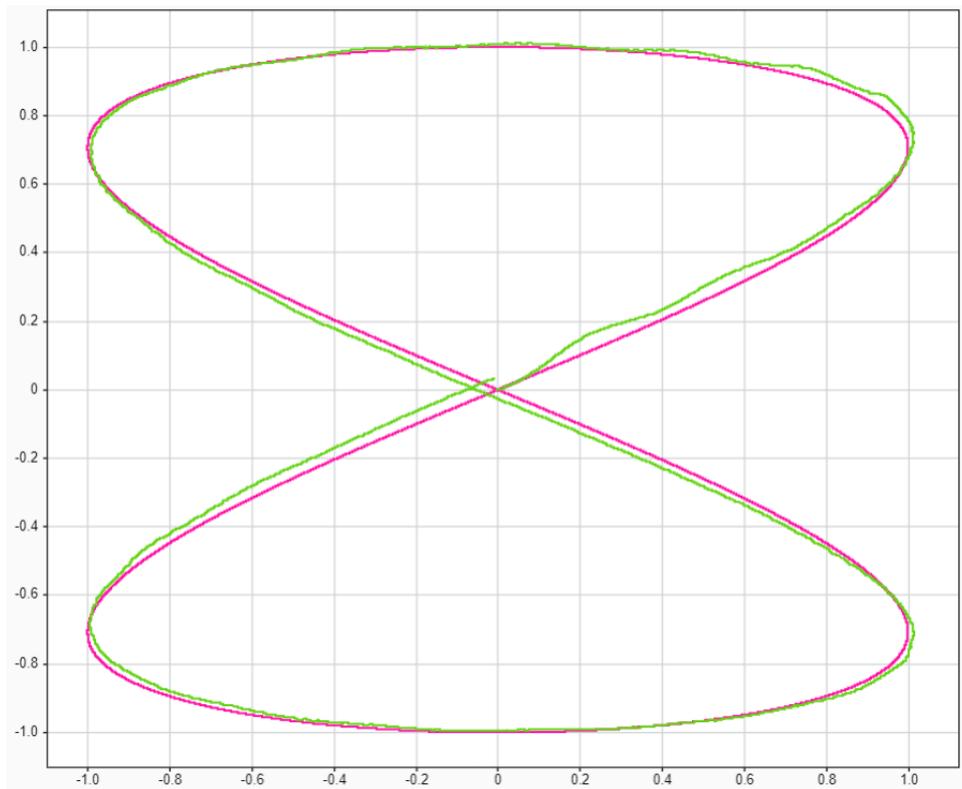


Figura 18. Seguimiento de trayectoria de la simulación

Con este tipo de grafica se puede observar mejor el error en el seguimiento de la trayectoria en la simulación, pero aun viéndose algo mayor sigue siendo un buen comportamiento.

Podemos observar como los mayores errores están al principio, esto se debe a que el BallBot parte de parado por lo que en el tiempo que tarda en arrancar la referencia se aleja lo que aumenta la acción de control de posición. También podemos ver como la primera curva tiene un error mayor que las otras, esto es debido a que aún no se ha estabilizado la distancia entre la trayectoria de referencia y la del BallBot, dicho de otra manera el desfase temporal entre ambas señales, que ha aumentado en el arranque, aún no se ha estabilizado. Una vez este valor se ha vuelto estable el seguimiento de la referencia mejora mucho, hasta el momento en el que esta llega a su punto final, lo cual hace que el BallBot se acerque al punto que marca la referencia, decelerando el BallBot y empeorando ligeramente su seguimiento de la trayectoria.

La otra parte fundamental del comportamiento, la cual es incluso más importante, se trata del control de estabilidad de la simulación.

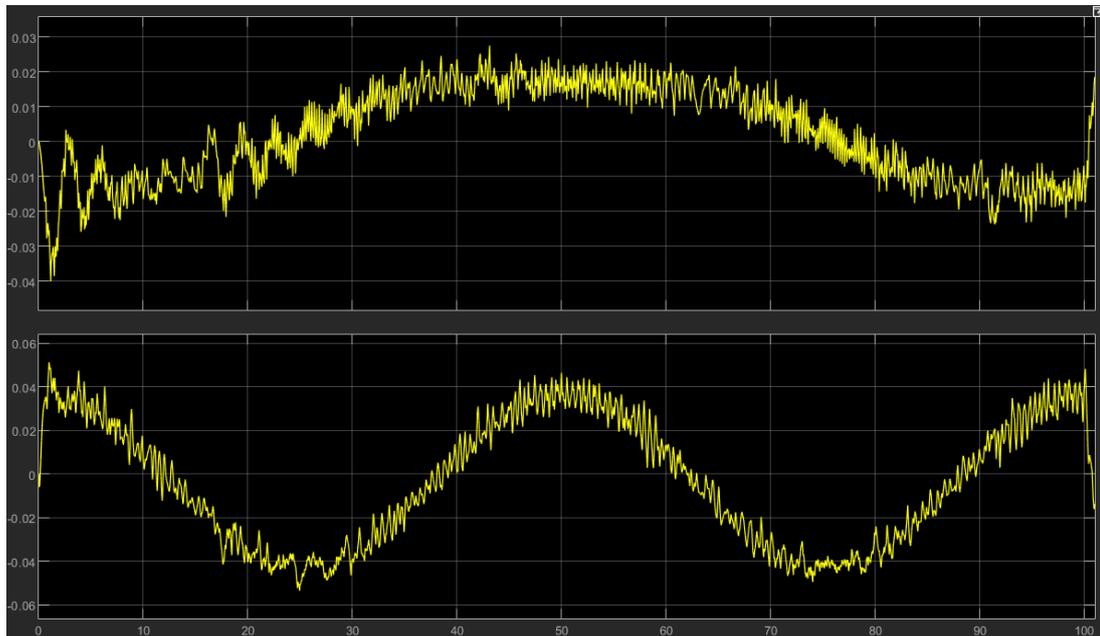


Figura 19. Estabilidad de la simulación

La forma de las ondas se debe al propio seguimiento de la referencia y son las oscilaciones rápidas las encargadas de estabilizar el sistema. Se puede observar como el sistema es bastante estable, las variaciones angulares necesarias para mantener el BallBot en pie son muy pequeñas a excepción del inicio y el final del recorrido puesto que se necesita esta inclinación para acelerar y decelerar el sistema. Debido al mismo efecto de estabilización del desfase temporal, el comportamiento una vez se ha estabilizado es mejor y las oscilaciones se reducen.

Como primera conclusión, a parte del correcto funcionamiento de la simulación del BallBot, se ha observado como las posiciones de las ruedas se pueden utilizar como un único pseudo-encoder absoluto que nos permite conocer la posición en los ejes X e Y del BallBot, siempre que asumamos que no hay deslizamientos. Otro factor que puede añadir algo de error a la medida instantánea de la posición mediante este método es la inclinación del BallBot, pero esto se debería acabar compensando con el controlador de estabilidad una vez el BallBot llegue a su posición final. En caso de querer saber la posición real se puede calcular teniendo en cuenta los ángulos de inclinación como factor corrector del error.

Si sumamos la posición angular que nos proporciona el encoder de cada una de las ruedas podemos obtener el ángulo total que el BallBot ha girado en el eje z.

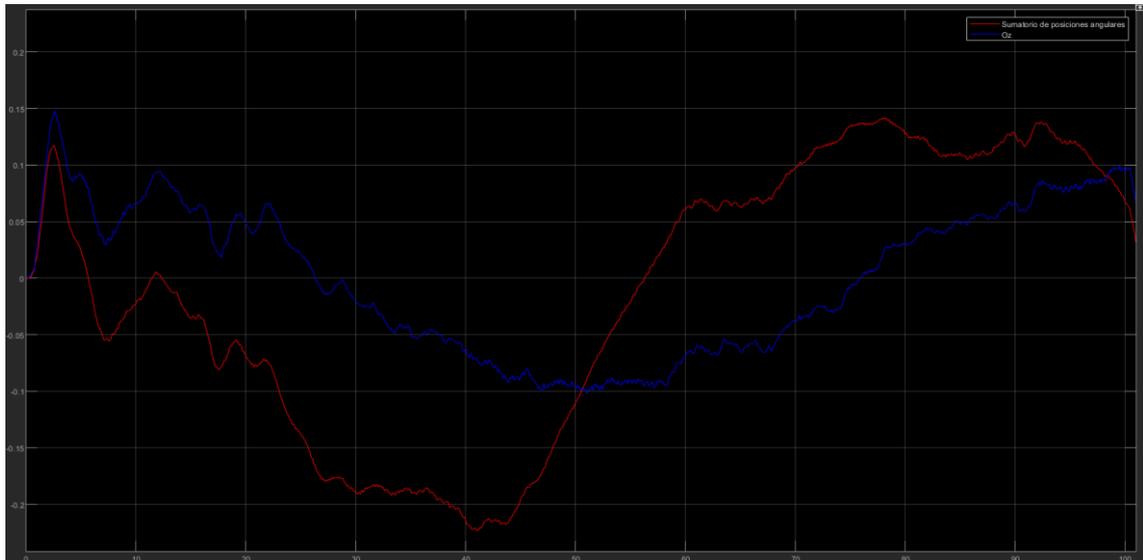


Figura 20. Sumatorio de los 3 encoders y rotación del eje Z con poca fricción

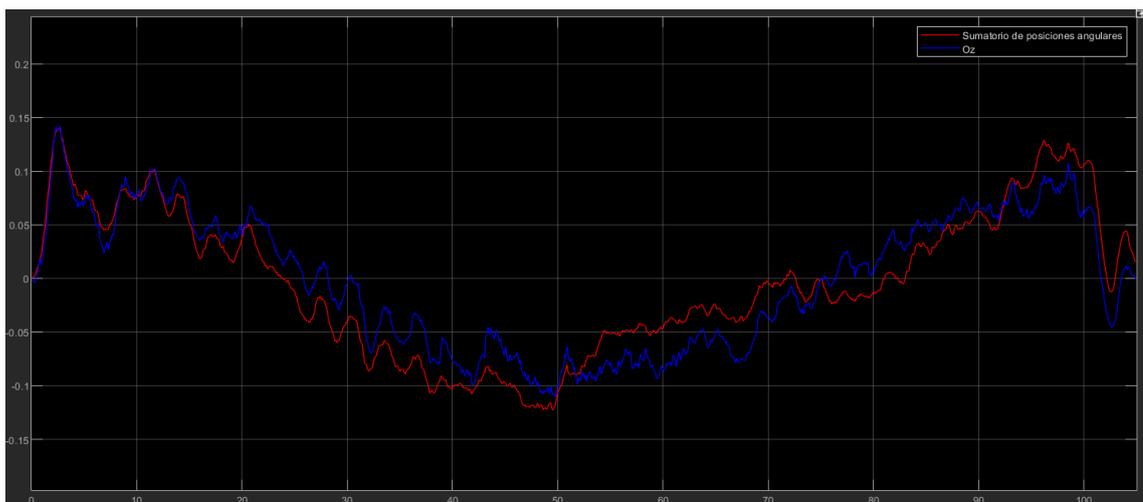


Figura 21. Sumatorio de los 3 encoders y rotación del eje Z con mucha fricción

Hay que tener en cuenta que estas gráficas se han obtenido con el giro en el eje Z ya controlado, por lo que el error observado en la gráfica es realmente pequeño y la diferencia entre el sumatorio del encoder de las 3 ruedas y la posición real es aún menor.

