



**UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO**

FACULTAD DE INGENIERÍA

**ESPACIOS INTELIGENTES CON
MANIPULADORES MÓVILES DOTADOS DE
INTUICIÓN ARTIFICIAL**

TESIS

**PARA OBTENER EL TÍTULO DE
INGENIERO MECATRÓNICO**

PRESENTA:

PABLO CARLOS ÁVILA



DIRECTOR DE TESIS:

DR. VÍCTOR JAVIER GONZÁLEZ VILLELA

México, Ciudad Universitaria, Mayo 2015



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Jurado Asignado.

Presidente: Dr. Marcelo López

Primer Vocal: Dr. Víctor Javier González Villela.*

Secretaria: Dra. María Del Pilar Corona Lira

Primer Suplente: Dr. Octavio Díaz Hernández

Segundo Suplente: M.I. Peña Medina Erik

***Tutor de Tesis.**

Lugar: Facultad de Ingeniería, Ciudad Universitaria, México, Mayo 2015.

Dedicatorias

A mi padre, por ser el mentor a lo largo de mi vida, y un pilar inquebrantable que siempre me guía. Te extraño.

A mi mamá, por su amor, comprensión y apoyo incondicional.

A mi tía Roció, por su ser la columna vertebral, desde mi niñez.

A mis tíos Ale y Vero, por su ser la inspiración ingenieril.

A mi familia, por ser el cobijo de mi felicidad.

A mis amigos, por sus múltiples sonrisas y aventuras a lo largo de cada batalla semestral, con quienes sufrí apasionadamente cada día a lo largo de esta grandiosa odisea en la Facultad.

Agradecimientos

Especial agradecimiento a mi tutor de tesis, maestro y amigo. Que con gusto me guió a través de este proceso de tesis, y ha sabido crear de mí un ser humano en busca inalcanzable de nuevo conocimiento de valor. Gracias Dr. Víctor J.G.V. por su dedicación, paciencia, amabilidad y por las oportunidades que me brindó.

“no sigas a la moda... ¡deja que la moda te alcance!”

A la Universidad Nacional Autónoma de México, por la calidad de su formación profesional. “POR MI RAZA HABLARÁ EL ESPÍRITU”

Agradezco en lo que corresponde a DGAPA, por el apoyo brindado para la realización de este trabajo, a través del proyecto PAPIIT IN117614: “Robótica intuitiva, adaptable, reactiva, híbrida y móvil aplicada al servicio, el rescate y la medicina”

Objetivos de la tesis

Dentro del trabajo de tesis se busca diseñar un espacio inteligente donde el manipulador móvil y el espacio sean conscientes de las actividades que se desarrollan dentro de sí. En la fase experimental la actividad del manipulador móvil estará enfocada al transporte de materiales en coexistencia con el ser humano.

De esta forma se proporcionará una posible alternativa de convivencia entre los robots y los seres humanos.

Resumen

En este trabajo de tesis se presenta un prototipo funcional de espacio inteligente, con el propósito de ser utilizado para que un ente robótico coexista con un ser humano ya sea o no en cooperación.

El espacio inteligente es capaz de saber que hay dentro de sí mismo por medio de un sistema de visión. También es capaz de controlar al ente robótico en tiempo real, por medio de una red de comunicación inalámbrica. Es necesario señalar que el ente robótico sólo es un actuador del espacio inteligente.

Un manipulador móvil es utilizado como ente robótico y tiene programado un algoritmo de intuición artificial, que permite al manipulador móvil desarrollar la tarea de transporte de materiales. El algoritmo de intuición artificial para la coordinación de movimientos está basado en un perfil de trayectoria cúbico.

Palabras claves: Intuición artificial, Espacio Inteligente, Manipulador Móvil, Coexistencia, fiduciales [ver subcapítulo 5.3.5. Sistema de visión].

Contenido

Jurado Asignado.	1
Dedicatorias	2
Agradecimientos	2
Objetivos de la tesis	3
Resumen	3
Contenido	4
Lista de ilustraciones	8
Lista de Tablas	10
Lista de Ecuaciones	10
Introducción a espacios inteligentes	12
1. Espacios Inteligentes	13
1.1. Introducción a los espacios inteligentes	13
1.2. Historia de los espacios Inteligentes.	13
1.2.1 Proyecto Xerox PARC	13
1.2.2 Proyecto Smart Room	14
1.2.3 Proyecto Intelligent Room	14
1.2.4 Proyecto Easy Living	14
1.2.5 Proyecto “The KidsRoom”	15
1.2.6 Proyecto ODISEA	15
1.3. Enfoque comercial de un espacio inteligente en la actualidad	15
1.4. Definición de espacio inteligente	15
1.5. Los espacios inteligentes y la Domótica.	16
1.6. Calidad de un espacio inteligente	16
1.7. Configuración habitual de un espacio inteligente	16
1.8. Ambientes inteligentes	17
1.9. Adquisición de datos en los espacios inteligentes	17
1.9.1 Sistemas de visión comúnmente usados.	17

2. Manipuladores Móviles	19
2.1. Definición de manipulador	19
2.2. Espacio de trabajo de un robot	20
2.3. Estructura mecánica de un robot serial	20
2.4. Grados de libertad en un robot	21
2.5. Robot serial	21
2.5.1 Robot Cartesiano	21
2.5.2 Robot Antropomórfico	22
2.5.3 Robot Esférico o Polar	22
2.5.4 Robot Scara	23
2.5.5 Robot Cilíndrico	23
2.6. Robot Paralelo	24
2.7. Robots móviles	24
2.7.1 Tipos de entornos donde opera un Robot móvil	25
2.7.2 Definición de Manipulador móvil	25
2.7.3 Holonomía en un robot móvil	25
2.7.4 Ventajas y desventajas en un Manipulador móvil	26
2.8. Robot Híbrido	26
2.9. Coexistencia y cooperación de robots móviles y seres humanos	27
2.10. Los robots móviles en simbiosis con espacios inteligentes	28
3. Intuición Artificial	30
3.1. Psique	30
3.2. La intuición en los seres humanos, perspectiva del siglo XVII.	30
3.3. La Intuición artificial y Robótica	31
4. Modelado cinemático del manipulador móvil	34
4.1. Cinemática directa del manipulador serial	34
4.2. Algoritmo De Denavit Hartenberg	35
4.3. Cinemática inversa del manipulador serial	37
4.4. Perfil Geométrico	37
4.4.1 Perfil de trayectoria	39
4.5. Plataforma del robot móvil	46

Diseño del espacio inteligente	51
5.1. Enfoque conceptual	51
5.2. Espacio de trabajo disponible	51
5.2.1 Calibración de la cámara	52
5.3. Diseño conceptual del espacio inteligente	52
5.3.1 Materiales	53
5.3.2 Tamaños	53
5.3.3 Arquitectura	53
5.3.4 Sistema de comunicaciones	54
5.3.5 Sistema de visión	54
Planeación de movimientos dentro del espacio inteligente	57
6.1. Método de campos potenciales	57
6.2. Evasión de obstáculos por medio de campos potenciales	59
6.3. Coordinación de movimientos del robot híbrido	60
6.4. Docking y predocking otra alternativa a la coordinación de movimientos.	63
6.4.1 Base de datos	64
6.5. Programación en el microcontrolador Arduino	65
Experimentación	67
7.1. Objetivos experimentales.	67
7.2. Diseño del experimento.	67
7.3. Resultado experimentales	67
7.3.1 Gasto computacional	71
Conclusiones y trabajo a futuro.	74
8.1. Discusión	74
8.2. Conclusiones	74
8.3. Trabajo a futuro	76
Bibliografía	77
Anexos	79
9.1. Tarjeta MD25 y motores EMG30	79
9.2. Etapa de potencia y alimentación	81

Programas	82
9.3. Programa, cinemática directa del brazo serial en configuración antropomórfica.	82
9.4. Programa, cinemática inversa del brazo serial en configuración antropomorfica.	85
9.5. Diseño de la plataforma para fiducial	97
9.6. Código de programación del microcontrolador Arduino R3	97
9.7. Programación en Matlab	104

Lista de ilustraciones

Ilustración 1. Robot serial de seis grados de libertad KUKA KR-6	19
Ilustración 2. Articulaciones básicas. La letra G se refiere a los grados de libertad que permite cada articulación.	20
Ilustración 3. Anatomía de un robot cartesiano.	21
Ilustración 4. Diagrama de un robot antropomórfico	22
Ilustración 5. Diagrama de un robot esférico	22
Ilustración 6. Robot en configuración Scara	23
Ilustración 7. Robot serial en configuración cilíndrica	23
Ilustración 8. Robot paralelo y su espacio de trabajo.	24
Ilustración 9. Diseño del Manipulador Móvil.	27
Ilustración 10. Manipulador móvil tipo humanoide PR2.	28
Ilustración 11. Intuición Artificial	33
Ilustración 12. Aplicación del método en el brazo antropomórfico, en el programa Mathematica 9. Ver anexo de programas.	36
Ilustración 13. trayectoria de ida que seguirá el manipulador. *Articulación ϑ_2 (unión de eslabones rojo y verde) *Articulación ϑ_3 (unión de eslabones azul y verde) *Articulación ϑ_1 (rotación de la barra roja sobre su eje en Z)	38
Ilustración 14. Primer perfil geométrico de regreso.	39
Ilustración 15. Segundo perfil geométrico de regreso.	39
Ilustración 16. Grafica de las aceleraciones de cada uno de los distintos perfiles de velocidad. Se destaca la menor amplitud del perfil cúbico y una mayor suavidad en el trazo de la curva. Las unidades de la gráfica son $[\text{rad}/\text{s}^2]$ vs $[\text{s}]$.	41
Ilustración 17. Variación de la posición angular de manera paramétrica en la articulación ϑ_1 . A lo largo del perfil geométrico de ida. La exageración de los decimales calculados fue necesaria.	42
Ilustración 18. Variación de la posición angular de manera paramétrica en la articulación ϑ_1 .	43
Ilustración 19. Variación de la posición angular de manera paramétrica en la articulación ϑ_2 .	43
Ilustración 20. Variación de la posición angular de manera paramétrica en la articulación ϑ_3 .	44
Ilustración 21. Sistemas de referencia para el modelado de la plataforma móvil.	47
Ilustración 22. Vectores de velocidad de un par de eslabones que se encuentran unidos.	48
Ilustración 23. Pirámide cuadrangular que ejemplifica el campo de visión de la cámara.	51
Ilustración 24. En la imagen se muestra un área de visión errónea (en color rojo). Se debe garantizar un área rectangular de visión rectangular paralela al piso para evitar deformaciones indeseadas.	52
Ilustración 25. Imágenes de distintos fiduciales	55
Ilustración 26. Esquema representativo del funcionamiento del espacio inteligente.	56
Ilustración 27. Definición del problema de llegar a la meta.	58
Ilustración 28. Método de campos potenciales con obstáculo	59
Ilustración 29. Diagrama esquemático donde se muestra la ubicación de los puntos y distancias	62
Ilustración 30. Docking y predocking	64
Ilustración 31. Secuencia fotográfica del experimento, el tiempo de ejecución solo es ejemplificativo.	69
Ilustración 32. Ilustración 30. Secuencia fotográfica del experimento, el tiempo de ejecución solo es ejemplificativo. solo son perspectivas diferentes	70

<i>Ilustración 33. Características del equipo de cómputo donde se realizó la prueba, la frecuencia de trabajo tiene un control dinámico por lo que puede cambiar.</i>	71
<i>Ilustración 34. Aplicaciones que estaban consumiendo recursos computacionales al hacer las pruebas.</i>	71
<i>Ilustración 35. Gasto computacional cinemática inversa a lo largo del perfil de trayectoria.</i>	72
<i>Ilustración 36. Gasto computacional del movimiento del brazo con las ecuaciones de movimiento obtenidas.</i>	72
<i>Ilustración 37. Comparación del gasto computacional, los datos mostrados corresponden al momento de mayor uso del CPU, que fue donde se realizó la simulación.</i>	73
<i>Ilustración 38. Vista superior de la tarjeta Md25</i>	80
<i>Ilustración 39. Motor Emg30 con caja de engranes acoplado</i>	80
<i>Ilustración 40. Diseño de la etapa de regulación de doce a 5 volts. Fue necesario la puesta en paralelo de los reguladores debido a la corriente necesitada.</i>	81
<i>Ilustración 41. Cinemática Directa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	82
<i>Ilustración 42. Cinemática Directa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	83
<i>Ilustración 43. Cinemática Directa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	84
<i>Ilustración 44. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	85
<i>Ilustración 45. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	86
<i>Ilustración 46. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	87
<i>Ilustración 47. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	88
<i>Ilustración 48. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	89
<i>Ilustración 49. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	90
<i>Ilustración 50. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	91
<i>Ilustración 51. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	92
<i>Ilustración 52. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	93
<i>Ilustración 53. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	94
<i>Ilustración 54. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	95
<i>Ilustración 55. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.</i>	96
<i>Ilustración 56. Diseño de la plataforma para el fiducial</i>	97
<i>Ilustración 57. Programa en Matlab, visión general</i>	104
<i>Ilustración 58. Procesamiento de los datos de visión.</i>	105
<i>Ilustración 59. Controlador y Cinemática de la plataforma móvil</i>	105
<i>Ilustración 60. Bloque de coordinación de movimientos</i>	106
<i>Ilustración 61. Máquina de estados</i>	107
<i>Ilustración 62. Envío de datos generados en Matlab al microcontrolador y cálculo de los ángulos para el manipulador</i>	108
<i>Ilustración 63. Generación de variables internas del programa</i>	109
<i>Ilustración 64. Selección de las distintas trayectorias a realizar por el manipulador.</i>	109

Lista de Tablas

Tabla 1. Parámetros correspondientes al manipulador serial tipo antropomórfico. Ver anexo de programas.	35
Tabla 2. Muestra las magnitudes de aceleración de cada perfil de trayectoria, se destaca la menor amplitud del perfil cúbico. Las unidades de aceleración serán vigentes si el manipulador recorre la trayectoria en 128 segundos	41
Tabla 3. Registros internos de la tarjeta MD25	79