Si observamos filtro paso bajo antes de la saturación y aplicación de las acciones de control han demostrado eliminar los picos instantáneos, además de tener una visibilidad más clara de los resultados y un mejor comportamiento:

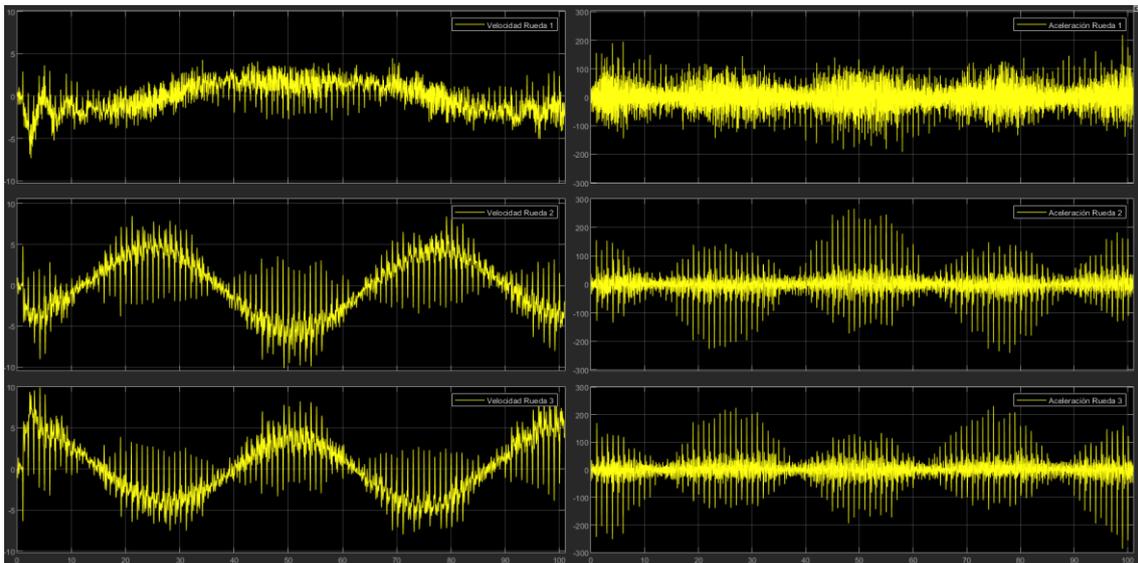


Figura 22. Velocidades y aceleraciones de la simulación de las 3 ruedas sin filtro

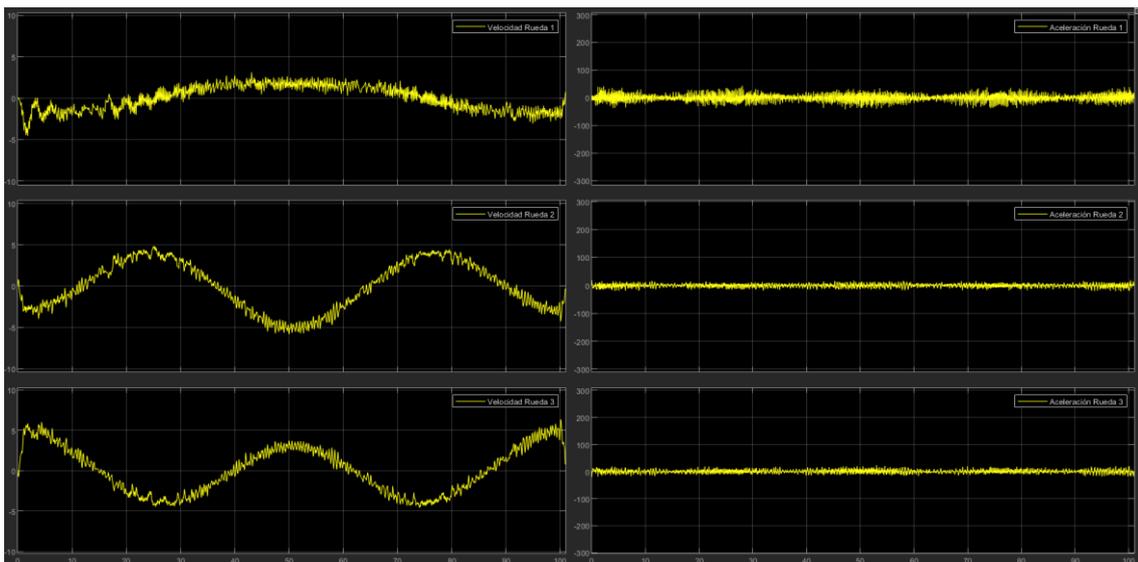


Figura 23. Velocidades y aceleraciones de la simulación de las 3 ruedas con filtro

Se puede observar como los picos instantáneos de las velocidades, gráficas de la izquierda, se han eliminado, cosa que aporta mayor realismo a la simulación, además de que las velocidades máximas necesarias para la trayectoria programada se reducen ligeramente y las oscilaciones en la velocidad son más lentas lo que indica que el sistema es más estable.

En cuanto a los picos de aceleración se han reducido a menos de la quinta parte y los picos instantáneos han desaparecido, los requerimientos de aceleración constante también se han reducido en gran medida. Toda esta reducción de aceleraciones implica



una reducción de la corriente consumida y aumenta la suavidad del sistema a la hora de realizar correcciones por lo que es más estable.

4. IMPLEMENTACIÓN REAL

Para la implementación real de la simulación es necesario el montaje de los componentes mecánicos y los componentes electrónicos necesarios, también hay que seleccionar o diseñar algunos de ellos ya que no todos aparecen en la simulación.

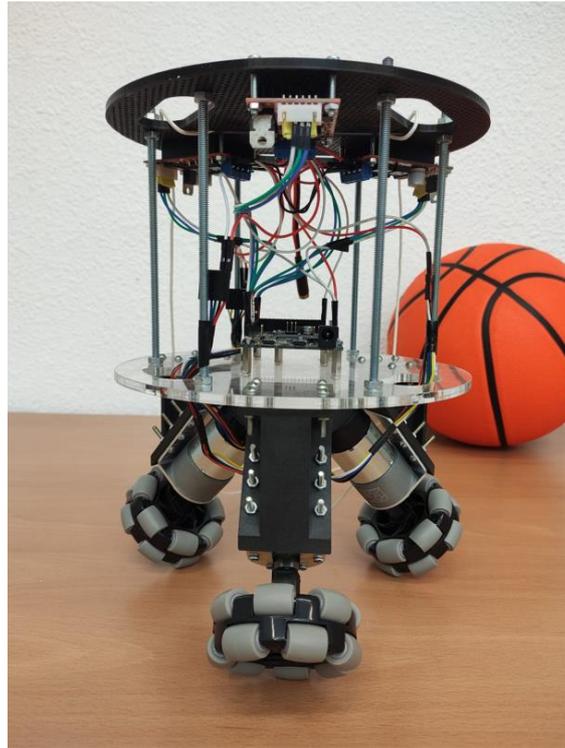


Figura 24. BallBot

El aspecto de la implementación real del BallBot, al igual que la simulación, es muy similar al BallIP desarrollado por el profesor Masaaki Kumagai de la universidad de Tohoku Gakuin, esto se debe a su reducido tamaño, facilidad de montaje y una cantidad reducida de partes que fabricar.

4.1. COMPONENTES MECÁNICOS

En el pasado ya se ha intentado la realización de un BallBot funcional en la universidad por lo que se pueden reutilizar algunos de los componentes mecánicos.

Se considerarán componentes mecánicos todas aquellas partes físicas que cumplan alguna función estructural o mecánica sin necesitar ni proporcionar ningún tipo

de señal eléctrica o impulso eléctrico. Dentro de esta categoría se engloban los siguientes componentes:

4.1.1. ESFERA

La esfera debe ser de un cierto tamaño con el fin de elevar el centro de gravedad del sistema y hacer que sea ligeramente más estable, esta también debe tener un buen agarre, una superficie lo más lisa posible y aguantar el peso del BallBot.

Se ha elegido una pelota de baloncesto tamaño 7, ya que esta cumple todos los requisitos anteriormente mencionados.



Figura 25. Pelota de baloncesto utilizada como esfera

La pelota de baloncesto no es una esfera perfecta, si bien la superficie es bastante lisa las líneas negras tienen cierta profundidad e introducen ligeras perturbaciones al BallBot. Por otra parte aunque es cierto que tiene buen agarre, este podría ser algo mejor, ya que en ocasiones las ruedas omnidireccionales pueden llegar a derrapar dependiendo del peso del BallBot y la acción de control de los motores.

4.1.2. RUEDAS OMNIDIRECCIONALES

Las ruedas omnidireccionales necesarias cuentan con rodillos que giran libremente para permitir que estas deslicen en la dirección perpendicular a la que ruedan.



Figura 26. Ruedas omnidireccionales

Las ruedas seleccionadas cuentan con un cuerpo de plástico, ya que no se necesita una gran resistencia y de esta manera las ruedas son más ligeras, mientras que los rodillos son de goma para mejorar el agarre que proporcionan.

4.1.3. UNIONES ENTRE LOS MOTORES Y LAS RUEDAS

Para unir los motores a las ruedas hace falta un acople ya que no tiene la misma forma el eje del motor que el agujero de las ruedas para el eje. Debido a esto se diseñó un acople directo del motor a la rueda pero la primera versión se rompió con facilidad, por lo que se procedió a suavizar algunos bordes interiores, que servían como concentrador de tensiones, y se ha ensanchado la pieza para aumentar su resistencia. Por último se hizo una tercera versión para ajustar a la décima de milímetro el tamaño del agujero donde se introduce el eje del motor y el eje que se introduce en la rueda se lijó

hasta conseguir un acople que también fija la rueda al motor sin la necesidad de adhesivos o algún otro tipo de soporte. El acople fue impreso en 3D con PLA.

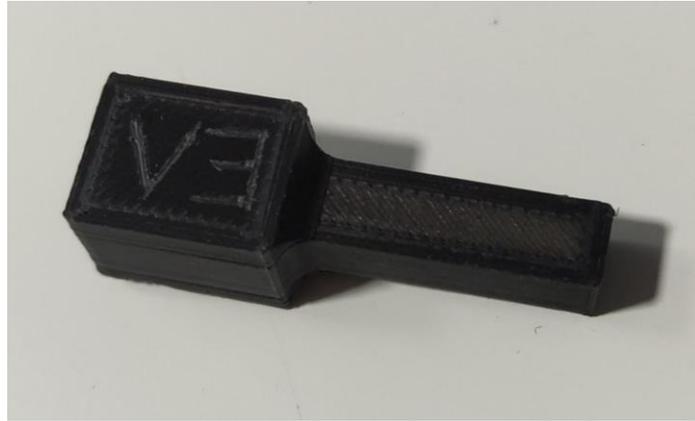


Figura 27. Acople V3 de los motores a las ruedas

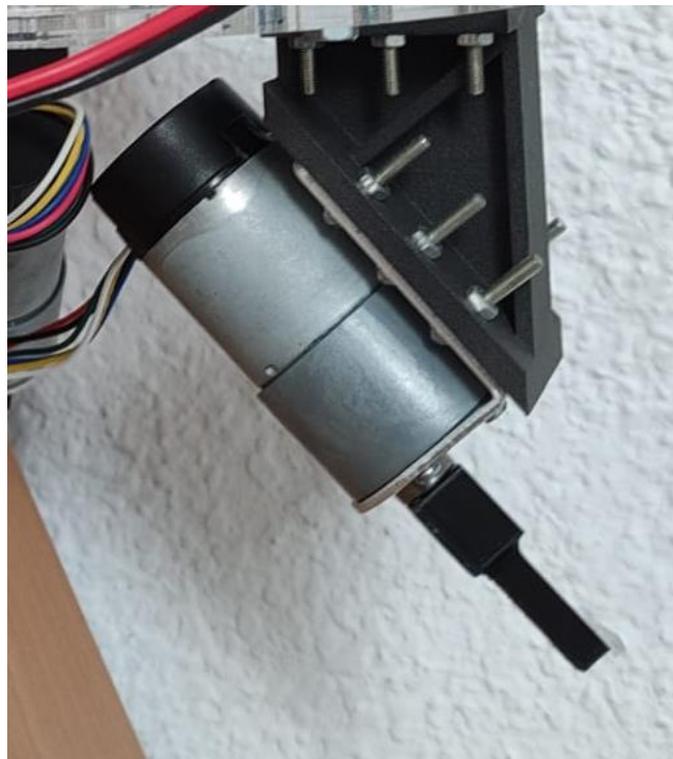


Figura 28. Acople V3 colocado en el motor

Se puede observar como el acople V3 encaja perfectamente con el eje del motor y está suficientemente ajustado como para que no caiga el acople ni estando hacia abajo, añadiéndole el peso de las ruedas y agitando el BallBot verticalmente.

4.1.4. SOPORTES DE LOS MOTORES

Puesto que ya había unos soportes para un BallBot (3) diseñados para una esfera del mismo tamaño al utilizado en este proyecto y cumple con las características necesarias se ha decidido reutilizarlos. Son soportes ligeros, resistentes y encajan tanto con el tamaño del BallBot como el de la esfera seleccionada.

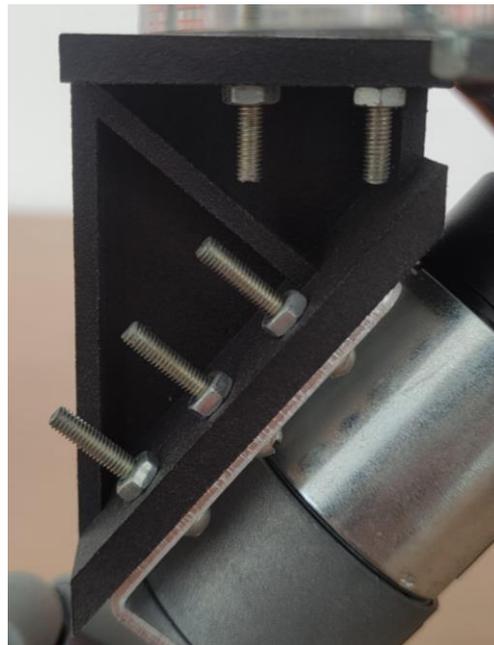


Figura 29. Soporte de los motores

A la hora de sujetar el motor también se han utilizado unas bridas (5) diseñadas específicamente para estos motores, aportadas por el mismo fabricante.

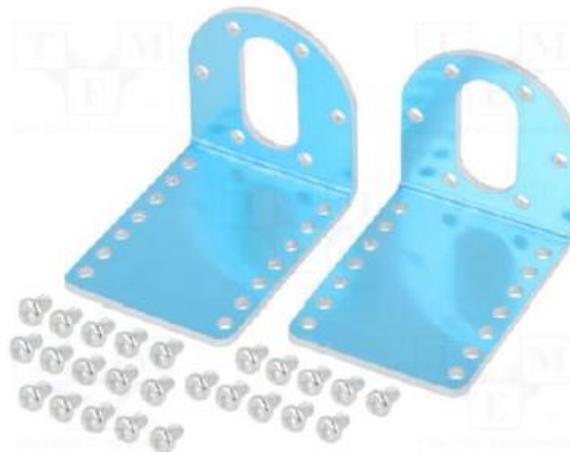


Figura 30. Bridas de los motores

4.1.5. PLATAFORMA SUPERIOR

En el caso de la plataforma superior esta se ha diseñado para albergar los 3 controladores de los motores, las 6 varillas roscadas encargadas de sostener dicha plataforma, la IMU lo más centrada posible y algunos agujeros para pasar los cables y aportar algo de rigidez, ya que las paredes de los agujeros son más resistentes que las partes centrales de la placa, ya que esta plataforma será impresa en 3D con un relleno muy inferior al 100% y sin capas lisas arriba o abajo.

La razón principal de no tener tapas y se pueda ver el patrón interno de relleno es facilitar la refrigeración de los controladores del motor, ya que estos tienen su disipador de calor pegados a la placa. Debido a que esta plataforma no necesita soportar mucho peso la resistencia del diseño es más que suficiente.

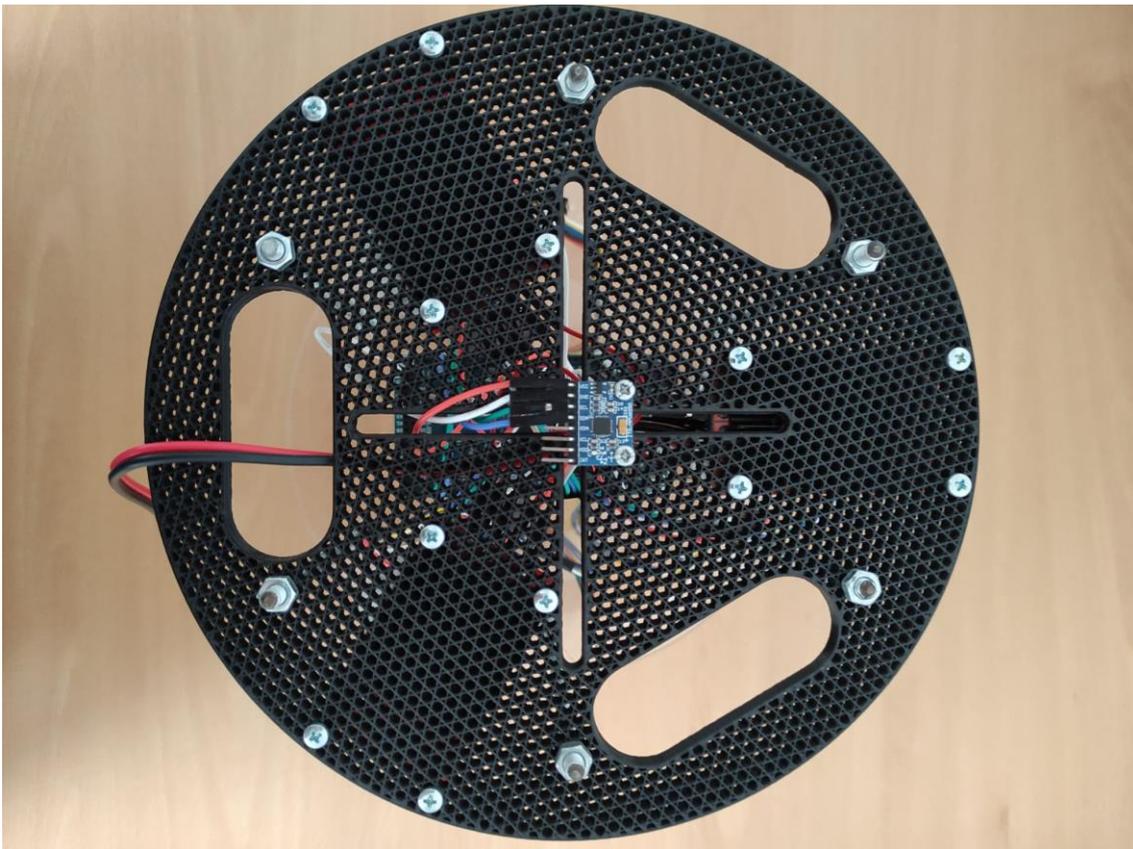


Figura 31. Plataforma superior

La colocación de los controladores de los motores y los agujeros está distribuidos con especial cuidado para que el peso esté centrado y eso de esa manera no se desestabilice el BallBot.

4.1.6. PLATAFORMA INFERIOR

Para la plataforma inferior se reutilizó la utilizada en el anterior proyecto de BallBot (3) ya que tiene los agujeros necesarios para pasar los cables y atornillar el microcontrolador, los soportes de los motores y las varillas roscadas que soportan la placa superior.