Lista de Ecuaciones

Ecuación 1. La intuición artificial en Robótica. Ver cita 18.	32
Ecuación 2. Cinemática del efector final sin el último grado de libertad.	36
Ecuación 3. Cinemática del efector final, considerando el último grado de libertad y una desviación de 3 cm.	36
Ecuación 4. Perfil de velocidad, armonio simple, ver referencia 19.	40
Ecuación 5. Perfil de velocidad, Cúbico, ver referencia 19.	40
Ecuación 6. Perfil de velocidad, Parabólico (mitad 1), ver referencia 19.	40
Ecuación 7. Perfil de velocidad, Parabólico (mitad 2), ver referencia a 19.	40
Ecuación 8. Perfil de velocidad, Cíclico, ver referencia 19.	40
Ecuación 9. Perfil de velocidad, Polinomio de quinto grado, ver referencia 19.	40
Ecuación 10. Ecuación de movimiento de la junta 1, perfil de ida.	44
Ecuación 11. Ecuación de movimiento junta 1, perfil de regreso 1. Incluye cambio de variable	44
Ecuación 12. Ecuación de movimiento junta 1, perfil de regreso 2. Incluye cambio de variable	44
Ecuación 13. Ecuación de movimiento junta 2, perfil ida	45
Ecuación 14. Ecuación de movimiento junta 2, perfil de regreso 1. Incluye cambio de variable	45
Ecuación 15. Ecuación de movimiento junta 2, perfil de regreso 2. Incluye cambio de variable	45
Ecuación 16. Ecuación de movimiento junta 3, perfil ida	45
Ecuación 17. Ecuación de movimiento junta 3, perfil de regreso 1. Incluye cambio de variable	45
Ecuación 18. Ecuación de movimiento junta 3, perfil de regreso 2. Incluye cambio de variable	45
Ecuación 19. Ecuación de movimiento para la junta 4.	45
Ecuación 20. Ecuación de movimiento para la junta cuatro perfil de regreso	46
Ecuación 21. Plataforma móvil dentro del sistema inercial	46
Ecuación 22. Matriz de rotación	47
Ecuación 23. Velocidad plataforma	47
Ecuación 24. Velocidad del sistema inercial	47
Ecuación 25. Propagación de velocidades, ver bibliografía 20	48
Ecuación 26. Propagación de velocidades. Ver bibliografía 20	48
Ecuación 27. Velocidad angular de la plataforma	48
Ecuación 28. Propagación de velocidades llanta derecha	49
Ecuación 29. Propagación de velocidades llanta izquierda	49
Ecuación 30. Velocidad del sistema inercial	49
Ecuación 31. Propagación de velocidades 3	49

<i>Ecuación 32. Propagación de velocidades en rueda derecha</i>	49
<i>Ecuación 33. Propagación de velocidades en rueda izquierda</i>	50
<i>Ecuación 34. Resumen, velocidad de la rueda derecha</i>	50
<i>Ecuación 35. Resumen, velocidad de la rueda izquierda</i>	50
<i>Ecuación 36. Restricciones cinemáticas de la plataforma móvil</i>	50
<i>Ecuación 37. Velocidad rueda derecha</i>	50
<i>Ecuación 38. Velocidad rueda izquierda</i>	50
<i>Ecuación 39. Distancia de la plataforma a la meta</i>	57
<i>Ecuación 40. Angulo a la meta de la plataforma</i>	57
<i>Ecuación 41. Angulo de error de la plataforma</i>	57
<i>Ecuación 42. Velocidad de la plataforma. Ver cita 20</i>	58
<i>Ecuación 43. Velocidad angular máxima de la plataforma</i>	58
<i>Ecuación 44. Campos potenciales Ver cita 20</i>	59
<i>Ecuación 45. Campos potenciales 2. Ver cita 20</i>	59
<i>Ecuación 46. Distancia total punto de inicio punto 1</i>	60
<i>Ecuación 47. Distancia del punto final 1 al punto 2</i>	60
<i>Ecuación 48. Distancia de la plataforma al punto 1</i>	60
<i>Ecuación 49. Distancia de la plataforma al punto 2</i>	61
<i>Ecuación 50. Parámetro de coordinación 1</i>	61
<i>Ecuación 51. Parámetro de coordinación 2</i>	61
<i>Ecuación 52. Parámetro de coordinación 1 que toma en cuenta el movimiento del brazo robótico. Parámetro α_p, también mencionado como parámetro tomar.</i>	61
<i>Ecuación 53. Parámetro de coordinación 2 que toma en cuenta el movimiento del brazo robótico. Parámetro β_p, también mencionado como parámetro dejar.</i>	61
<i>Ecuación 54. Compensación del error debido a la no homonimia de la plataforma</i>	63

Introducción a espacios inteligentes

Un sueño para la humanidad ha sido la convivencia colaborativa de los robots en la vida cotidiana de cada ser humano, la pantalla grande se vuelve una herramienta que nos ayuda a imaginar cómo podría ser este tipo de convivencia, omitiendo por supuesto, los finales catastróficos. Pero, ¿es posible coexistir con un robot dentro de un ambiente inteligente?

Los seres humanos coexisten con distintas especies de animales, por ejemplo en los hogares la gente suele tener a un perro como mascota. Ambos seres vivos *comparten el mismo espacio haciendo cosas diferentes al mismo tiempo* (esto es, coexistencia), por ejemplo el perro puede estar vigilando al tiempo que tú lees un libro en el jardín.

Actualmente hay robots comercialmente hablando que ya pueden coexistir con los seres humanos en una casa, un ejemplo de este tipo de robots viene de la compañía LG “HOM-BOT SQUARE”, es un robot capaz de barrer y aspirar los pisos de un hogar. El robot limpia de manera automática pero no es capaz de saber que hay alrededor, únicamente evade obstáculos de todo tipo para poder realizar su tarea.

En el caso del ambiente se plantea cambiar de paradigma y diseñar un espacio inteligente para que el robot se sienta como en casa, pero sin que el espacio deje de ser familiar para nosotros. Se vuelve importante mencionar que la convivencia y la coexistencia se planea sean realizadas en un ambiente donde el manipulador móvil pueda saber que hay alrededor y por otra parte el propio ambiente sea capaz de percibir lo que ocurre dentro de sí y sea consciente de las actividades que se realizan dentro de sí mismo. La tarea que será desarrollada dentro del espacio será el transporte de materiales.

1. Espacios Inteligentes

1.1. Introducción a los espacios inteligentes

La inteligencia de manera general es definida como la capacidad de entender o *comprender*, de acuerdo con la teoría vigente postulada en 1983 por Howard Gardner (Psicólogo estadounidense, galardonado con el Premio Príncipe de Asturias de Ciencias Sociales en 2011)” [1].

De manera general para poder hablar de un espacio inteligente estrictamente tendría que haber una capacidad del propio espacio para comprender lo que ocurre dentro de sí mismo.

1.2. Historia de los espacios Inteligentes.

La idea de un espacio inteligente surge en los años setenta donde se buscaba un mayor nivel de confort en los hogares y edificios, que permitieran al usuario realizar menos tareas domésticas. Es difícil hablar de espacios inteligentes ya que solo se tenía maquinaria accionada por botones para tareas muy específicas por ejemplo abrir o cerrar puertas, de esta forma solo podemos decir que se automatizaron ciertas tareas en el hogar.

1.2.1 Proyecto Xerox PARC

Un pionero en este ámbito fue la compañía Xerox, quien en el año de 1970 funda una división de investigación y desarrollo en Palo Alto (California, Estados Unidos), del proyecto *Xerox PARC (Palo Alto Research Center)* destaca el científico Mark Weiser, quien a finales de los años ochenta propone la idea de “Computación Ubicua (ubicop)”, de esta idea se deriva el concepto de espacio inteligente. Weiser destaca ya que intenta integrar la informática con los seres humanos. Y desarrolla una primera red de sensores intercomunicados, donde buscaba cuatro objetivos específicos:

- *Ubicuidad*. Múltiples sistemas con comunicación mutua.
- *Conocimiento*. Capacidad del sistema para percibir lo que ocurre en el entorno.
- *Inteligencia*. Capacidad de adaptarse al mundo.
- *Interacción Natural*. Capacidad de comunicación entre el entorno y los usuarios.

En el artículo “The Computer for the 21st Century”, Weiser predice que los ordenadores se volverán “invisibles” y estarán en nuestro alrededor integrados en todos los objetos sin que nos demos cuenta. Weiser es considerado un gran visionario que predijo el uso de dispositivos como tablets. Si bien algunas de sus predicciones se han cumplido, hay otras que aun suenan descabelladas.

1.2.2 Proyecto Smart Room

A finales de los noventa destacan dos proyectos que nacen en el MIT (Massachusetts Institute of Technology), el primer proyecto *Smart Room (1996)* ^[10]; tiene diversos tipos de sensores entre los que sobresalen cámaras y micrófonos, con la finalidad de detectar a los usuarios por medio de su tono de voz y apariencia. Este proyecto tiene una línea de aprendizaje a través del usuario, es decir que el usuario puede literalmente enseñarle nuevas geometrías y sus nombres.

1.2.3 Proyecto Intelligent Room

El segundo proyecto del MIT, *Intelligent Room (1998)* ^[11]; En la actualidad aun es considerado un proyecto muy evolucionado. Y destaca porque es capaz de interpretar objetivos mediante cámaras. El propósito del sistema es buscar una interacción del usuario con el entorno.

A pesar de que son dos proyectos que comparten similitudes fueron desarrollados de manera independiente por dos divisiones distintas: “Media Lab” y “Artificial Intelligent Laboratory” ambos en el MIT.

1.2.4 Proyecto Easy Living

En el caso de Microsoft tiene un proyecto llamado *Easy Living* ^[12]. Que busca desarrollar nuevas tecnologías para la creación de espacios inteligentes que faciliten la interacción de personas con dispositivos tecnológicos, así los dispositivos tecnológicos podrán ayudar a los usuarios en sus tareas cotidianas.

De manera similar a los proyectos anteriores se busca establecer la comunicación del entorno con el usuario de la manera más natural para nosotros como humanos.

1.2.5 Proyecto “The KidsRoom”

The KidsRoom es un ambiente creado para la interacción enfocada a niños. Tiene como objetivo principal divertir a los pequeños por medio de sistemas de visión computacionales, que interpretan las acciones de los infantes y en base a ello un sistema de narración reacciona. Fue creado en el MIT con fines didácticos.

1.2.6 Proyecto ODISEA

Proyecto ODISEA, desarrollado en la Universidad Autónoma de Madrid, es un entorno inteligente que se conforma de dos capas: la capa de interacción con el entorno físico que incluye el conjunto de dispositivos que componen la infraestructura del ente inteligente, y la capa de contexto que es la que ofrece información sobre lo que pasa dentro del espacio inteligente, buscando la interpretación de los deseos del usuario.

1.3. Enfoque comercial de un espacio inteligente en la actualidad

Por otro lado la empresa *Electric Objects* tiene un lema muy enigmático: “Internet está lleno de cosas bellas, y estamos construyendo un dispositivo que está a la altura de ello. Vamos a poner Internet en tu pared”. El enfoque que actualmente tienen es buscar pantallas de televisión tecnológicamente cada vez más avanzadas. Lo anterior como una primera iteración de dispositivos inteligentes de los cuales ya podemos gozar en nuestros hogares. Y que sin duda seguirán inundando nuestros espacios. En el caso de los nuevos televisores son conscientes de lo que pasa enfrente de ellos y son capaces de interpretar lo que el usuario desea sin necesidad de un control remoto habitual, únicamente por medio de sensores de voz y movimiento.

1.4. Definición de espacio inteligente

Los Espacios Inteligentes son habitaciones o áreas capaces de percibir y entender lo que ocurre en ellas. Están equipados con diferentes tipos de sensores y dispositivos de comunicación. A través de los sensores, los “Espacios Inteligentes” captan lo que ocurre en ellos, analizan las diferentes situaciones, reaccionan ante ellas y se comunican con los usuarios a través de los actuadores. ^[9] En estos espacios los seres humanos y máquinas colaboran entre sí. Deben extraer y mantener una conciencia de los eventos y actividades humanas que se producen dentro del entorno. ^[6]

1.5. Los espacios inteligentes y la Domótica.

Es necesario aclarar que en Domótica también se habla de espacios inteligentes. Y en este caso se dota de dispositivos automáticos a una habitación en busca de un mayor confort, y de sustentabilidad. Esto puede llegar a generar confusiones y se aclara que en este trabajo se buscara la convivencia de un robot móvil con el entorno y no de dispositivos automáticos que reaccionen aisladamente.

La idea es fomentar la convivencia de un robot con el entorno, donde el robot sea capaz de percibir lo que pasa a su alrededor y así en un futuro los robots puedan proporcionar un servicio a los humanos o ayudarles a realizar una tarea. Lo anterior dentro de un espacio inteligente.

1.6. Calidad de un espacio inteligente

Lo que hace que un espacio sea denominado inteligente es cuando toma decisiones de manera autónoma, en el caso del presente trabajo se habla de un espacio inteligente porque será capaz de “ser consciente” de las cosas que ocurren dentro del espacio de trabajo, y en base a ello tomar decisiones dependiendo de la tarea a desarrollar, como primera iteración transporte de material.

1.7. Configuración habitual de un espacio inteligente

Un espacio inteligente debe ser capaz de “estar consciente” de lo que pasa dentro de él, y así poder actuar de acuerdo a lo deseado dentro del espacio, se habla de cuatro capas que conforman al espacio.

Capa Pasiva de Percepción: Está compuesta de sensores que captan la información, pueden ser de distintos tipos de acuerdo a lo que se requiere percibir. Los sensores pueden estar fijos o estar dentro de la capa de interacción con los usuarios.