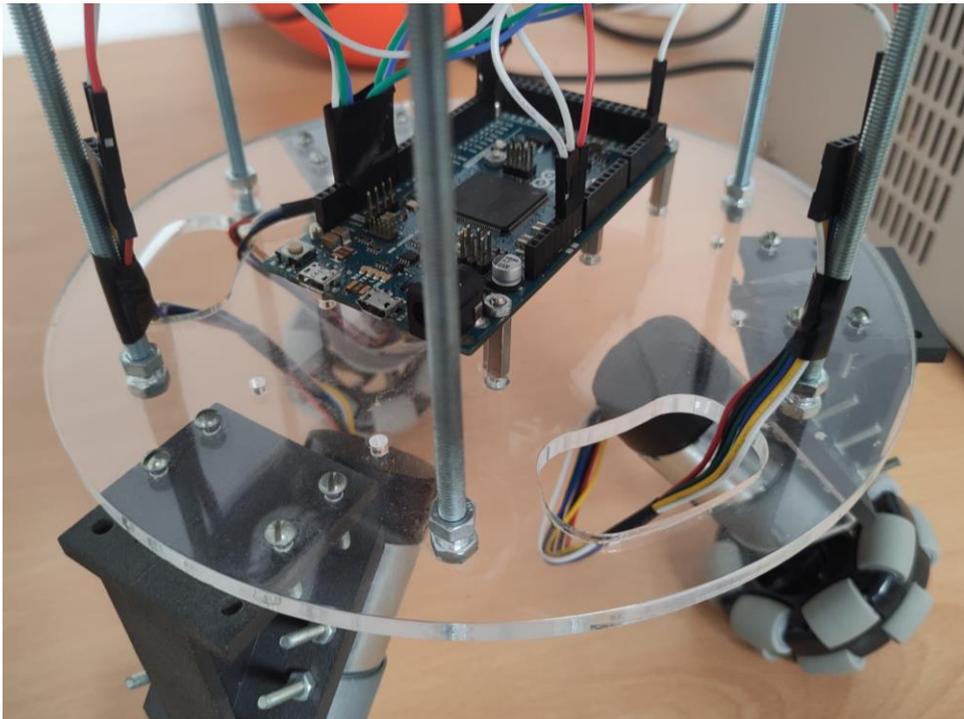


Figura 32. Plataforma inferior

4.2. COMPONENTES ELÉCTRICOS Y ELECTRÓNICOS

Se considerarán dentro de esta categoría todos aquellos componentes que necesiten o proporcionen algún tipo de señal eléctrica o impulso eléctrico, independientemente de su potencia o función.

4.2.1. MOTORES

El motor utilizado será un Pololu de 12V y teniendo en cuenta las conclusiones del proyecto de O. García (3) se ha seleccionado con una reductora 30:1, ya que aunque la reductora 70:1 debería ser suficiente para alcanzar la velocidad máxima necesitada en la simulación teniendo en cuenta que el caso real será más inestable precisará más velocidad.



Figura 33. Motor Pololu

Cada uno de los cables del motor corresponden a lo siguiente:

Color	Function
Red	motor power (connects to one motor terminal)
Black	motor power (connects to the other motor terminal)
Green	encoder GND
Blue	encoder Vcc (3.5 – 20 V)
Yellow	encoder A output
White	encoder B output

(4)

4.2.2. CONTROLADORES DE LOS MOTORES

Para controlar los motores se han utilizado 3 MZX MSDN470 (6) ya que no aumenta la zona muerta del motor y soporta intensidades muy superiores a las necesitadas por el BallBot. El único aspecto negativo es que cada uno de los controladores puede controlar solo uno de los motores.

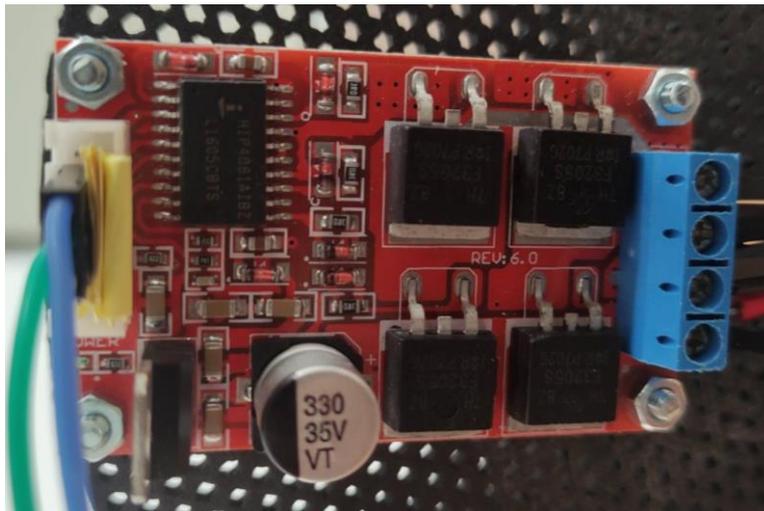


Figura 34. Controlador del motor

4.2.3. MICROCONTROLADOR

Por disponibilidad de la universidad, número de puertos, funcionalidades y puesto que Simulink incluye una librería específica para trabajar con placas de Arduino se ha seleccionado Arduino DUE como microcontrolador.



Figura 35. Arduino Due

4.2.4. IMU

Necesitamos una Inertial Measurement Unit (IMU) para conocer los ángulos de inclinación del BallBot. Se ha seleccionado la MPU6050 debido a su reducido tamaño, disponibilidad y que tiene todo lo suficiente para poder calcular el ángulo de inclinación real del BallBot en los X e Y.

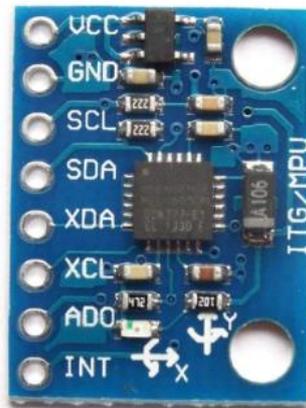


Figura 36. IMU MPU6050

4.2.5. FUENTE DE ALIMENTACIÓN

Para asegurar que tenemos suficiente corriente pico y poder ver cuanta corriente circula por el BallBot se ha utilizado una fuente de alimentación durante las pruebas. Con un cable suficientemente largo puede sustituir a las baterías que necesitaría el BallBot, además no necesitamos cargarlas y no se agotan.

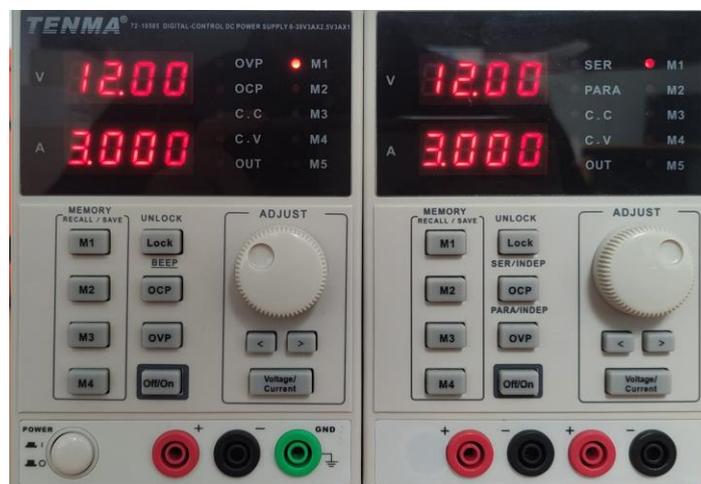


Figura 37. Fuente de alimentación

4.3. MONTAJE DEL BALLBOT Y VARIACIONES

El montaje se ha realizado por partes después de comprobar que cada uno de los componentes funciona y todo encajaba correctamente. Una vez montado se han ido realizando distintas variaciones para solucionar problemas que han ido surgiendo.

4.3.1. CAMBIO EN LOS ACOPLEROS DE LOS MOTORES

Los acoples V3 diseñados para este proyecto son más cortos que los utilizados en el proyecto de O. García (3), a raíz de esto la rueda omnidireccional no hace perfecto contacto con la esfera. Esto provoca que al pasar de estar apoyado en un rodillo al siguiente no lo haga de manera suave y esto contribuye a la desestabilización del BallBot.

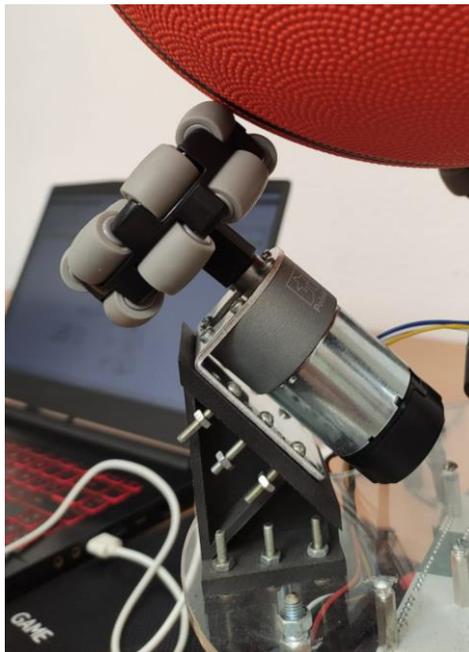


Figura 38. Posición original del soporte

Para solucionar este problema se ha desplazado hacia fuera los soportes de los motores, pasando de estar atornillados a la plataforma inferior con 6 tornillos a estar atornillados con 4. Al hacer esto se aumenta el espacio entre las ruedas y se consigue que estas vuelvan a ser perfectamente perpendiculares a la superficie de contacto con la esfera.



Figura 39. Posición modificada del soporte

Una vez adoptada la solución, la transición entre rodillos cuando giran las ruedas vuelven a ser imperceptibles y por tanto ya no desestabiliza lo más mínimo al BallBot.

4.3.2. PESO EXTRA

Al trabajar con el BallBot se puede observar como las ruedas patinan con relativa facilidad y las aceleraciones rápidas hacen que el BallBot gire sobre sí mismo y caiga de la esfera. Debido a estos problemas probó a añadir sacos de medio Kg de arena.



Figura 40. Saco de peso extra

Al principio se probó con 3 sacos pero el BallBot no era capaz de moverlo por lo que se bajó a 2 y aunque sí se movía, este movimiento era muy lento y el ángulo máximo en el que el BallBot puede recuperar su posición de equilibrio se reducía mucho.

Al colocar un único saco fijado a la plataforma superior los problemas por los que se decidió añadir el peso desaparecieron, el BallBot era capaz de moverse con facilidad y su ángulo máximo no se reducía. El hecho de colocar el peso en la parte de arriba del BallBot sube el centro de masas del sistema que forman el BallBot y la esfera, haciendo que tarde algo más de tiempo en superar el ángulo de inclinación máximo.

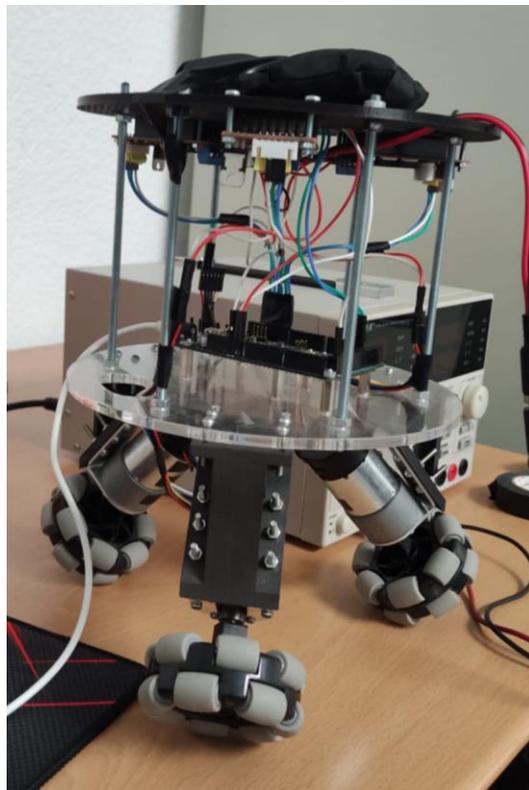


Figura 41. BallBot con peso extra

4.3.3. CAMBIO DE ESFERA

Con el fin de aumentar el tiempo que tiene el BallBot para responder se ha probado a sustituir la esfera por una más grande, en este caso la única esfera más grande, con un buen agarre y fácil transporte hasta el laboratorio es una pelota de yoga.

Esta pelota es bastante lisa y en principio cumple con todos los requisitos que tiene la esfera de un BallBot.

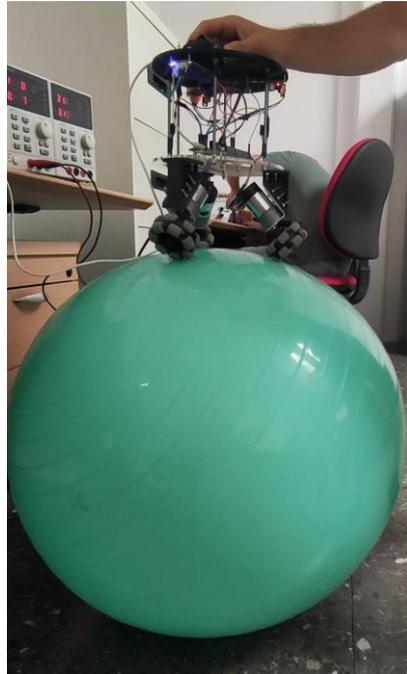


Figura 42. BallBot sobre la pelota de yoga

Al hacer las pruebas de estabilidad del BallBot las correcciones provocaban que las ruedas se hundiesen en la pelota ligeramente y rebotasen, generando grandes inestabilidades e incluso llegando a hacer rebotar el BallBot y que sus ruedas se despeguen de la esfera. Por este motivo la pelota de yoga fue descartada.

4.3.4. SISTEMA ANTICAIDA

En principio como sistema para evitar que el BallBot se cayese al suelo, con el consiguiente peligro de rotura de las piezas, se utilizó una cuerda enganchada a la plataforma superior del BallBot.

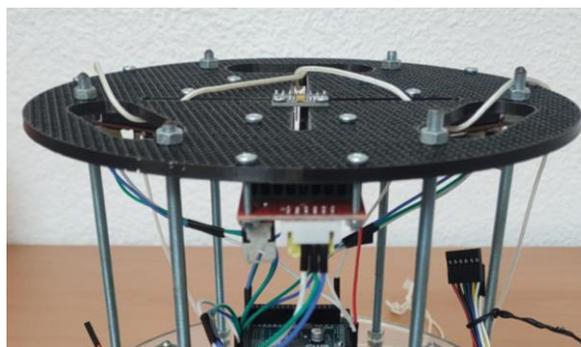


Figura 43. Sistema anticaida original

Esta solución hace que con únicamente sujetar la cuerda se puedan evitar una gran cantidad de daños. Para evitar que los tirones de la cuerda dañen la plataforma superior se debe mantener cierta tensión, así además se evitan posibles enredos de la cuerda con otros componentes.

El hecho de que la cuerda esté constantemente con algo de tensión hace que le quitemos peso al BallBot, disminuyendo el efecto del peso extra. Además el peso que le quitamos no es siempre el mismo puesto que el BallBot está en constante movimiento y no podemos seguir su movimiento a la perfección con la cuerda, por tanto modificamos constantemente su comportamiento. Puesto que el cambiar el comportamiento del BallBot no es algo deseable se acabó eliminando la cuerda y siguiéndolo muy de cerca con las manos y estando atento para cogerlo antes de que caiga al suelo, si bien es algo más arriesgado se elimina por completo la influencia en el sistema.

4.4. ESQUEMA DE CONTROL

Los esquemas de control se realizarán en Simulink con la ayuda de la librería de Hardware de arduino.

4.4.1. PRUEBA DE LOS MOTORES Y LA IMU

El primer paso es comprobar que tanto los motores como la IMU funcionan correctamente y ver las direcciones en las que giran cada uno de los motores con sus dos canales pwm para darles el signo que les correspondan. También se observan los valores de aceleración en los ejes X e Y mientras el BallBot se encuentra perfectamente vertical para corregir ese error.

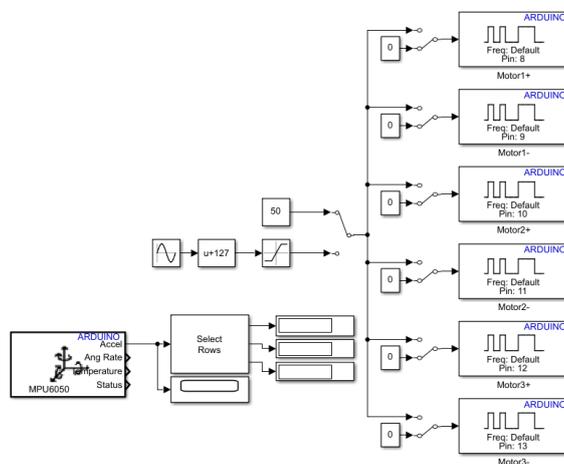


Figura 44. Esquema de prueba de los motores y la IMU

4.4.2. ESQUEMA DE ESTABILIDAD ESTANDAR

Para diseñar el esquema de estabilidad estándar se divide en 3 partes, la primera de las partes consiste corregir el valor de las aceleraciones en los ejes X e Y, filtrar las señales para eliminar el ruido y aplicarles un control proporcional derivativo.

Seguidamente se observa el signo de cada eje para ver por cuál de los canales pwm hay que enviar la señal de control, se aplican las formulas calculadas teóricamente y se vuelve a comprobar la dirección de giro de cada uno de los motores.

Por último si el valor de la señal pwm de un canal es mayor de 0, se le suma un valor que sirve para compensar la zona muerta de los motores, se le aplica una saturación al valor y se envía a los motores.

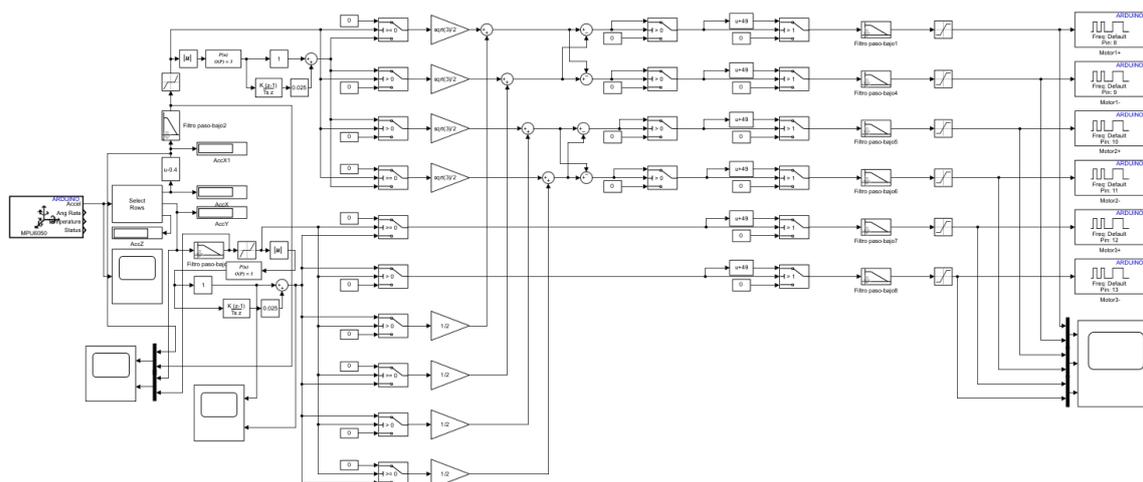


Figura 45. Esquema de estabilidad

Solo con este primer esquema de control el BallBot reacciona en las direcciones correctas y de la manera esperada hasta el momento de colocarlo encima de la esfera, al hacer esto el BallBot detecta las aceleraciones producidas por los motores como inclinación haciendo que sus propias respuestas lo hagan inestable.

4.4.3. COMPENSACIÓN CON EJE Z

Con el fin de eliminar el problema de la interferencia entre la aceleración producida por los motores y la producida por la inclinación real del BallBot se probó a utilizar la aceleración medida en el eje Z, ya que este es perpendicular a la aceleración

producida en los ejes X e Y, para saber cómo de inclinado está el sistema realmente y modular el valor de la respuesta en función de esto.

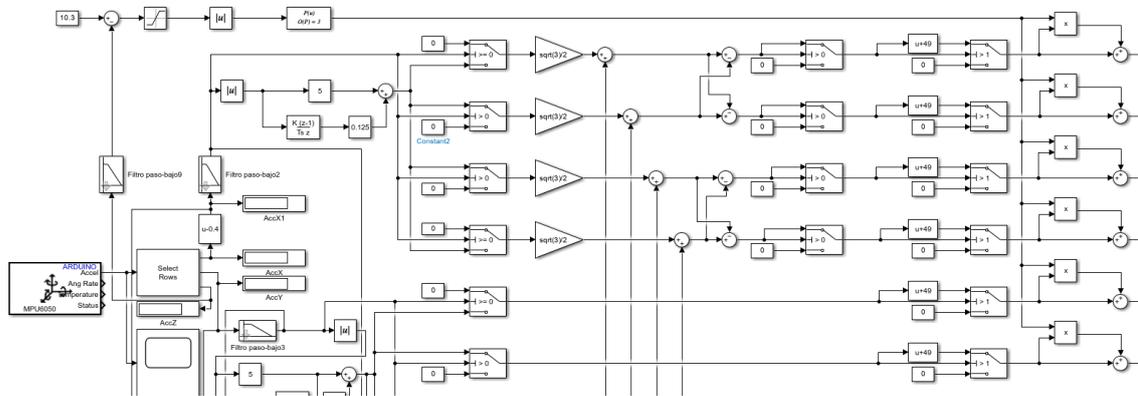


Figura 46. Esquema de estabilidad con compensación del eje Z

La parte nueva de compensación con el eje Z es la línea superior y las multiplicaciones de la parte derecha.