Capa de Interacción: La componen los elementos que pueden controlarse e interactúan de manera física con el medio, recordando al párrafo anterior pueden ayudar a medir alguna variable deseada (como capa activa de percepción), y también ayudan a realizar la función del espacio inteligente.

Capa de Comunicaciones: Esta capa se refiere a la red de interconexiones que hay entre los diversos elementos para comunicarse.

Capa de Inteligencia: Es la parte encargada de procesar la información generada en la capa de percepción, y quien controla a la capa de interacción. También es la responsable del intercambio de información mediante la capa de comunicaciones.

Es necesario mencionar dentro del espacio inteligente puede haber objetos estáticos, objetos dinámicos y entes autónomos. Los objetos estáticos son simplemente objetos en reposo, los objetos dinámicos son objetos en movimiento, y un ente autónomo es quien interactúa con el espacio inteligente pero no es controlado por la capa de inteligencia, por ejemplo un ser humano.

1.8. Ambientes inteligentes

La idea de un espacio inteligente ha ido en auge en los últimos años debido en gran medida a un decremento en el coste de los sensores y equipos de cómputo, llevado de la mano de una mayor capacidad de procesamiento. El desarrollo de espacios inteligentes en todo tipo de áreas permitirá *ambientes inteligentes*, es decir un conjunto de espacios inteligentes.

1.9. Adquisición de datos en los espacios inteligentes

Uno de los problemas en los espacios inteligentes es la adquisición de datos para realizar la tarea correctamente, por ejemplo si tú le pides a un compañero de salón que te alcance el lápiz que se te acaba de caer cerca de él, le será una tarea fácil. Pero en robótica se vuelve un reto debido a que no es tan trivial para un robot percibir el ambiente.

Una técnica habitualmente usada para la adquisición de datos son los sistemas de visión artificial como cámaras, este tipo de sistemas suelen inspirarse en la forma en que naturalmente nosotros percibimos al entorno (por medio del sentido de la vista).

1.9.1 Sistemas de visión comúnmente usados.

Actualmente hay una tendencia muy fuerte por imitar la forma como vemos el entorno, lo anterior ha llevado múltiples avances en los sistemas de visión que se esfuerzan por obtener imágenes del mundo exterior y procesarlas de manera adecuada para que el robot pueda percibir el entorno.

Una tendencia muy fuerte es el reconocimiento de objetos tridimensionales, un reto formidable que ha tenido múltiples avances y ha sido posible gracias a equipos de cómputo cada vez más robustos. Ya que aparte de detectar objetos también deben de ser capaces de ubicarlos espacialmente.

Otra metodología para la detección de objetos en los sistemas de visión es el modelo de características o método SIFT (Scale Invariant Feature Transform), basado en la detección de puntos de interés en las imágenes, fue presentado por el David Lowe en 2004^[8]. Esta alternativa realiza *matching* con imágenes patrón de objetos que se desea detectar y una imagen que es tomada por una cámara que contenga al objeto. Este modelo relativamente reciente se ayuda de técnicas de estimación para evitar un falso *matching*. Esta forma de percibir al medio suele ayudarse de varias cámaras para tener una mejor efectividad, sin embargo la deformación proyectiva de las cámaras suele ser un problema, porque la imagen del objeto se deforma y no se detectan con éxito los objetos. Este sistema suele ayudarse de marcas artificiales del objeto o marcas naturales del objeto en sí. La tendencia anterior suele tener múltiples problemas debido a que todas las imágenes de los objetos tridimensionales deben ser caracterizadas con anterioridad, esto es para garantizar una mayor efectividad.

Existe también la tendencia de obtener los modelos geométricos más simples que conforman a un objeto. Lo anterior es logrado extrayendo de la imagen puntos característicos propios de cada geometría, y al comparar en con imágenes guardadas previamente puede determinar que objeto es. Esta tendencia suele ser más robusta pero la efectividad depende de que tan caracterizado este el objeto.

2. Manipuladores Móviles

2.1. Definición de manipulador

Los manipuladores robóticos son dispositivos que pueden llevar a cabo un número variado de tareas con rapidez y precisión, además de que se pueden adaptar a diversas posiciones con solo modificar el programa que comanda los movimientos de la máquina. En todo manipulador se pueden distinguir cuatro componentes: Mecánico, Hardware, control y software. Es la suma de ellos lo que hace que los manipuladores sean versátiles, inteligentes, robustos y de excelente repetitividad. [21]

Los robots manipuladores suelen trabajar en medios protegidos y conocidos, mediante movimientos planeados con anterioridad.

Ejemplos de tareas a nivel industrial que realizan: manipulación de materiales, aplicación de pegamentos, procesos de soldadura, corte, pintura, ensamble, fundición, inspección, control de calidad, pulido de partes, alimentación de piezas a una máquina, empaque de productos entre otras. [21]

De acuerdo a su arquitectura los robots pueden ser clasificados en seriales, paralelos o híbridos, la anterior es solo una de las múltiples clasificaciones que pueden tener.

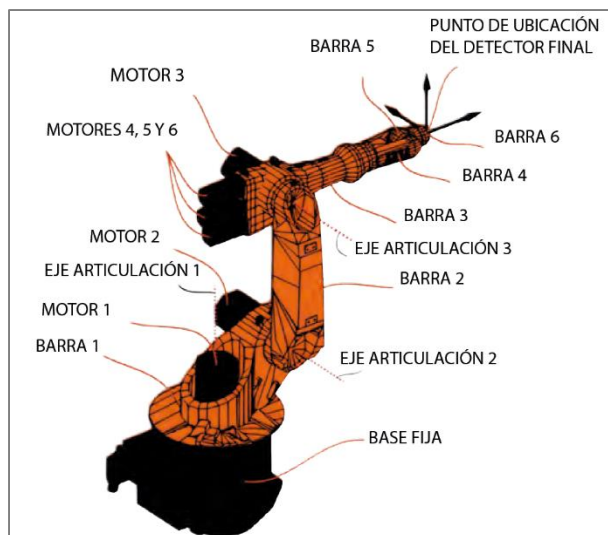


Ilustración 1. Robot serial de seis grados de libertad KUKA KR-6

2.2. Espacio de trabajo de un robot

El espacio de trabajo de un robot también es denominado Espacio de trabajo Efectivo, y “*está definido por el grupo de puntos que pueden ser alcanzados por su efector final*” (Cao 2011) ^[16].

2.3. Estructura mecánica de un robot serial

Mecánicamente un robot se forma por los eslabones y las articulaciones. Las articulaciones son los elementos mecánicos que permiten el movimiento relativo entre sus eslabones.

Las articulaciones delimitan en gran medida el espacio de trabajo del robot. Las articulaciones más comunes en los robots son las articulaciones rotacionales, y las articulaciones prismáticas. En la figura tres encontramos las articulaciones los diagramas de distintos tipos de articulaciones y los grados de libertad que permiten cada una de estas.

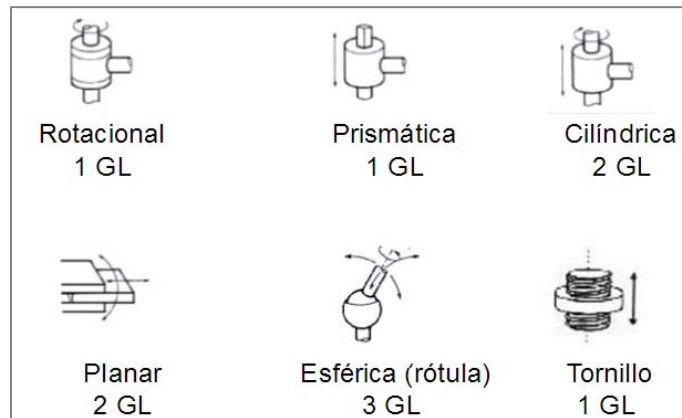


Ilustración 2. Articulaciones básicas. La letra G se refiere a los grados de libertad que permite cada articulación.

2.4. Grados de libertad en un robot

Cada uno de los movimientos independientes que puede realizar cada articulación con respecto al anterior se denomina grado de libertad. Los grados de libertad de un robot son la suma de los grados de libertad de las articulaciones que forman parte de su arquitectura

2.5. Robot serial

Un Robot de tipo serie tiene una cadena cinemática abierta, dependiendo del tipo de las primeras tres juntas que posean hay una segunda clasificación según J.M. Seling.

2.5.1 Robot Cartesiano

Robot Cartesiano (PPP), sus tres primeras juntas son prismáticas y son ortogonales entre sí, por lo que cada movimiento en una junta prismática suele ser asociado con cada uno de los ejes coordenados (X, Y, Z).

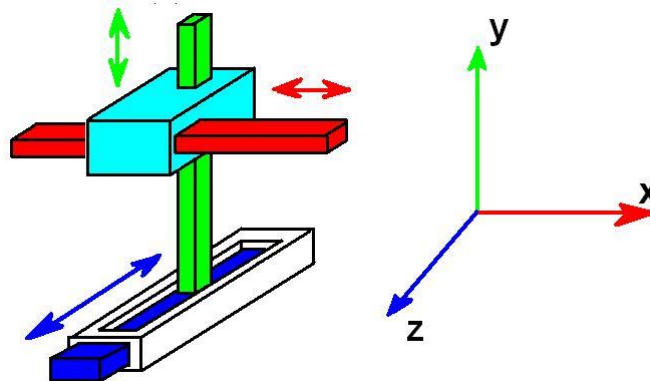


Ilustración 3. Anatomía de un robot cartesiano.

2.5.2 Robot Antropomórfico

Robot Antropomórfico (RRR), sus tres principales juntas son rotacionales, simulan a groso modo a un brazo humano.

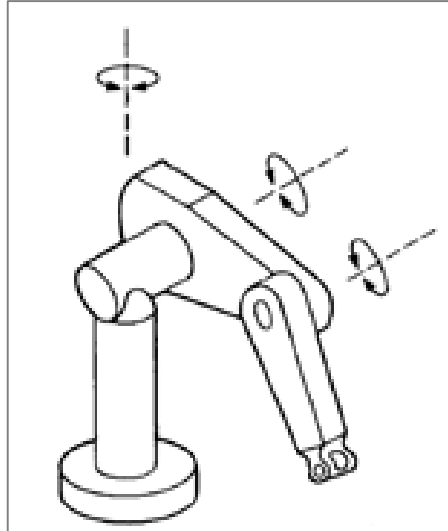


Ilustración 4. Diagrama de un robot antropomórfico

2.5.3 Robot Esférico o Polar

Un robot esférico (RRP), es un robot cuyos ejes conforman un sistema de coordenadas polares, la primera y la segunda junta son rotacionales y son perpendiculares entre sí; su última junta es prismática.

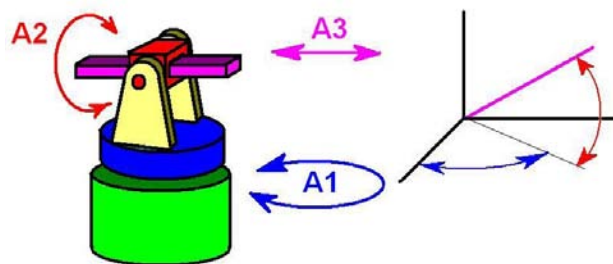


Ilustración 5. Diagrama de un robot esférico

2.5.4 Robot Scara

El Robot Scara (RRP) tiene una estructura conformada por dos articulaciones rotacionales y una prismática. Esta configuración es de las más comunes comercialmente ya que es ideal para trabajar sobre el plano de trabajo (X-Y).

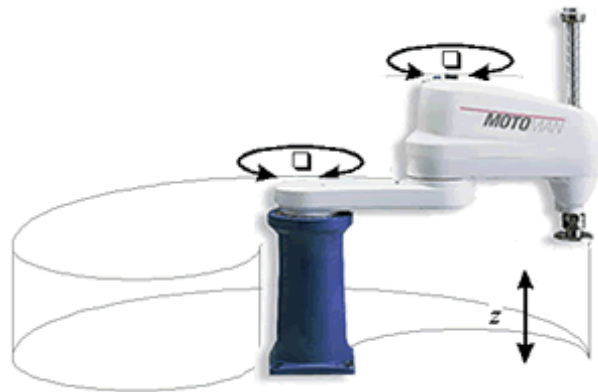


Ilustración 6. Robot en configuración Scara

2.5.5 Robot Cilíndrico

Los robot en configuración cilíndrica (PRP), tienen un movimiento rotacional en la base y dos juntas prismáticas en forma ortogonal, los que simula un sistema de coordenadas cilíndricas.

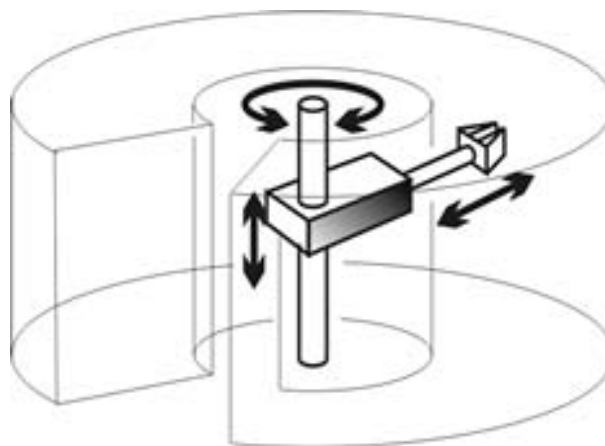


Ilustración 7. Robot serial en configuración cilíndrica

2.6. Robot Paralelo

Existe por otro lado un segundo tipo de robots manipuladores. Se caracterizan por tener una cadena cinemática cerrada, es decir tienen una base fija y una plataforma móvil unida por varias extremidades, generando así cadenas cinemáticas en paralelo. Este tipo de configuración permite soportar grandes cargas, y altas velocidades de operación así como mayor repetitividad pero tienen un espacio de trabajo más reducido.