Se mide el valor de la aceleración filtrado y este se resta a 10.3, que es el valor con el BallBot totalmente vertical, después se le aplica una saturación y un valor absoluto. Por último este valor se multiplica por la señal pwm calculada y se le suma el resultado a la señal original para cada rueda.

El hecho de que al inclinarse el BallBot se desplace ligeramente en el eje Z hace que la propia compensación tenga el mismo tipo de error que la señal original, eso sí algo menor. Debido a esto la compensación no se nota mucho en el comportamiento, sigue siendo inestable aunque haya mejorado ligeramente.

4.4.4. FILTRO COMPLEMENTARIO

El filtro complementario se basa en combinar el ángulo calculado con los acelerómetros, el cual es susceptible a los movimientos del MPU, y el ángulo calculado con el giroscopio, el cual va acumulando error. Si se combina correctamente el resultado del acelerómetro tiene un filtro paso bajo, amortiguando en gran medida los movimientos bruscos, y el calculado por el giroscopio tiene un filtro paso alto, eliminando el drift. (2)

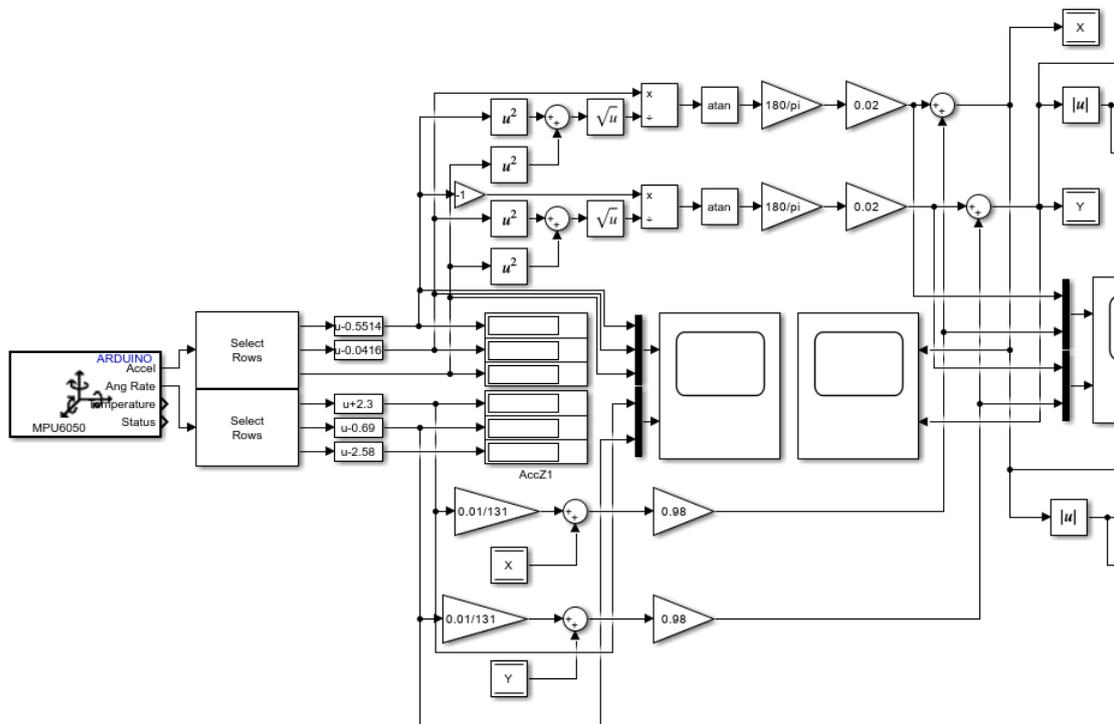


Figura 47. Esquema de estabilidad con filtro complementario

La parte superior es la correspondiente al acelerómetro y la inferior la del giroscopio, una vez calculadas cada una se multiplica por un factor y se suman para formar la medida final, este valor se ve muy poco afectado por los movimientos bruscos y no introduce drift al sistema por lo que soluciona el mayor problema que presentaba el control sin empeorar otros aspectos.

Una vez implementado este control el comportamiento del BallBot mejora mucho y ya no se desestabiliza con sus propias reacciones.

4.4.5. FILTRO COMPLEMENTARIO CON GANANCIA POLINOMIAL

En vista de que las reacciones del BallBot son mucho más fuertes cuanto más se aleja del punto de equilibrio y algo más débil de lo esperado cerca de este se ha optado por añadir una ganancia polinomial antes de los controladores. Para ver la progresión de la respuesta con la ganancia polinomial se hizo un esquema de prueba con una señal de rampa y varias pruebas incluyendo una onda cosenoidal.

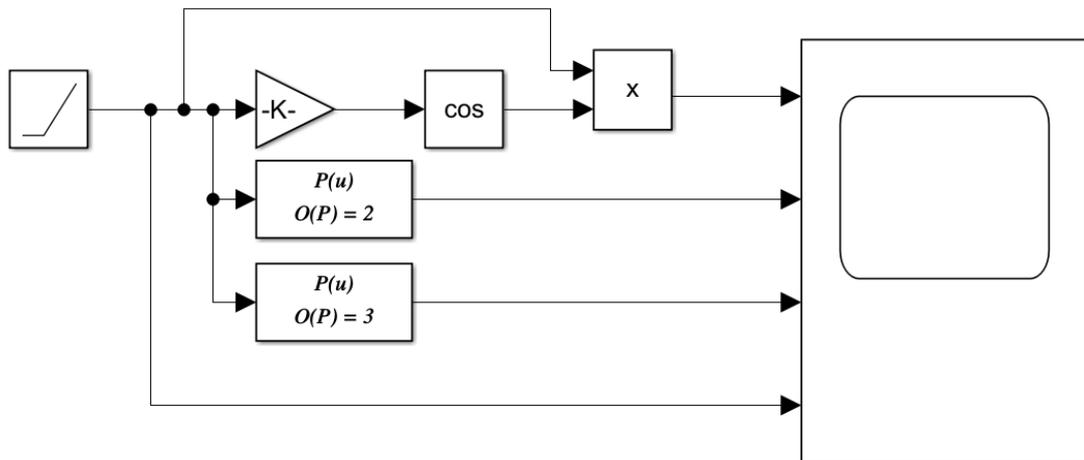


Figura 48. Esquema de pruebas de polinomios

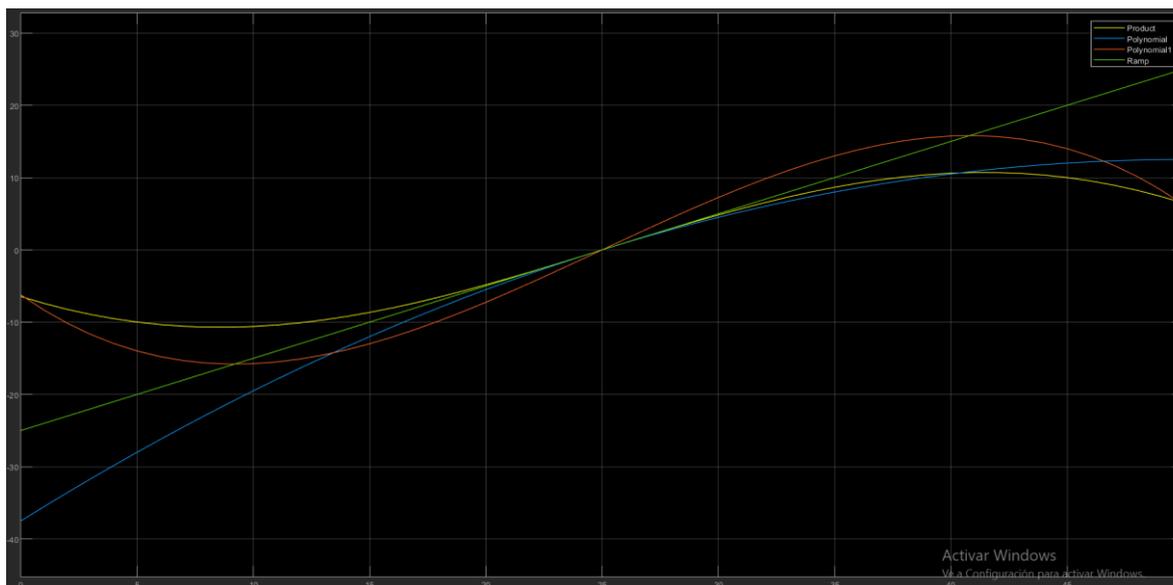


Figura 49. Resultado de pruebas de polinomios

La línea verde es la rampa de referencia, esta se ha incluido para poder comparar las demás ondas la original.

Entre las otras 3 señales, la cosenoidal es demasiado suave y no aumenta el valor cerca del punto de equilibrio, la función cuadrática no se comporta igual para los valores negativos, aunque esto se puede arreglar colocando la polinomial justo antes de la aplicación de la acción de control ya que esta es siempre positiva, y la función cúbica aumenta el valor de la señal cerca del punto de equilibrio y la reduce en los extremos.

Se acaba seleccionando la función cúbica ya que esta varía el comportamiento del sistema de la manera deseada.

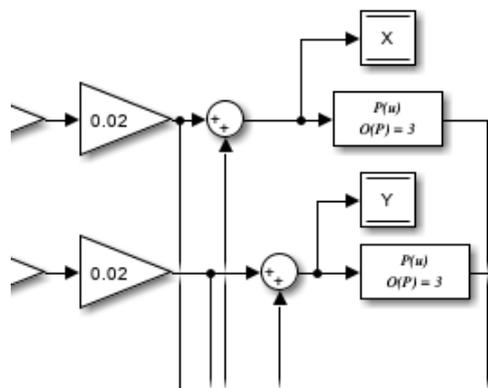


Figura 50. Esquema de estabilidad con filtro complementario y ganancia polinomial

El comportamiento real del BallBot se modifica de manera algo menos marcada pero si en la dirección esperada, por lo que mejora ligeramente.

4.5. RESULTADOS Y CONCLUSIONES

Entre los esquemas de control el que mejor comportamiento tiene es sin duda el que cuenta con un filtro complementario, bien sea con ganancia polinomial o sin ella.