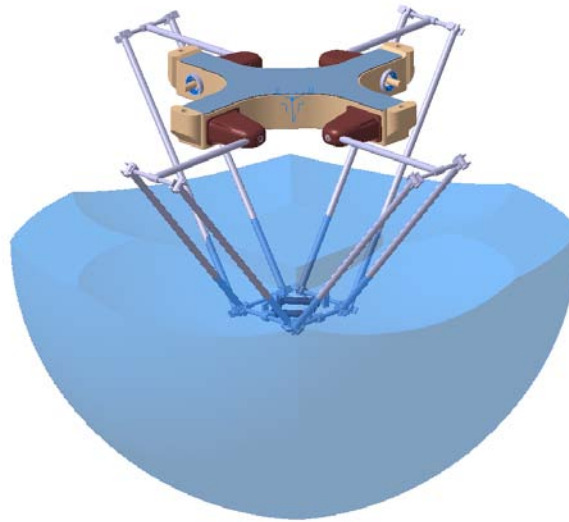


Ilustración 8. Robot paralelo y su espacio de trabajo.

2.7. Robots móviles

Los robots móviles son robots con sistemas de locomoción, por lo general ruedas, orugas o patas que permiten el desplazamiento de un lugar a otro. Este tipo de robots cuentan con una gran capacidad de desplazamiento de esta forma su espacio de trabajo aumenta considerablemente. Existen múltiples tipos de robots móviles, en el presente trabajo solo nos enfocaremos en robots móviles terrestres.

2.7.1 Tipos de entornos donde opera un Robot móvil

El tipo de entorno donde operará el robot móvil define en gran medida la estructura del mismo. En este ámbito tenemos una clasificación muy marcada. Si el robot móvil operara en un ambiente externo o interno.

El ambiente de trabajo es considerado interno si se encuentra rodeado por un medio artificial, que impide esté en contacto con el medio ambiente. Los medios artificiales son paredes, techo, suelo e inmobiliario en general. Y la iluminación puede venir de un medio artificial como lámparas.

El ambiente de trabajo se considera externo si el área de trabajo no está claramente delimitada y el tipo de iluminación es principalmente natural.

2.7.2 Definición de Manipulador móvil

Existen diversos tipos de manipuladores móviles dependiendo del tipo de tarea que se busque que desarrollen. En el presente trabajo únicamente nos enfocaremos en manipuladores móviles que estén enfocados a tareas dentro de un espacio cerrado, buscando una posible convivencia a futuro con humanos.

Un Manipulador móvil consiste en un robot manipulador que es montado sobre un robot móvil. Su propósito consiste en sujetar y manipular objetos de un lugar a otro.

2.7.3 Holonomía en un robot móvil

Una forma de clasificar a un robot es por su movilidad, es decir si es o no, holonómico. Un robot se dice que es holonómico si puede modificar su dirección instantáneamente y sin la necesidad de haber rotado previamente, esta consideración no toma en cuenta a la masa del robot. Es decir que el robot tiene plena movilidad en el plano.

En este caso se usará un robot manipulador móvil que tiene una configuración de ruedas diferencial, es decir que cambia de dirección gracias a una diferencia de velocidades de las dos ruedas que posee. Esta configuración es no holonómica.

2.7.4 Ventajas y desventajas en un Manipulador móvil

Una ventaja notable es que el espacio de trabajo aumenta considerablemente en comparación con el espacio de trabajo de un manipulador únicamente. Pero en términos de control puede aumentar la complejidad del manipulador, sobre todo cuando la plataforma móvil es no holonómica o cuando las llantas presentan deslizamiento como lo es el caso de las orugas.

Su tarea básica consiste en manipular objetos, así como sujetarlos. Suelen ser usados en líneas de producción para el ensamble de componentes. Un problema habitual en un manipulador móvil es el constante cambio de su centro de masa en función del desplazamiento del manipulador móvil, esta variación puede provocar oscilaciones mecánicas en el sistema.

Existe un problema habitual en los robots manipuladores móviles llamado redundancia cinemática, problema que será abordado más adelante.

La planificación de la trayectoria es un problema habitual, por ejemplo en el presente trabajo se utilizará una plataforma no holonómica, que no permite cualquier trayectoria.

2.8. Robot Híbrido

Es la sinergia de al menos un par de estructuras robóticas distintas, que toman ventaja de las propiedades de cada estructura.

En el presente trabajo utilizaremos un robot híbrido. Es decir es la fusión de dos tipos distintos de robots, el sistema está conformado por una plataforma móvil en configuración diferencial, un brazo serial de tipo antropomórfico que cuenta con un gripper como efector final. El manipulador móvil fue diseñado por el alumno Pedro M. Escobedo Castillo miembro del grupo de Mechatronics Research Group.

El robot híbrido fue escogido como una herramienta para que ayude al espacio inteligente a manipular los objetos de un lado a otro. El robot manipulador móvil tiene una arquitectura modular manufacturada en acrílico.

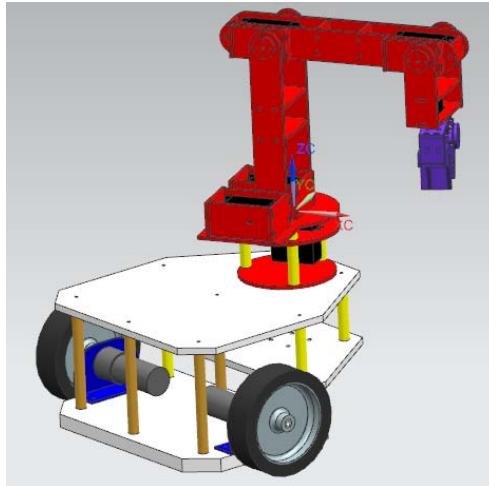


Ilustración 9. Diseño del Manipulador Móvil.

2.9. Coexistencia y cooperación de robots móviles y seres humanos

Definiciones:

- *Coexistencia*: El humano y el robot comparten el mismo espacio de trabajo al mismo tiempo mientras trabajan en tareas diferentes.
- *Cooperación*: El humano y el robot comparten el mismo espacio de trabajo al mismo tiempo mientras trabajan en la misma tarea.

La convivencia robot-humano en una misma tarea es algo prohibitivo en ambientes industriales debido a normas como resultado de accidentes laborales que se han suscitado en el pasado. Actualmente en ambientes industriales no existe una coexistencia entre ambos entes, pero si existen líneas de producción donde el robot trabaja en tareas no recomendables para humanos y la mano de obra humana se desempeña en tareas versátiles no aptas para robots. Por ejemplo una línea de producción de automóviles, los robots y los humanos trabajan para poder hacer literalmente un automóvil. En este caso las tareas son realizadas independientemente pero en una misma línea de producción.

Existen múltiples intentos por establecer una coexistencia ya sea o no en cooperación con los seres humanos. Comercialmente existe un lanzamiento de un robot manipulador móvil de tipo humanoide lanzado al mercado por la compañía Willow Garage. La compañía ha realizado múltiples lanzamientos del robot PR2.

El robot de tipo humanoide se caracteriza por poseer código abierto de programación, así como tener la versatilidad de cooperar y coexistir en tareas asignadas por el usuario. Su primer lanzamiento fue realizado en 1998. La empresa se caracteriza por desarrollar robots

con fines no militares. PR2 tiene movilidad en ambientes humanos y te permite ser más productivo.

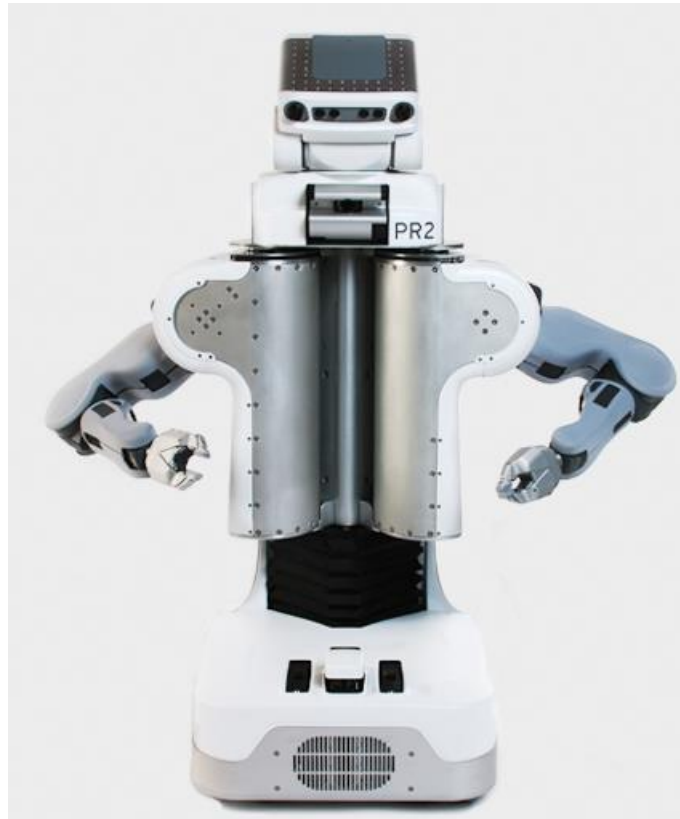


Ilustración 10. Manipulador móvil tipo humanoide PR2.

En el presente trabajo se fomenta una coexistencia entre el manipulador móvil y el ser humano. El riesgo mayor que podría implicar para el ser humano es una colisión entre ambos entes sin que implique riesgos para el ser humano debido a la diferencia de tamaños. Es necesario señalar el espacio de trabajo de ambos entes es el espacio inteligente que evitara en la medida de lo posible colisiones de todo tipo con el manipulador móvil.

2.10. Los robots móviles en simbiosis con espacios inteligentes

Una característica de los espacios inteligentes es su habilidad de percibir lo que pasa dentro de ellos, y es posible gracias a sistemas de visión cada vez más evolucionados.

El desarrollo de los robots móviles ha tenido por décadas la meta de crear entes autónomos con capacidad de operar en ambientes desconocidos. El desarrollo de la robótica móvil dentro de los espacios inteligentes soluciona muchos problemas como la localización del robot, planeación de tareas, evasión de obstáculos entre otros.

El gran problema en robótica móvil se reduce al reconocimiento y localización de los objetos. Ha sido un reto que han intentado solucionar por medio de sistemas de visión sin lograr gran versatilidad. Por ejemplo el problema en la planeación de trayectorias se reduce a la obtención de los mapas que componen al espacio. El primer método usa “mapas estáticos”, que son definidos por el usuario. El segundo método el robot realiza una “exploración autónoma”. Ambos métodos fallan si algo es movido del lugar debido a que primero se realiza la obtención del plano y después se planea la trayectoria del robot móvil, es necesario aclarar que ambos procesos se hacen por separado.

El problema anterior se soluciona si el robot móvil opera dentro de un espacio inteligente, debido a que el robot móvil puede utilizar sensores del espacio inteligente y de esta forma el robot puede obtener información de todo el espacio inteligente y de su ubicación dentro del mismo.

La simbiosis del robot móvil con espacio inteligente, permite que el robot móvil sea quien realice tareas dentro del espacio, de esta forma el robot móvil es un agente dinámico que es controlado por el espacio inteligente. En resumen el robot móvil se convierte en el actuador del espacio inteligente.

3. Intuición Artificial

Desde la antigüedad el hombre ha especulado sobre la naturaleza del comportamiento humano, los egipcios y babilónicos entendieron al comportamiento humano en términos de la actividad de ciertos “espíritus” o “almas” inmateriales. Los griegos le asignan el término de *physis*, del que deriva el vocablo “psicología”; *physis* se liga etimológicamente con aliento (*pneuma*) o viento (*onians*), y se asocian íntimamente con la acción del corazón y pulmones. El término *physis* en la Grecia antigua fue concebido como el principio general de la vida, y que podía o no, ser inmaterial. Según la *teoría de las formas* de Platón (429-325 a.C.), todo conocimiento es remembranza que posee el *psique inmortal*, y se olvidan temporalmente en cada reencarnación.

3.1. Psique

Aristóteles (384-322 a.C.), habla de varios niveles de psique, el nivel más bajo está en la *psique nutritiva*, la esencia de las plantas. La *psique sensible*, esencia de los animales, responsable del placer, dolor, imaginación, memoria, y la locomoción por medio del deseo. La psique racional o mente (*nous*), que es la esencia de los seres humanos, sirve para varias funciones cognitivas, como la abstracción, la toma de decisiones, y la memoria; esta última psique adopta a las dos psiques anteriores. Aristóteles afirmaba que la psique racional es una capacidad única de los humanos, que nos distingue de los animales y plantas. ^[2]

3.2. La intuición en los seres humanos, perspectiva del siglo XVII.

René Descartes en el siglo XVII, en su obra “Tratado Sobre el hombre”, fue el primero en ofrecer una explicación fisiológica a lo que comúnmente llamamos reflejos, los determina como algo automático e involuntario:

Si alguien de manera súbita moviera su mano delante de nuestros ojos como si fuera a pegarnos, entonces aunque supiéramos que es nuestro amigo, que sólo lo hace por diversión y que tendrá cuidado de no lastimarnos, nos resulta difícil no cerrar los ojos. Esto muestra que no es por medio de la meditación de nuestra alma que se cierran, ya que esta acción es contraria a nuestra voluntad, la cual es la única, o al menos la principal actividad del alma. Se cierran más bien porque el mecanismo de nuestro cuerpo está tan ordenado que el movimiento de la mano hacia nuestros ojos produce otro movimiento en nuestro cerebro, el cual dirige a los espíritus animales de nuestros músculos que hacen que nuestros parpados se cierren. ^[4]

Descartes afirmaba que esta forma de actuar mecanicista se extiende a todo comportamiento animal y buena parte de la conducta humana. También afirmaba que el conocimiento sobre la naturaleza podría alcanzarse solo por medio de la *intuición racional*, independiente a la experiencia sensorial. Por tanto argumentaba que algunos principios teóricos generales se conocen *a priori*, es decir independientemente de la experiencia sensorial.

Lo anterior ha sido una prueba de que a lo largo de miles de años el propio ser humano ha intentado descifrar cómo es posible que un pedazo de materia sea capaz de pensar, sentir, razonar, percibir y actuar. Y aun es algo completamente místico que no ha podido ser descifrado. La única teoría alternativa real *es la mística: las mentes funcionan en algún reino místico que está más allá de la ciencia física* ^[5]. Esta última es la teoría más aceptada y por eso es que múltiples avances científicos no han podido recrear la mente humana.

En esta primera parte se destaca la teoría de Descartes, y se hace hincapié a que muchas actividades humanas son hechas involuntariamente sin la necesidad de pensar exhaustivamente, por ejemplo cuando tropezamos con algún obstáculo al caminar, solemos reaccionar rápidamente sin tener conciencia de ello y de esta forma evitamos caernos, lo anterior gracias a eventos similares que se adquieren gracias a experiencias previas. Este tipo de procesos mentales suelen pasar muy a menudo en la vida cotidiana sin que estemos plenamente consciente de ello, por ejemplo al ver un una mesa sabemos que es y para que nos puede servir sin la necesidad de estar pensando ¿qué es ese objeto en medio del comedor? y ¿para qué sirve?, esto es gracias a antiguas experiencias que hemos tenido con el paso de los años.

La intuición humana es parte de nuestras vidas, una parte de esta, ha sido adquirida gracias a experiencias personales que cada individuo ha experimentado, y existe otra parte de la intuición humana que ha sido heredada gracias a la naturaleza propia del ser humano.

3.3. La Intuición artificial y Robótica

La necesidad de dotar a un ente robótico de intuición artificial es buscar que el robot realice una tarea de manera muy rápida y confiable que emule un comportamiento humano.

La intuición artificial es una rama de la inteligencia artificial, que se dedica a emular el reconocimiento automático de patrones de información para generar respuestas rápidas y normalmente certeras o confiables. ^[18]

Características de un proceso intuitivo según Martin Seligman: es rápido, carece del uso de conciencia, está basado en experiencias previas, toma decisiones en múltiples dimensiones, no es sencillo de explicar y casi siempre se logra ejecutar la tarea con éxito.

Y de acuerdo a la teoría vigente publicada por el Dr. Octavio Díaz Hernández miembro del grupo MRG. *La intuición artificial en Robótica, es un subespacio de configuraciones específicas por una tarea realizada con intuición, al que llamamos Espacio Intuitivo de la Tarea, o $A(q)_{\text{intuición}}$.*^[18] Se denomina subespacio debido a que es parte del espacio de trabajo de la tarea, que a su vez pertenece al espacio de trabajo del robot, matemáticamente:

$$A(q)_{\text{intuición}} \subseteq A(q)_{\text{tarea}} \subseteq W \subseteq \text{Mundo}$$

Ecuación 1. La intuición artificial en Robótica. Ver cita 18.

Una característica primordial de los algoritmos intuitivos es que la respuesta generada debe de ser única en un evento o momento particular aunque cada algoritmo intuitivo produzca una respuesta individual dependiendo claro, de las entradas que se le den al *sistema intuitivo*. En el presente trabajo el algoritmo intuitivo no tiene mecanismos de aprendizaje, porque busca una respuesta rápida y sin un exceso de gasto computacional. Pero si es necesario mencionar que parte de la intuición se basa en cosas que ya se conocen con anterioridad, pero existen otro tipo de tareas que son nuevas y a pesar de ello se realizan con intuición.

En el caso del presente capítulo el brazo serial basa sus movimientos en un perfil de trayectoria que ha sido normalizado en base a patrones de visión que se obtienen de la cámara y todo esto permite que el manipulador móvil realice la tarea con éxito. Es necesario hacer hincapié que el lugar geométrico que se obtiene por medio de la intuición artificial que ha sido programada varía de acuerdo a la trayectoria que sigue la plataforma móvil, este tema se tocara más a fondo en el capítulo seis.

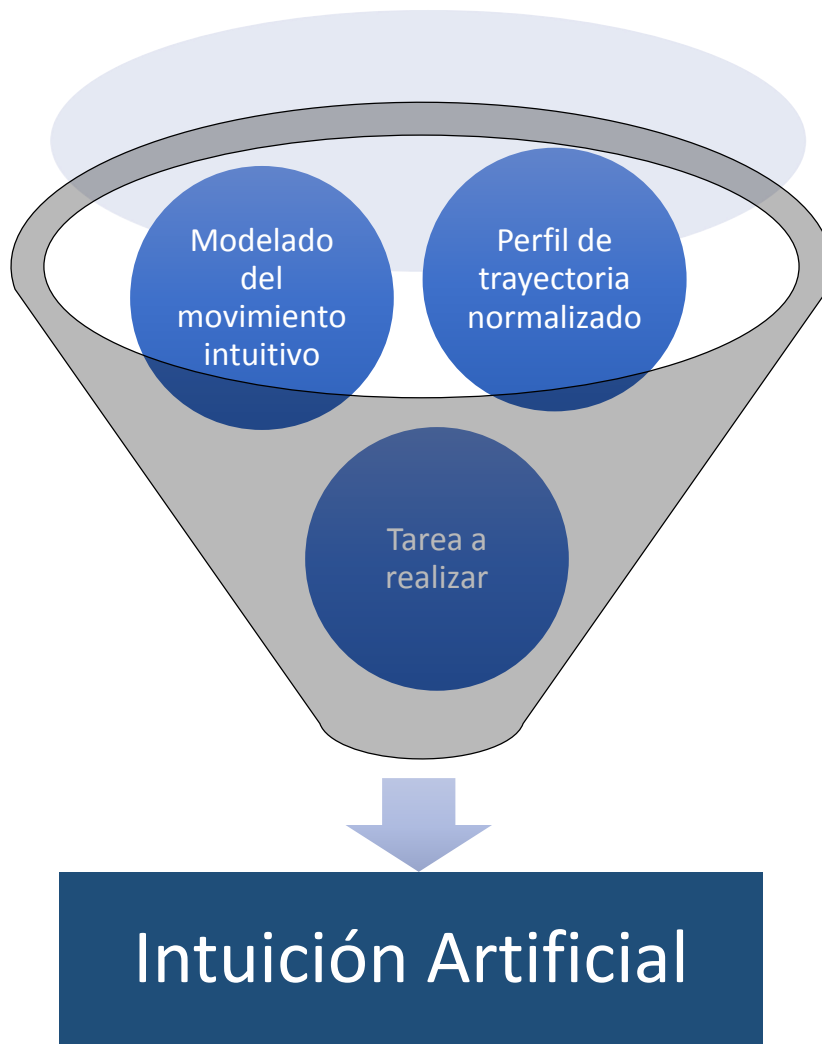


Ilustración 11. Intuición Artificial

4. Modelado cinemático del manipulador móvil

Este capítulo estará enfocado al modelo cinemático del brazo serial para poder moverlo con éxito. Recordando que el Manipulador móvil a utilizar en el presente trabajo consiste en un brazo robótico de tipo antropomórfico que es montado sobre una plataforma móvil en configuración diferencial con el propósito de sujetar y manipular objetos de un lugar a otro.

4.1. Cinemática directa del manipulador serial

Para la robótica es muy importante conocer la ubicación en el espacio tridimensional y la orientación de los objetos que se verán involucrados en la tarea a realizar. Habitualmente se utiliza un sistema de referencia coordenado (X, Y, Z) global y se suele auxiliarse de sistemas de referencia locales.

A manera de definición la cinemática es la ciencia que estudia el movimiento de los cuerpos sin tomar en cuenta las causas de este. Y la cinemática directa es el estudio de la manipulación mecánica, que es el problema geométrico de calcular la posición y orientación del efector final del manipulador en una situación estática.

Un robot serial está formado por eslabones y juntas con un efector final en el otro extremo, y aplicando la cinemática directa es posible conocer la posición y orientación final del efector con respecto a la base, si tenemos la posición de sus articulaciones (por lo general ángulos).

Aplicando el método, los eslabones del robot se enumeran comenzando por la base del brazo, hasta terminar por el otro extremo; la base será la junta 0, y seguirán el orden que les corresponde a los demás. Recordemos que cada eslabón está unido por medio de una articulación a otro eslabón, y en este caso las articulaciones serán descritas en sus movimientos más simples, y la función de cada eslabón es mantener una relación fija entre dos ejes de una articulación. La relación se puede describir con dos parámetros que son la longitud del eslabón [a] y su ángulo de torsión [α], hay veces que es difícil identificar estos parámetros, sobre todo en brazos robóticos con geometrías complejas.

Existe una convención conocida como *Método de Denavit Hartenberg* para poder describir la ubicación de cada eslabón con los eslabones adyacentes:

1. Identificar los ejes de acción en cada articulación que conformen la arquitectura del brazo robótico
2. Identifique los ejes con perpendicularidad en común con el i -ésimo eje, o identifique el punto de intersección de estos ejes.
3. Asigne el eje \hat{Z}_i para que apunte sobre el i -ésimo eje de la articulación, recuerde describir a cada articulación en sus movimientos más sencillos, es decir, lineal o angular.
4. Asigne el \hat{X}_i para que apunte sobre la perpendicular común, o si los ejes se cruzan entre sí, asigne \hat{X}_i para que sea normal al plano que contienen los dos ejes.
5. Asigne el eje \hat{Y}_i siguiendo la regla de la mano derecha para completar el sistema.
6. Asigne {0} para que concuerde con {1} cuando la primera variable de articulación sea cero. Para {N} seleccione la ubicación del origen y la dirección de \hat{X}_N libremente, pero generalmente de manera que haga que la mayor parte de los parámetros de los vínculos sea cero.

4.2. Algoritmo De Denavit Hartenberg

Nos auxiliaremos de la notación de Denavit – Hartenberg para describir la arquitectura de cada cadena cinemática del brazo como lo son la orientación y posición de ejes y articulaciones.

La notación de los parámetros a utilizar es:

- α_{i-1} : ángulo de \hat{Z}_i a \hat{Z}_{i+1} medida sobre el eje de \hat{X}_i
- a_{i-1} : distancia entre eje \hat{Z}_i a \hat{Z}_{i+1} medida sobre el eje de \hat{X}_i
- d_i : distancia entre el eje \hat{X}_{i-1} a \hat{X}_i medida sobre \hat{Z}_i
- θ_i : ángulo de \hat{X}_{i-1} a \hat{X}_i medida sobre el eje de \hat{Z}_i

En la siguiente tabla se muestran los parámetros obtenidos, ya con la consideración de cada uno de los sentidos de giro de los servomotores. También se agrega un pequeño eslabón extra para apreciar el último giro que se realiza en el efector final.

i	α_{i-1} [rad]	a_{i-1} [cm]	d_i [cm]	θ_i [rad]
1	$\frac{\pi}{2}$	Altura inicial =21.5	0	$-\frac{\pi}{2} - \theta_1$
2	0	0	Largo del brazo =15	θ_2
3	0	0	Largo del antebrazo=15	$\pi - \theta_3$
4	$\frac{\pi}{2}$	0	0	θ_4
5	0	Largo del efector=12	0	θ_5
6	0	0	Ancho del efector final=3	$\frac{\pi}{2}$

Tabla 1. Parámetros correspondientes al manipulador serial tipo antropomórfico. Ver anexo de programas.

El efector final tiene una ecuación cinemática sin tomar en cuenta el último giro del efector final. La obtención del siguiente par de ecuaciones esta descrito a fondo en los anexos.

$$\begin{pmatrix} 3 * \text{Sin}[\theta_1](-5 * \text{Cos}[\theta_2] + 5 * \text{Cos}[\theta_2 - \theta_3] + 4 * \text{Sin}[\theta_2 - \theta_3 + \theta_4]) \\ 3 * \text{Cos}[\theta_1](-5 * \text{Cos}[\theta_2] + 5 * \text{Cos}[\theta_2 - \theta_3] + 4 * \text{Sin}[\theta_2 - \theta_3 + \theta_4]) \\ 21.5 + 12 * \text{Cos}[\theta_2 - \theta_3 + \theta_4] + 15 * \text{Sin}[\theta_2] - 15 * \text{Sin}[\theta_2 - \theta_3] \end{pmatrix}$$

Ecuación 2. Cinemática del efector final sin el último grado de libertad.

Considerando el último giro del efector final $[\theta_5]$ con una desviación de 3 [cm] con respecto al centro del gripper, queda como:

$$\begin{pmatrix} -3(\text{Cos}[\theta_1]\text{Cos}[\theta_5] + \frac{1}{2}\text{Sin}[\theta_1](10\text{Cos}[\theta_2] - 10\text{Cos}[\theta_2 - \theta_3] - 8\text{Sin}[\theta_2 - \theta_3 + \theta_4] - \text{Sin}[\theta_2 - \theta_3 + \theta_4 - \theta_5] + \text{Sin}[\theta_2 - \theta_3 + \theta_4 + \theta_5])) \\ 3(\text{Cos}[\theta_5]\text{Sin}[\theta_1] + \text{Cos}[\theta_1](5\text{Cos}[\theta_2](-1 + \text{Cos}[\theta_3]) + 5\text{Sin}[\theta_2]\text{Sin}[\theta_3] + 4\text{Sin}[\theta_2 - \theta_3 + \theta_4] - \text{Cos}[\theta_2 - \theta_3 + \theta_4]\text{Sin}[\theta_5])) \\ 21.5 + 12 * \text{Cos}[\theta_2 - \theta_3 + \theta_4] + 1.5\text{Cos}[\theta_2 - \theta_3 + \theta_4 - \theta_5] - 1.5\text{Cos}[\theta_2 - \theta_3 + \theta_4 + \theta_5] + 15 * \text{Sin}[\theta_2] - 15 * \text{Sin}[\theta_2 - \theta_3] \end{pmatrix}$$

Ecuación 3. Cinemática del efector final, considerando el último grado de libertad y una desviación de 3 cm.