Viendo las gráficas de las acciones de control del esquema de control final, estas son respuestas rápidas y adecuadas, el tipo de esquema de control seleccionado así como la magnitud de los parámetros del control PD parece la correcta. Los motores tienen velocidad y par suficientes y las ruedas no producen perturbaciones ni deslizan.

Desgraciadamente no se ha conseguido la estabilización del BallBot, esto se debe a que uno de los drivers de los motores se bloquea al cambiar de dirección durante algo más de 400 milisegundos, cuando el tiempo que tiene para reaccionar antes de superar el ángulo máximo son aproximadamente 340 milisegundos.

Si aportamos un ligero contacto para dar algo de estabilidad al sistema y que el BallBot tenga más tiempo para reaccionar es capaz de equilibrarse pero el hecho de que una de las ruedas se bloquee produce desequilibrios.



En vista de la respuesta de las otras dos ruedas y de que los demás componentes funcionan correctamente se puede concluir que el sistema podría estabilizarse sin problemas si la rueda que se bloquea funcionase correctamente.



5. PLIEGO DE CONDICIONES

Puesto que este se trata de un proyecto académico no incluye condiciones que hagan referencia a normativas específicas. Cada parte del proyecto cuenta con sus requisitos propios.

5.1. SIMULACIÓN

1. Todas las partes simuladas deberán ser lo más similares posibles a las que previsiblemente se utilizarán en la implementación real.
2. Se debe diseñar un control de estabilidad capaz de evitar que el BallBot caiga de la esfera y contrarrestar ligeras perturbaciones.
3. Se debe diseñar un control de posición que sea capaz de seguir trayectorias programadas.
4. La construcción del mundo del modelo de la simulación debe ser lo más cercano a la realidad posible, es decir, no asignar valores irreales a parámetros ni establecer relaciones incorrectas entre los elementos de dicho mundo y que los componentes presentes en la simulación tengan la misma libertad de movimiento que en el caso real.
5. El control del BallBot no se realizará con la medición de datos que no puedan ser obtenidos o que no sean mínimamente fiables en el mundo real.

5.2. IMPLEMENTACIÓN REAL

1. Se debe diseñar e implementar un control de estabilidad que pueda mantener al BallBot sobre la esfera sin caer.
2. Para reducir los costes se reutilizarán los componentes disponibles que pertenezcan a la universidad siempre que estos cumplan los requisitos.
3. Con el fin de asegurar una rápida ejecución del esquema de control, este deberá ser lo más sencillo posible.



4. En cualquier esquema de control antes de la aplicación de algún tipo de acción de control se asegurará que esta se encuentra en el rango permitido por el componente que vaya a recibirla.



6. PRESUPUESTO

Para la realización de este presupuesto se tomará el precio de venta de cada uno de los componentes comprados, el precio de elaboración de las piezas diseñadas y el precio que tenía en su momento cada uno de los elementos reutilizados. Tanto la mano de obra propia como el diseño y la programación no se han incluido en este presupuesto.

6.1. COMPONENTES DISEÑADOS

En cuanto al precio de los componentes diseñados, ya que estos han sido impresos en 3D, se tomará como referencia los precios aproximados que aporta el software de impresión 3D una vez se ha introducido tanto el modelo como las características de impresión.

En el caso de la plataforma superior esta ha sido diseñada sin un acabado liso en sus caras por lo que se ahorra gran cantidad de material y tiempo de impresión. Adicionalmente a esto el patrón interno seleccionado para imprimir la pieza permite reducir el porcentaje de relleno, lo que también ahorra materia. Los costes estimados son de 1,40€ por el material, 0,17€ por la electricidad consumida durante el tiempo de impresión y 0,06€ de amortización de la máquina, por lo que el coste total de impresión de la plataforma superior asciende a 1,63€.

Para los acoples se contará el precio de las 3 versiones, ya que estas han sido necesarias para llegar al diseño final. En la tabla resumen de precios el coste de las dos primeras versiones se incluirán en el de los acoples V3 utilizados. La primera versión del acople tenía un coste de 0,02€ por el material, 0,02€ de electricidad y 0,01€ de amortización de la impresora, por lo que su coste unitario es de 0,06€, este es igual al total, ya que únicamente se fabricó una pieza. El coste unitario del tercer modelo es de 0,05€ de material, 0,03€ de electricidad y 0,02€ de amortización de la impresora, haciendo un total 0,10€ por unidad. En cuanto al modelo 2 apenas tiene pocas décimas de milímetro de diferencia con la tercera versión por lo que sus costes de fabricación son iguales. Teniendo en cuenta que se imprimieron tres piezas del modelo 3 y una única pieza de los modelos 1 y 2 los costes totales de impresión de los acoples del motor a las ruedas es de 0,45€. Teniendo en cuenta el precio total, incluido el del desarrollo, podemos suponer un precio de 0,15€ por cada acople.



6.2. TOTAL

Categoría	Producto	Precio unitario	Unidades	Precio total
Componentes mecánicos	Pelota de baloncesto	5,78	1	5,78
	Ruedas omnidireccionales	19,77	3	59,31
	Unión V3	0,15	3	0,45
	Soportes y bridas	21,13	3	63,39
	Plataforma superior	1,63	1	1,63
	Plataforma inferior	17,81	1	17,81
	Tornillería	-	-	8,85
Componentes electrónicos	Motor	23,14	3	69,42
	Controlador del motor	21,68	3	65,04
	Arduino DUE	29,24	1	29,24
	IMU MPU6050	3,48	1	3,48
	Cables	-	-	0,38
Total sin IVA				324,78
IVA (21%)				68,20
Total				392,98

Los precios de la tabla se encuentran en euros.



7. CONCLUSIONES Y POSIBLES MEJORAS

En referencia al apartado de la simulación podemos concluir que el BallBot tiene un funcionamiento muy correcto, se ha conseguido que tenga respuestas suaves y precisas, que el sistema sea muy estable y que siga trayectorias con un error bajo. Adicionalmente se ha desarrollado un nuevo modo de medición del ángulo que ha rotado el BallBot en el eje Z utilizando la suma de los 3 encoders de las ruedas, el cual si no hay deslizamiento cuenta con un error muy bajo y fácilmente corregible. Por todo ello podemos considerar que se han cumplido con creces todos los objetivos de este apartado.

El montaje de la implementación real, así como el diseño de las piezas necesarias, también se ha dado de manera satisfactoria. Las primeras prueba reales parecían prometedoras, ya que el BallBot reaccionaba de manera proporcionada y en la dirección correcta, una vez se aplicó el filtro complementario para eliminar los problemas generados por la naturaleza de los acelerómetros y los giroscopios, parecía que únicamente faltaba acabar de ajustar el controlador para que el sistema fuese capaz de estabilizarse por sí mismo. Después de varios días sin entender por qué seguía necesitando un leve contacto para mantenerse estable y probar un gran número de cambios y posibles soluciones, observé al levantar el robot como una de las ruedas se bloqueaba al cambiar de dirección. Las otras dos ruedas funcionaban correctamente por lo que el problema no era el tiempo de ejecución del esquema de control ni un problema con el microcontrolador, revisé las señales enviadas a los drivers de las ruedas y eran correctas. En los movimientos que únicamente implicaban las dos ruedas que funcionan correctamente el BallBot sí que era capaz de recuperar la estabilidad, pero en el momento que la otra rueda entraba en acción hacía que todo el sistema se descompensase. Considerando todo esto seguramente solo con cambiar el driver que no funciona correctamente sería suficiente para que el BallBot se mantuviese estable de manera autónoma sin ningún tipo de intervención externa.

Para el posicionamiento de la implementación real del BallBot se podría extraer una estimación bastante precisa operando con los valores de los encoders de los motores. Los motores 2 y 3 únicamente tienen comportamientos distintos al moverse en el eje X, por lo que la diferencia de sus encoders se debe al desplazamiento en este eje. El motor 1 solo influye en el desplazamiento en giro en Z y el desplazamiento en Y, puesto que ya sabemos cómo calcular el giro en Z con los encoders de los 3 motores solo hay que sumar, ya que van en direcciones opuestas, este valor al del encoder del motor 1 para obtener el desplazamiento en el eje Y. Para que estos supuestos se cumplan debería no haber deslizamiento por lo es probable que se vaya acumulando un ligero de error de posición después de varios movimientos.



La selección de componentes parece la correcta ya que todos los componentes que no tienen fallos de funcionamiento cumplen con su función. La reductora de 30 de los motores funciona correctamente y tiene una velocidad más que suficiente, se podría utilizar motores con una reductora de 50 de la misma marca para obtener algo más de par, ya que la velocidad seguiría siendo suficiente.

Además de añadir el control de posición y cumplir con los objetivos de este proyecto otras posibles mejoras o futuras líneas de trabajo serían:

- Elevar el centro de gravedad del sistema daría algo más de tiempo a este para reaccionar, haciéndolo más estable.
- Dotar al BallBot de un sensor para poder seguir líneas en el suelo y que decida seguir unas u otras dependiendo del destino deseado. Esto puede ser muy útil en un entorno industrial controlado, además al tener las líneas del suelo de referencia se eliminan los errores de posición acumulados sin utilizar sistemas de detección externos al BallBot.
- Añadir sensores para la detección de obstáculos con el fin de evitar colisiones y que el BallBot pueda circular por entornos con personas.
- Si se le añade una cámara y un control remoto podría servir como cámara de seguridad móvil.
- Se le podrían añadir protecciones para evitar dañar el BallBot de posibles golpes y caídas.
- Con un microcontrolador más rápido se podría bajar el tiempo de ejecución y hacer que tenga un comportamiento más suave.
- Dotar al BallBot de motores más potentes, una bandeja para colocar los objetos a transportar y sensores para detectar tanto el peso de estos objetos como su distribución, con el fin de ajustar el control del BallBot y que mantenga un comportamiento muy similar al que tiene sin carga.

La realización de este trabajo ha sido una gran manera de unir los conocimientos adquiridos en el máster y profundizar en ellos, también me ha dado la oportunidad de ir más allá de problemas teóricos o prácticas guiadas y resolver dificultades reales, probando creativas soluciones, las cuales no siempre han funcionado pero si han contribuido a mejorar mi entendimiento de ciertas áreas.