Es necesario hacer hincapié en la diferencia de ambas ecuaciones, debido a que el modelado matemático considera al efector final como un punto por lo que anula el último giro, y fue necesario agregar una desviación para poder observar el giro del efector final.

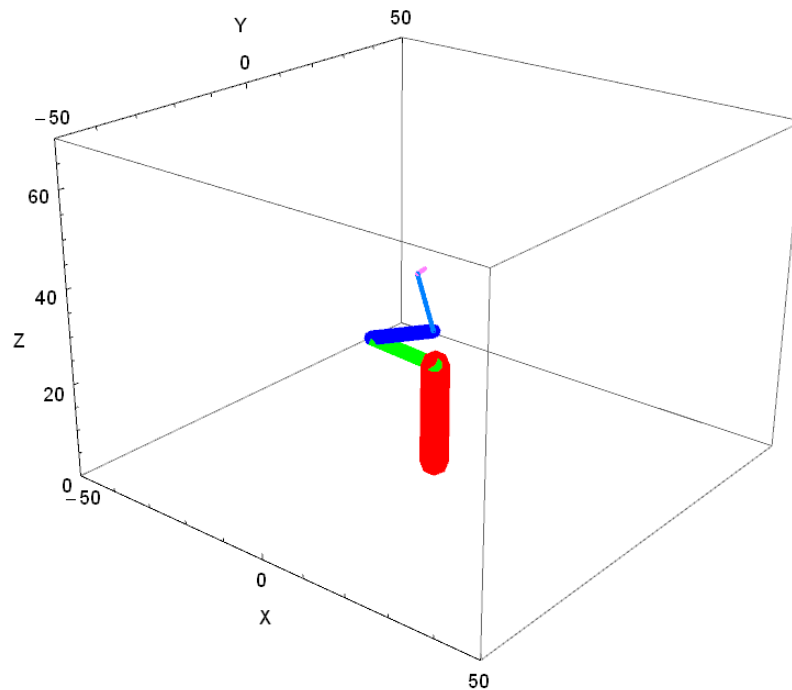


Ilustración 12. Aplicación del método en el brazo antropomórfico, en el programa Mathematica 9. Ver anexo de programas.

4.3. Cinemática inversa del manipulador serial

El cálculo de la cinemática inversa se refiere al conjunto de ángulos necesarios para hacer que el efector final se mueva de una posición y orientación inicial a una orientación y posición final deseada. Este tipo de cálculos lo hacemos de manera inconsciente al mover nuestra mano de un lado a otro. Pero en el caso de los robots seriales es necesario la realización de algoritmos computacionales para poder controlarlos, aunque este no se vuelve el único medio para controlar un brazo serial, debido a que también se puede controlar de manera manual eligiendo los ángulos que se deseen en cada articulación.

El cálculo de la cinemática inversa suele complicarse, debido a la existencia de múltiples soluciones y también a las ecuaciones cinemáticas debido a que no son lineales en el caso de robots redundantes como este, ya que poseen un mayor número de grados de libertad a los necesarios para posicionarse y orientarse en un espacio de trabajo. Muchas veces la solución de las ecuaciones no es posible, en parte a la convergencia del método numérico a utilizar. En el caso de que la solución no sea posible significa que no es posible obtener la posición y orientación deseadas, o que estas se encuentran fuera del espacio de trabajo o dentro de una singularidad.

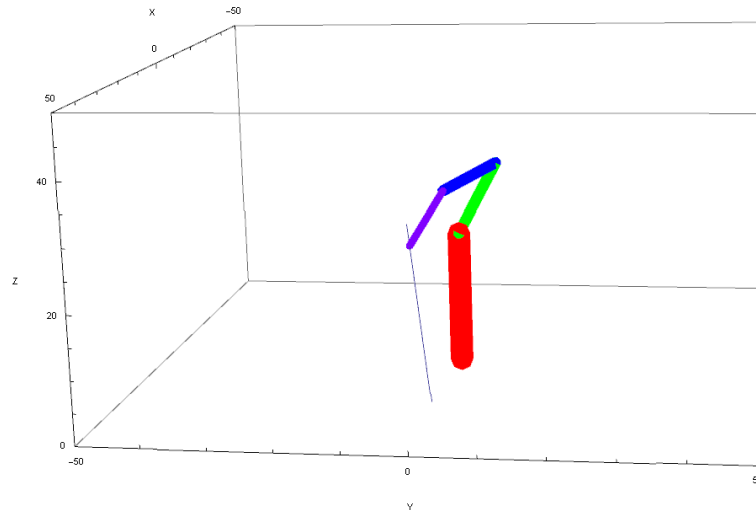
También es posible que el punto que se quiera alcanzar este dentro del espacio de trabajo pero sea una singularidad que depende de la parte física, y en estas singularidades los actuadores se fuerzan mucho. Una analogía con el cuerpo humano es tratar de cargar una masa de 5 [kg] a la altura del hombro con solo un brazo, esto es posible para la mayoría de las personas adultas, pero si la condición es levantar la misma masa con el brazo extendido se vuelve un reto y puede que no sea posible para algunas personas.

La cinemática inversa del manipulador móvil fue calculada con un perfil de trayectoria por el software Mathematica 9, que utiliza el método numérico de “Newton-Raphson”, se consideraron 50 iteraciones para asegurar una buena aproximación del método numérico. La trayectoria del manipulador se discretizó en 128 partes, lo anterior haciendo una homología con la tarjeta MD25, la idea es manejar 128 posiciones de ida y 128 posiciones de regreso, de esta forma se tienen 256 posiciones diferentes en todo el recorrido.

4.4. Perfil Geométrico

Para generar una trayectoria específica a seguir por un brazo serial es necesario definir puntos que describan a grosso modo la trayectoria deseada, sobre todo si la trayectoria que se busca es muy compleja.

El perfil geométrico que ha sido definido para que el manipulador recoja un objeto será una línea recta que está definida por un par de puntos, uno de ellos será un punto inicial donde el manipulador estará retraído en su posición de reposo, y el segundo punto será donde se encuentre el objeto a manipular o donde se vaya a depositar. Esta es el perfil de trayectoria de ida.



*Ilustración 13. trayectoria de ida que seguirá el manipulador. *Articulación $[\theta_2]$ (unión de eslabones rojo y verde)
*Articulación $[\theta_3]$ (unión de eslabones azul y verde) *Articulación $[\theta_1]$ (rotación de la barra roja sobre su eje en Z)*

Para el regreso se han definido dos perfiles geométricos que llegan al punto de reposo. Fue necesario hacer un perfil de trayectoria combinado, debido a que el par requerido para levantar el brazo era muy considerable y los servos en la articulación $[\theta_2]$ no alcanzaban a proporcionar el par indicado. El primer perfil de regreso comienza reduciendo el par necesario que necesita el brazo, y en el segundo perfil se busca que el brazo se posicione en su posición de reposo.

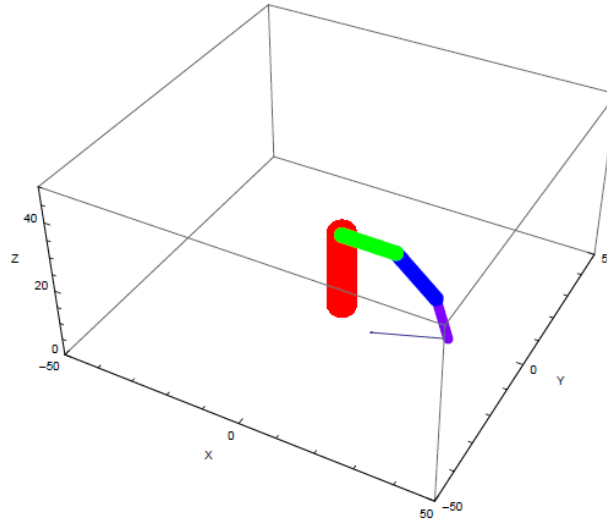


Ilustración 14. Primer perfil geométrico de regreso.

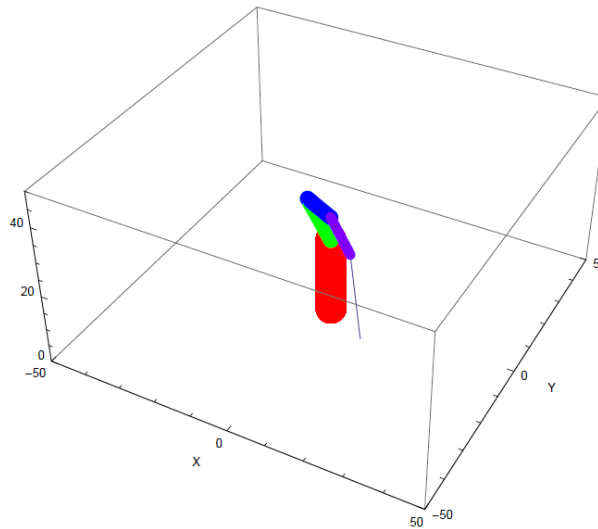


Ilustración 15. Segundo perfil geométrico de regreso.

4.4.1 Perfil de trayectoria

La trayectoria discretizada se utiliza para calcular la posición de cada articulación en los distintos puntos que conforman el perfil de trayectoria, en este caso es una recta que se discretizó en 128 partes. Se buscó un movimiento del manipulador natural. Con las restricciones de velocidad del efector final de cero en el punto inicial y final de la trayectoria.

Se calcularon distintos perfiles de trayectoria que permiten al manipulador moverse de manera más natural, y evitan cambios bruscos de velocidades:

- Perfil armónico simple o Cosenoidal o perfil armónico Simple.

$$P(t) = P_0 + \frac{1}{2} \left(1 - \cos \frac{\pi t}{t_f} \right) (P_f - P_0) \quad \text{en } 0 \leq t \leq t_f$$

Ecuación 4. Perfil de velocidad, armonio simple, ver referencia 19.

- Perfil Cúbico.

$$P(t) = P_0 + \left(\frac{3t^2}{t_f^2} - \frac{2t^3}{t_f^3} \right) (P_f - P_0) \quad \text{en } 0 \leq t \leq t_f$$

Ecuación 5. Perfil de velocidad, Cúbico, ver referencia 19.

- Perfil Parabólico. Se conforma de dos parábolas, la primer parábola tiene dominio en $0 \leq t \leq (0.5t_f)$ y la segunda parábola en $(0.5t_f) \leq t \leq t_f$.

$$P(t) = P_0 + \left(\frac{2t^2}{t_f^2} \right) (P_f - P_0) \quad \text{en } 0 \leq t \leq \frac{t_f}{2}$$

Ecuación 6. Perfil de velocidad, Parabólico (mitad 1), ver referencia 19.

$$P(t) = P_f + \left(\frac{4t}{t_f} - \frac{2t^2}{t_f^2} - 2 \right) (P_f - P_0) \quad \text{en } \frac{t_f}{2} \leq t \leq t_f$$

Ecuación 7. Perfil de velocidad, Parabólico (mitad 2), ver referencia a 19.

- Perfil Cíclico o Senoidal.

$$P(t) = P_0 + \left(\frac{t}{t_f} - \frac{1}{2} \sin \frac{2\pi t}{t_f} \right) (P_f - P_0) \quad \text{en } 0 \leq t \leq t_f$$

Ecuación 8. Perfil de velocidad, Cíclico, ver referencia 19.

- Perfil con un polinomio de quinto grado.

$$P(t) = P_0 + \left(10 \frac{t^3}{t_f^3} - 15 \frac{t^4}{t_f^4} + 6 \frac{t^5}{t_f^5} \right) (P_f - P_0) \quad \text{en } 0 \leq t \leq t_f$$

Ecuación 9. Perfil de velocidad, Polinomio de quinto grado, ver referencia 19.

Una vez aplicado el cálculo de la trayectoria con cada uno de los distintos perfiles se encontró que el perfil de trayectoria cubico es el más amigable con los actuadores presentando menos picos de aceleraciones que se traduce en un menor estrés para los actuadores y de acuerdo con el autor S.A. Alshahrani [19], este perfil de velocidad es el que presenta un menor gasto energético si el movimiento del manipulador no se desarrolla en un periodo de tiempo muy largo.

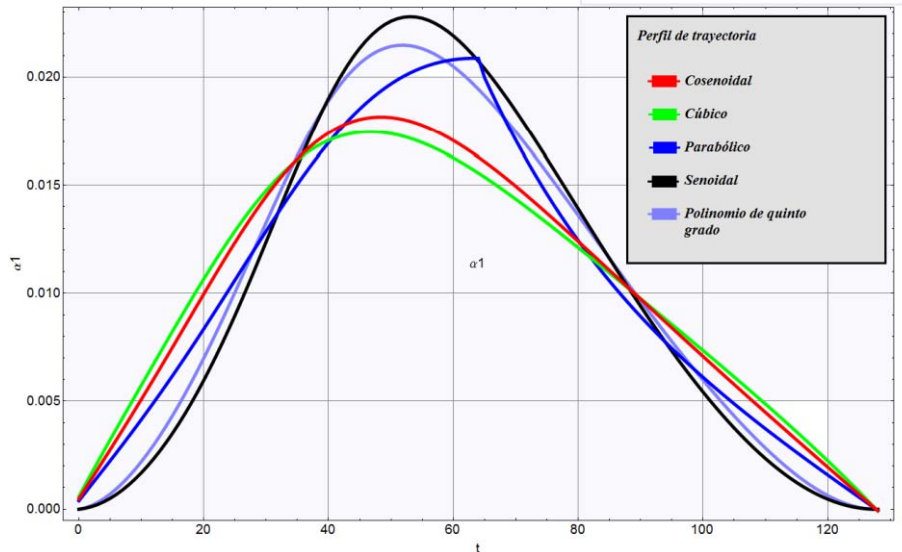


Ilustración 16. Grafica de las aceleraciones de cada uno de los distintos perfiles de velocidad. Se destaca la menor amplitud del perfil cúbico y una mayor suavidad en el trazo de la curva. Las unidades de la gráfica son [rad/s²] vs [s].

Perfil de trayectoria	Magnitud de aceleración de la articulación a_1 [rad/seg ²]
Cosenoidal	0.0181405
Cúbico	<u>0.0174623</u>
Parabólico	0.0208698
Senoidal	0.0227965
Polinomio de quinto grado	0.0214783

Tabla 2. Muestra las magnitudes de aceleración de cada perfil de trayectoria para la junta 1, se destaca la menor magnitud del perfil cúbico. Es necesario mencionar que en la parte experimental la posición del brazo no dependerá del tiempo, si no de la posición. Por lo que las unidades de aceleración estarán en función de la posición de los objetos y no en función del tiempo.

Para poder obtener cada una de las distintas posiciones a lo largo de la trayectoria fue necesario que la articulación [04] mantuviera un ángulo fijo, en este caso se asignó un ángulo de 120[°]. De lo contrario se tenía un sistema de ecuaciones con múltiples soluciones. El movimiento de la articulación θ_4 , tendrá un movimiento que ha sido definido aleatoriamente con la condición que al final se posicione en 120[°], de esta forma el efector final llegará al lugar deseado.

La obtención de la cinemática inversa nos dio la posición de cada uno de los ángulos de las tres articulaciones $[\theta_1, \theta_2, \theta_3]$ en las 256 posiciones diferentes, y para poder manejar los datos adecuadamente se optó por una regresión polinómica en cada uno de los ángulos, de esta forma el movimiento del brazo serial se reduce a ecuaciones de posición angular que estarán en función de la distancia de la plataforma móvil a la meta deseada, este punto se tocará más a fondo en el capítulo 6.

Es necesario mencionar que las ecuaciones de posición angular permiten un movimiento más fluido, ya que admiten valores racionales. De esta forma el manipulador se mueve de manera más fluida.

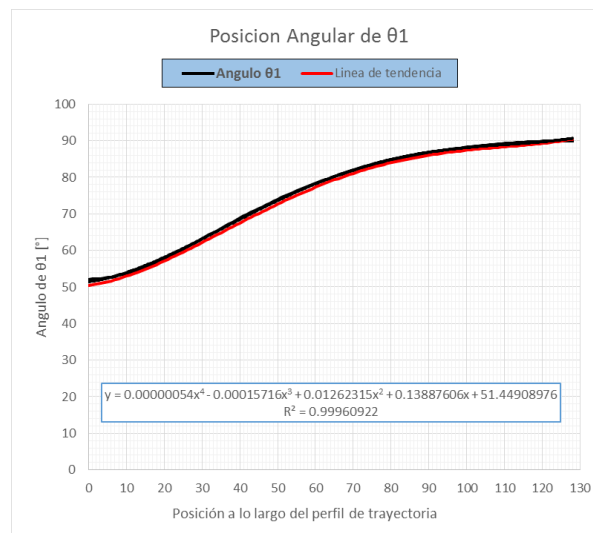


Ilustración 17. Variación de la posición angular de manera paramétrica en la articulación θ_1 . A lo largo del perfil geométrico de ida. La exageración de los decimales calculados fue necesaria. .

Se recorrió a una interpolación polinómica en este caso de cuarto grado, debido a que se busca la minimización del error de posición con respecto a los datos calculados. Por lo que se resalta en la imagen anterior que el coeficiente de determinación muestral $[R^2]$ tiene un valor muy cercano a uno. Lo que indica que el modelo matemático obtenido por la regresión polinómica describe muy bien el comportamiento angular de $[\theta_1]$ en este caso. Hecho lo anterior con cada uno de los comportamientos angulares de las articulaciones se muestran las gráficas de cada uno de los movimientos a lo largo de los tres perfiles geométricos.

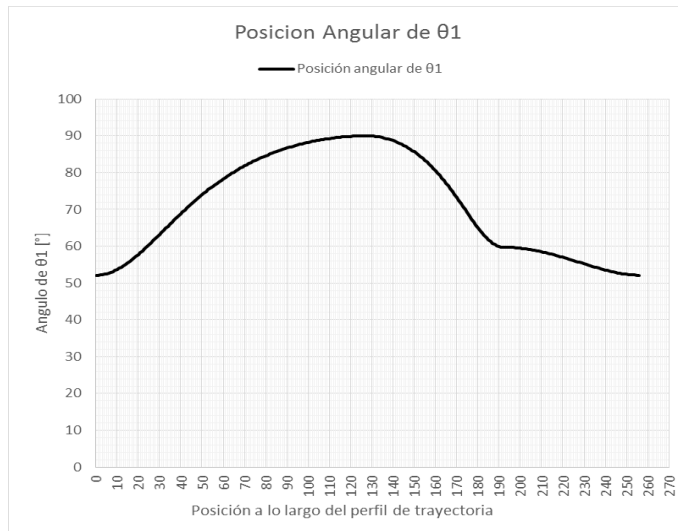


Ilustración 18. Variación de la posición angular de manera paramétrica en la articulación θ_1 .

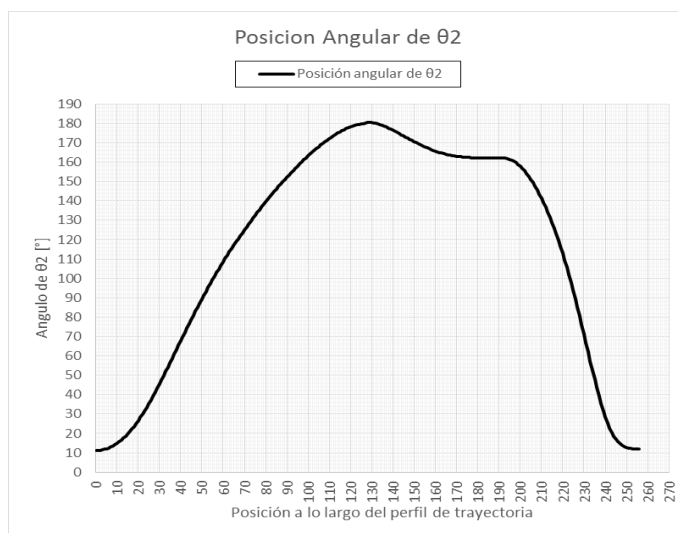


Ilustración 19. Variación de la posición angular de manera paramétrica en la articulación θ_2 .

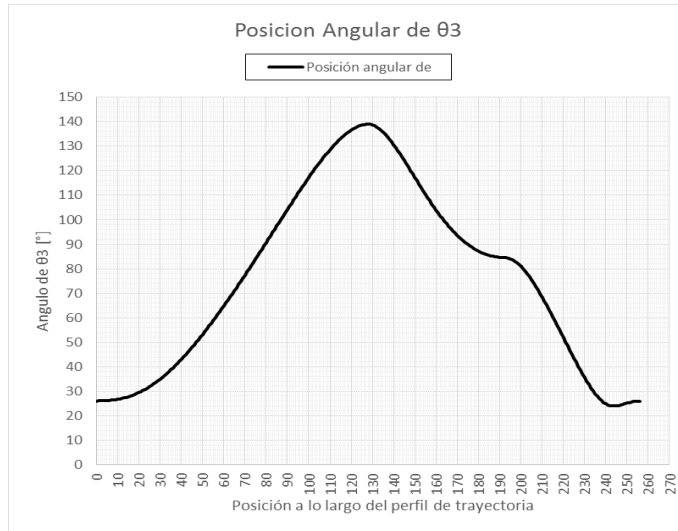


Ilustración 20. Variación de la posición angular de manera paramétrica en la articulación θ_3 .

De cada una de las gráficas es necesario destacar que el movimiento a lo largo de los perfiles geométricos se realiza de manera paramétrica y la posición angular estará en función de la posición de la plataforma móvil respecto a la meta deseada, y no en función del tiempo.

El recorrido del manipulador a lo largo de los tres perfiles geométricos está definido por tres ecuaciones de movimiento en cada una de las tres articulaciones principales (θ_1 , θ_2 , θ_3). Para las ecuaciones con dominio $[128,192]$ y $[192,255]$ es necesario hacer un cambio de variable porque el cálculo de la regresión polinómica se realizó con los valores iniciales.

- Para la articulación θ_1

- $\theta_1(x) = 0.00000054x^4 - 0.00015716x^3 + 0.01262315x^2 + 0.13887606x + 51.44908976$
 - ♦ $D_{\theta_1}: x \in [0,128)$

Ecuación 10. Ecuación de movimiento de la junta 1, perfil de ida.

- $\theta_1(h) = 0.00000581h^4 - 0.00057874h^3 + 0.00838981h^2 - 0.17274787h + 90.34282122$
 - ♦ $D_{\theta_1}: x \in [128,192)$
 - ♦ $h = x - 128$

Ecuación 11. Ecuación de movimiento junta 1, perfil de regreso 1. Incluye cambio de variable

- $\theta_1(u) = 0.000057u^3 - 0.005883u^2 + 0.023755u + 59.664474$
 - ♦ $D_{\theta_1}: x \in [192,255]$
 - ♦ $u = x - 192$

Ecuación 12. Ecuación de movimiento junta 1, perfil de regreso 2. Incluye cambio de variable

- Para la articulación θ_2

- $\theta_2(x) = 0.00000186x^4 - 0.00059158x^3 + 0.05568245x^2 + 0.01765028x + 9.91542937$
 - ♦ $D_{\theta_2}: x \in [0,128]$

Ecuación 13. Ecuación de movimiento junta 2, perfil ida

- $\theta_2(h) = -0.00000584h^4 + 0.00084902h^3 - 0.03414534h^2 - 0.05395999h + 180.71653661$
 - ♦ $D_{\theta_2}: x \in [128,192]$
 - ♦ $h = x - 128$

Ecuación 14. Ecuación de movimiento junta 2, perfil de regreso 1. Incluye cambio de variable

- $\theta_2(u) = 0.00003126u^4 - 0.00225163u^3 - 0.01455054u^2 - 0.31165647u + 162.26559778$
 - ♦ $D_{\theta_2}: x \in [192,255]$
 - ♦ $u = x - 192$

Ecuación 15. Ecuación de movimiento junta 2, perfil de regreso 2. Incluye cambio de variable

- Para la articulación θ_3

- $\theta_3(x) = -0.00000047x^4 + 0.00002650x^3 + 0.01180990x^2 - 0.05908377x + 26.01063203$
 - ♦ $D_{\theta_3}: x \in [0,128]$

Ecuación 16. Ecuación de movimiento junta 3, perfil ida

- $\theta_3(h) = 0.000364h^3 - 0.026422h^2 - 0.656364h + 140.855179$
 - ♦ $D_{\theta_3}: x \in [128,192]$
 - ♦ $h = x - 128$

Ecuación 17. Ecuación de movimiento junta 3, perfil de regreso 1. Incluye cambio de variable

- $\theta_3(u) = 0.000727u^3 - 0.058109u^2 - 0.133778u + 85.140285$
 - ♦ $D_{\theta_3}: x \in [192,255]$
 - ♦ $u = x - 192$

Ecuación 18. Ecuación de movimiento junta 3, perfil de regreso 2. Incluye cambio de variable

- Para la articulación θ_4 , en el caso particular de esta articulación se propone un movimiento suave, en el cálculo de la cinemática inversa esta articulación se fijó con un ángulo de 120° , pero en la aplicación práctica se propone un movimiento suave que inicie en 90° y llegue a 120° .

- $\theta_4(x) = 0.234375 * u + 90$
 - ♦ $D_{\theta_4}: x \in [0,128]$

Ecuación 19. Ecuación de movimiento para la junta 4.

- $\theta_4(x) = -0.23622x + 150.23622$
 - ♦ $D_{\theta_4}; x \in [128,255]$

Ecuación 20. Ecuación de movimiento para la junta cuatro perfil de regreso

La cinemática inversa con la que se decidió mover el manipulador, tenía problemas debido a que el par de torsión proporcionado por los servos no era el suficiente, y la etapa de potencia sufría sobrecalentamiento. Por lo que fue necesario cambiar los servomotores a unos con mayor par de torsión. Para un posicionamiento adecuado de la plataforma móvil fue necesario diseñar una estructura adecuada para poner el fiducial. Para mayor detalle ver los anexos.

4.5. Plataforma del robot móvil

El robot híbrido elegido posee una plataforma móvil en configuración diferencial, bajo las consideraciones de que el robot es un cuerpo rígido no deformable, las llantas no presentan deslizamientos y su eje de rotación es paralelo al plano de trabajo.

El modelado de la plataforma móvil posee dos sistemas coordenados, un sistema inercial fijado en la plataforma móvil y un sistema de referencia absoluto, en este caso definido por el sistema de visión.

La ubicación de la plataforma móvil dentro del sistema de referencia inercial es descrita por el vector:

$$\xi = [x \quad y \quad \theta]^T$$

Ecuación 21. Plataforma móvil dentro del sistema inercial

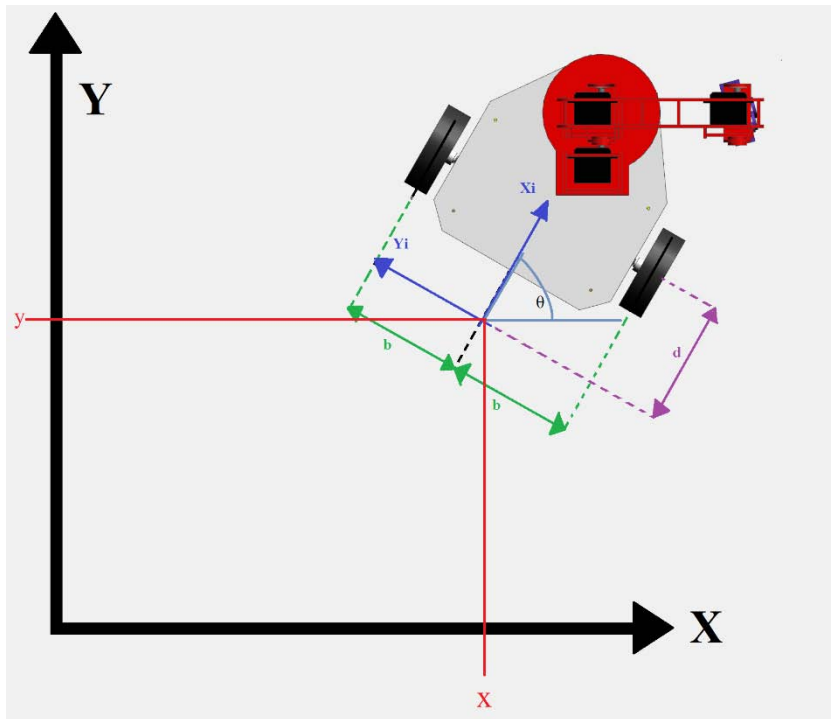


Ilustración 21. Sistemas de referencia para el modelado de la plataforma móvil.