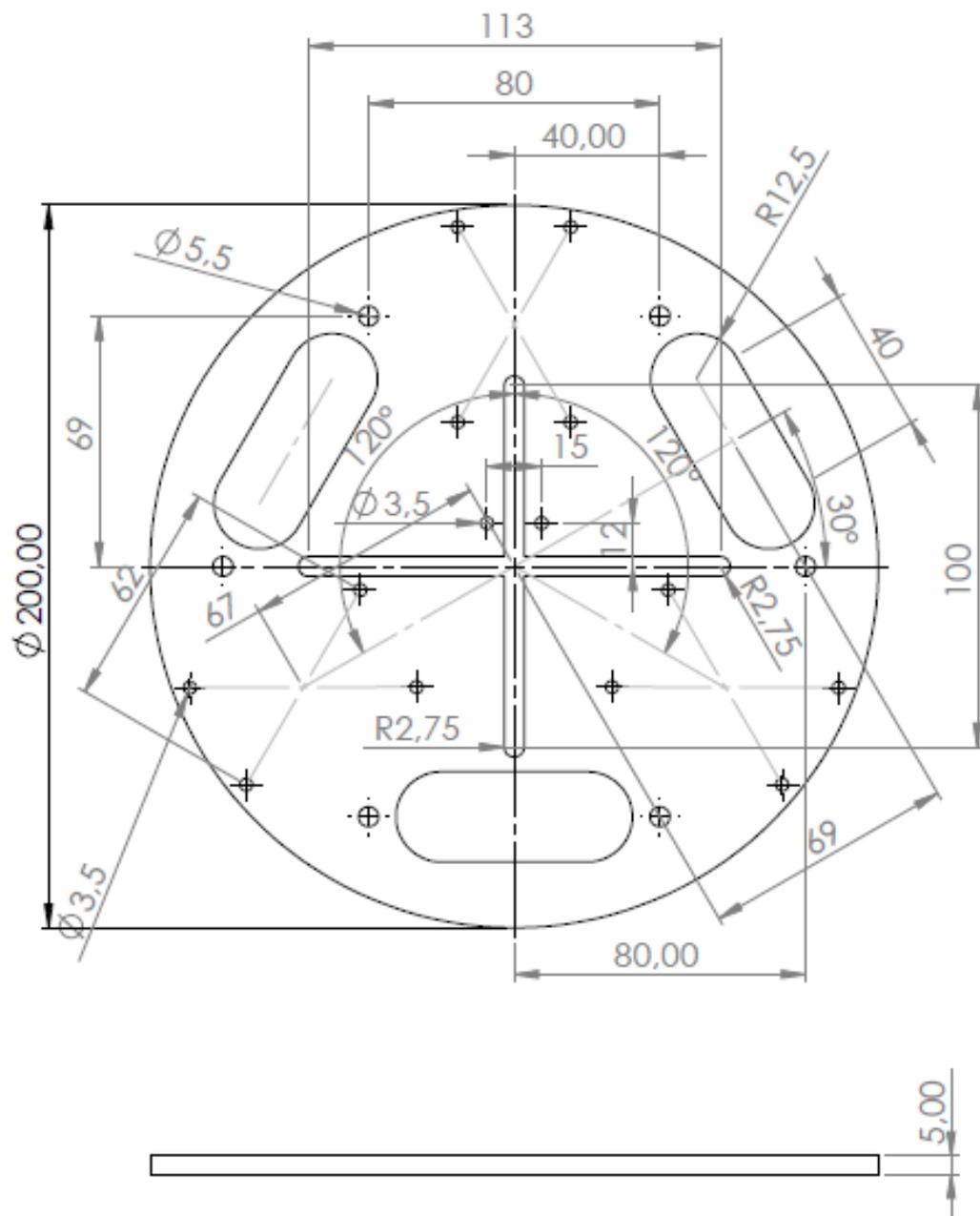
8. BIBLIOGRAFÍA

1. “Ballbot” <https://en.wikipedia.org/wiki/Ballbot>
2. “Tutorial MPU6050, Acelerómetro y giroscopio”
https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html
3. O. García Luengo, “Diseño, implementación y control de un prototipo de BallBot.” 2023.
4. “30:1 Metal Gearmotor 37Dx68L mm 12V with 64 CPR Encoder”
<https://www.pololu.com/product/4752>
5. “TEM Components. Brida de aluminio.” <https://www.tme.eu/es/details/pololu-1084/accesorios-para-micromotores/pololu/stamped-aluminum-l-bracket-pair-for-37d>
6. “MZX. MSDN470 Motor Controller.” https://www.alibaba.com/product-detail/Perfect-Anti-dead-zone-H-Bridge_60590319410.html



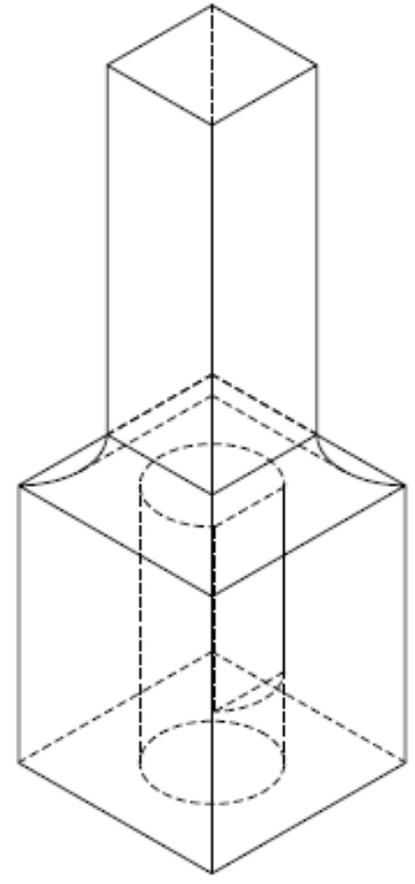
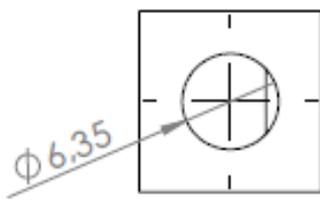
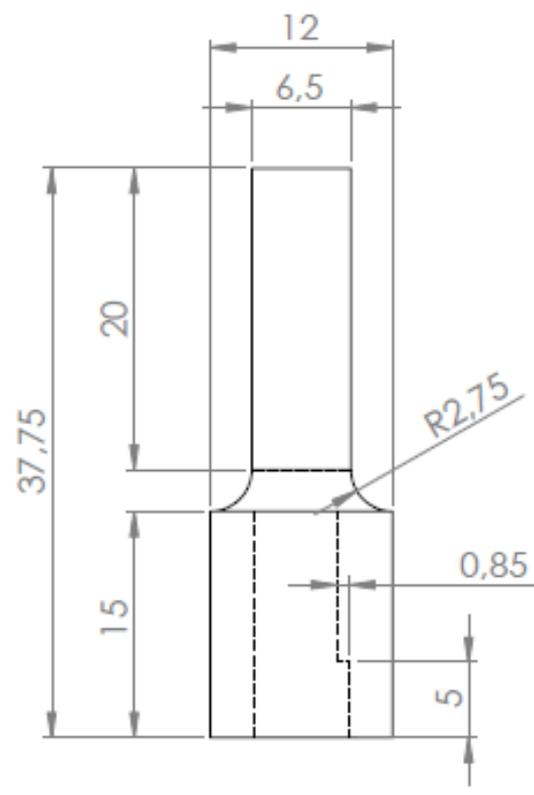
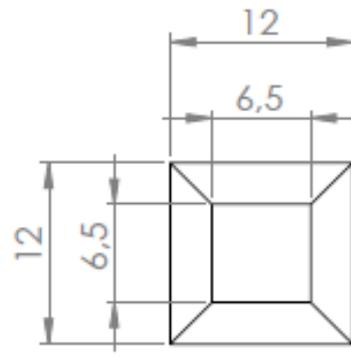
9. ANEXOS

9.1. PLANOS



UNDADES EN MM

	NOMBRE	FECHA	CANTIDAD:	UNIVERSIDAD POLITECNICA DE VALENCIA	
	MODELADO	ANTONIO MARCO ARNAL	14/05/24		1
	DIBUJADO	ANTONIO MARCO ARNAL	14/05/24		
	REVISADO	ANTONIO MARCO ARNAL	14/05/24		
	FABRICADO	ANTONIO MARCO ARNAL	14/05/24		
A	MATERIAL:	ESCALA:		Dibujos:	
	PLA	1:2		BASE SUPERIOR	



ESCALA 3D 3:1

UNIDADES EN MM

	NOMBRE	FECHA	CANTIDAD:	UNIVERSIDAD POLITECNICA DE VALENCIA
	ANTONIO MARCO ARNAL	08/05/24	3	
	ANTONIO MARCO ARNAL	08/05/24		
	ANTONIO MARCO ARNAL	08/05/24		
A	FABRICADO	ANTONIO MARCO ARNAL		08/05/24
	MATERIAL:	ESCALA:	Dibujo:	
	PLA	2:1	ACOPLE V3	



9.2. OBJETIVOS DE DESARROLLO SOSTENIBLE



ANEXO I. RELACION DEL TRABAJO CON LOS OBJETIVOS DE DESARROLLO SOSTENIBLE DE LA AGENDA 2030

Anexo al Trabajo de Fin de Grado y Trabajo de Fin de Máster: Relación del trabajo con los Objetivos de Desarrollo Sostenible de la agenda 2030.

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

Objetivos de Desarrollo Sostenibles	Alto	Medio	Bajo	No Procede
ODS 1. Fin de la pobreza.				X
ODS 2. Hambre cero.				X
ODS 3. Salud y bienestar.		X		
ODS 4. Educación de calidad.				X
ODS 5. Igualdad de género.				X
ODS 6. Agua limpia y saneamiento.				X
ODS 7. Energía asequible y no contaminante.				X
ODS 8. Trabajo decente y crecimiento económico.		X		
ODS 9. Industria, innovación e infraestructuras.	X			
ODS 10. Reducción de las desigualdades.				X
ODS 11. Ciudades y comunidades sostenibles.			X	
ODS 12. Producción y consumo responsables.				X
ODS 13. Acción por el clima.				X
ODS 14. Vida submarina.				X
ODS 15. Vida de ecosistemas terrestres.				X
ODS 16. Paz, justicia e instituciones sólidas.				X
ODS 17. Alianzas para lograr objetivos.				X

Descripción de la alineación del TFG/TFM con los ODS con un grado de relación más alto.

***Utilice tantas páginas como sea necesario.

Se relaciona con el ODS 9 en un alto grado ya que la mejoría e innovación de tipos de robots como son el BallBot, en el que se basa el TFM, pueden mejorar partes de la industria, por ejemplo, ayudando a transportar cargas en zonas en las que haya personas.

Por otro lado el simple hecho de trabajar en una tecnología que no ha sido aún muy explotada puede abrir nuevas líneas de investigación para seguir innovando en el campo de la robótica o cualquier otro, aunque a priori puedan no guardar una relación aparente.