La matriz de rotación que describe la ubicación angular del sistema inercial dentro del sistema de referencia absoluto

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

Ecuación 22. Matriz de rotación

La velocidad de la plataforma móvil dentro del sistema inercial de la plataforma móvil es descrita por el vector:

$$\xi_1 = [\dot{x} \quad \dot{y} \quad \dot{\theta}]^T$$

Ecuación 23. Velocidad plataforma

El siguiente vector representa la velocidad del sistema inercial dentro del sistema absoluto de referencia. Es necesario señalar que la velocidad angular del sistema inercial $[\dot{\theta}]$ se expresa solo dentro del sistema inercial.

$$\dot{\xi} = R^T(\theta)\dot{\xi}_1$$

Ecuación 24. Velocidad del sistema inercial

Por propagación de velocidades calcularemos la velocidad de las llantas, la ubicación de las llantas dentro del sistema inercial tiene por coordenadas ($\pm d$, $\pm b$) según sea el caso.

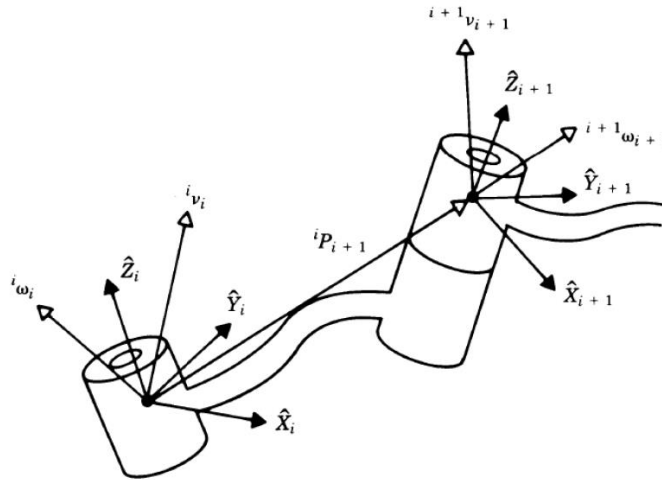


Ilustración 22. Vectores de velocidad de un par de eslabones que se encuentran unidos.

Propagación de velocidades establece que la velocidad angular del eslabón $i+1$, es la velocidad angular del eslabón i más la adición de la componente que agregue el eslabón $i+1$. Matemáticamente:

$${}^{i+1}\omega = {}^{i+1}R^i \omega_i + \dot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1}$$

Ecuación 25. Propagación de velocidades, ver bibliografía 20

En el caso de la velocidad lineal en el eslabón $i+1$ será la misma del eslabón i más la componente de la velocidad angular del eslabón en i . Matemáticamente

$${}^i v_{i+1} = {}^i v_i + {}^i \omega_i \times {}^i P_{i+1}$$

Ecuación 26. Propagación de velocidades. Ver bibliografía 20

Tomando en cuenta propagación de velocidades para calcular el vector ξ_1 . Y considerando la ubicación de las llantas descrita. Se tiene que:

$${}^1\omega_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix}$$

Ecuación 27. Velocidad angular de la plataforma

Para el caso de la llanta derecha

$${}^1P_{2D} = \begin{bmatrix} -d \\ -b \\ 0 \end{bmatrix}$$

Ecuación 28. Propagación de velocidades llanta derecha

Para el caso de la llanta izquierda

$${}^1P_{2I} = \begin{bmatrix} -d \\ +b \\ 0 \end{bmatrix}$$

Ecuación 29. Propagación de velocidades llanta izquierda

Propagando las velocidades a cada extremo donde se encuentran las llantas, en el presente caso es del sistema inercial a la llanta derecha y del sistema inercial a la llanta izquierda.

La velocidad del sistema inercial.

$${}^1v_1 = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta \\ \dot{y} \cos \theta - \dot{x} \sin \theta \\ 0 \end{bmatrix}$$

Ecuación 30. Velocidad del sistema inercial

Formula a aplicar

$${}^i v_{i+1} = {}^i v_i + {}^i \omega_i \times {}^i P_{i+1}$$

Ecuación 31. Propagación de velocidades 3

Aplicando la formula en la rueda derecha

$${}^2v_{2D} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta \\ \dot{y} \cos \theta - \dot{x} \sin \theta \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix} \times \begin{bmatrix} -d \\ -b \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta + b\dot{\theta} \\ \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} \\ 0 \end{bmatrix}$$

Ecuación 32. Propagación de velocidades en rueda derecha

Aplicando la formula en la rueda izquierda

$${}^2v_{2l} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta \\ \dot{y} \cos \theta - \dot{x} \sin \theta \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix} \times \begin{bmatrix} -d \\ +b \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta - b\dot{\theta} \\ \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} \\ 0 \end{bmatrix}$$

Ecuación 33. Propagación de velocidades en rueda izquierda

La velocidad lineal de una rueda se puede obtener en base a su velocidad angular $[\dot{\theta}_d]$ y a su radio $[r]$.

Matemáticamente para la rueda derecha:

$$\begin{bmatrix} r \dot{\theta}_d \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta + b\dot{\theta} \\ \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} \\ 0 \end{bmatrix}$$

Ecuación 34. Resumen, velocidad de la rueda derecha

Matemáticamente para la rueda izquierda;

$$\begin{bmatrix} r \dot{\theta}_i \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta - b\dot{\theta} \\ \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} \\ 0 \end{bmatrix}$$

Ecuación 35 Resumen, velocidad de la rueda izquierda

La plataforma móvil que se tiene, es una plataforma no holonómica debido a que puede tener desplazamientos sobre su eje de simetría y también puede rotar, pero la configuración de ruedas impide que tenga un desplazamiento lateral. Para poder realizar un movimiento lateral primero tendría que rotar la plataforma para poder después desplazarse en forma recta. De esta restricción cinemática se establece la ecuación:

$$0 = \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta}$$

Ecuación 36. Restricciones cinemáticas de la plataforma móvil

Debido a la restricción cinemática por propagación de velocidades solo nos queda la componente en x de ambas ruedas.

Para la rueda derecha:

$$r \dot{\theta}_d = \dot{x} \cos \theta + \dot{y} \sin \theta + b\dot{\theta}$$

Ecuación 37. Velocidad rueda derecha

Para la rueda izquierda:

$$r \dot{\theta}_i = \dot{x} \cos \theta + \dot{y} \sin \theta - b\dot{\theta}$$

Ecuación 38. Velocidad rueda izquierda

Diseño del espacio inteligente

5.1. Enfoque conceptual

Es necesario definir conceptualmente la meta a lograr en el desarrollo del espacio inteligente.

En este capítulo se buscará diseñar el espacio de interacción donde coexistirán tres elementos clave en el desarrollo del presente trabajo de tesis: el manipulador móvil, el espacio inteligente y el ser humano.

5.2. Espacio de trabajo disponible

El área de trabajo en el plano (X-Y) está delimitada por el campo de visión de la cámara, la cual está posicionada a dos metros y medio del piso sobre el eje Z, por lo que hay un cono de visión efectivo. En el caso de esta cámara se forma un área máxima que tiene una longitud de 2.4 [m] y un ancho de 1.8 [m].

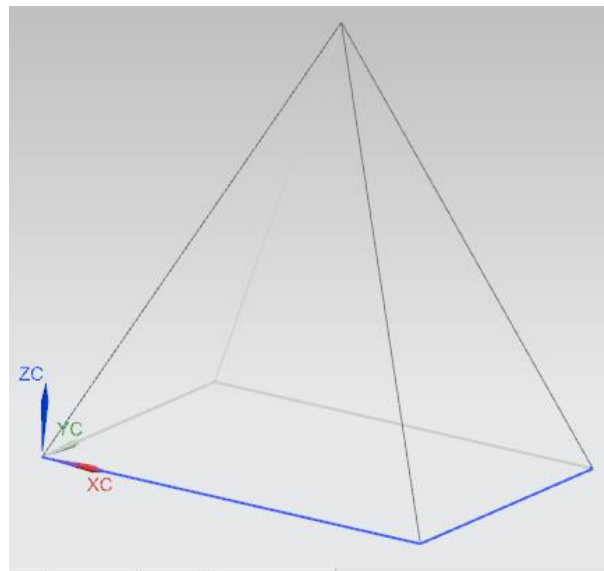


Ilustración 23. Pirámide cuadrangular que ejemplifica el campo de visión de la cámara.

Es necesario tener en cuenta que el diseño de mobiliario dentro del espacio de trabajo reducirá el área de trabajo. Por ejemplo en el caso de un objeto que este a 25 [cm] del suelo el área de trabajo reduce un diecinueve por ciento, en el caso del largo y ancho solo se ven reducidos en solo diez por ciento.

5.2.1 Calibración de la cámara

La cámara de visión utilizada es de alta definición y con un mínimo de retraso, es vital este tipo de características para que el manipulador móvil este controlado en tiempo real. Existe una deformación presente en el tipo de visión unifocal sobre todo en la periferia del área de visión, el software de la cámara hace una compensación de este tipo de deformaciones, en el caso del presente trabajo solo hay que asegurar que el plano de visión de la cámara tenga una base rectangular en el piso para evitar una mala compensación del software y tener una mala lectura de los datos.

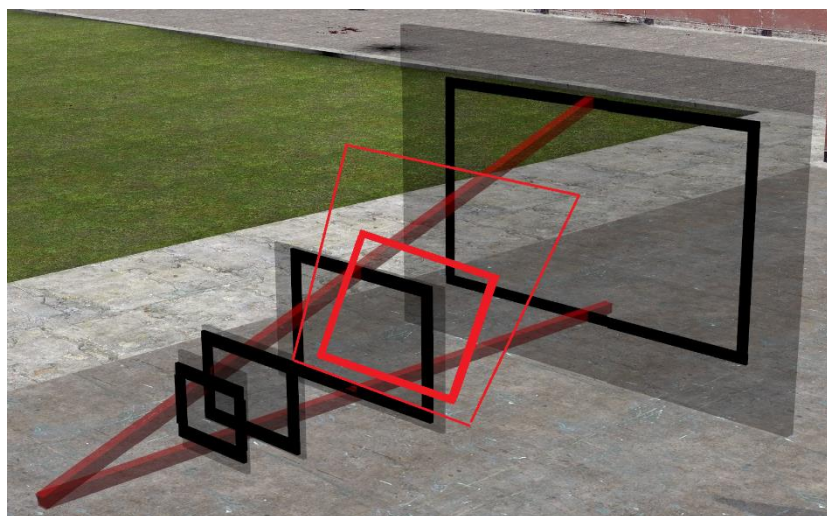


Ilustración 24. En la imagen se muestra un área de visión errónea (en color rojo). Se debe garantizar un área rectangular de visión rectangular paralela al piso para evitar deformaciones indeseadas.

5.3. Diseño conceptual del espacio inteligente

El diseño del espacio inteligente está pensado en una oficina del edificio del CIA (Centro de Ingeniería Avanzada), se contemplarán dentro del diseño un par de elementos estáticos que no sean manipulables por el robot móvil como mesas, y un elemento manipulable para el robot.

El diseño contempla al menos dos mesas rectangulares como elementos estáticos no manipulables para el robot, pero que pueden cambiarse de ubicación espacial. Se contempla una pared como elemento estático no manipulable e inamovible. El objeto a transportar es una pequeña pieza.

5.3.1 Materiales

El mobiliario diseñado será simulado en una primera etapa de pruebas. Para futuras pruebas puede que sea necesario la manufactura de los modelos representativos. Y se ha pensado en utilizar mdf (Medium Density Fibreboard), empleado habitualmente en muebles de oficinas. Es un material que reemplaza a la madera en algunas aplicaciones, suele ser fácil de manufactura y bajo costo. El mdf no es transparente, y esto puede interferir con los objetivos de la tesis, debido a que utilizamos un sistema de visión antes mencionado, y al obstruir el campo de visión de la cámara, el espacio inteligente dejaría de ver algunas regiones que pueden ser clave para el reconocimiento de algún patrón.

Como segunda opción para manufacturar el mobiliario y poder ver a través del material, se ha considerado la manufactura en acrílico (polimetilmetacrilato). Es el polímero con mayor transparencia encontrado habitualmente en los faros de los carros. Es resistente a golpes y es de fácil manufactura. Y en comparación con el cristal muy ligero y es más resistente.

5.3.2 Tamaños

Se diseñarán modelos a escala que permitan una primera iteración dentro del espacio de trabajo, es necesario recordar que el espacio de trabajo disponible está limitado por el campo de visión de la cámara en el plano de trabajo X-Y (suelo). El robot móvil se vuelve un factor importante para definir la escala, ya que se busca que los objetos a manipular estén dentro del espacio de trabajo del manipulador, por lo tanto la escala definida es de 1:3, y fue definida en base a la altura del manipulador.

5.3.3 Arquitectura

Es necesario que la estructura del espacio que permita la adición y exclusión de elementos estáticos y dinámicos. Así como una reconfiguración del espacio de trabajo sin problema alguno. Lo anterior es un requisito indispensable para que en un futuro el modelo pueda ser escalado.

5.3.4 Sistema de comunicaciones

La capa de comunicaciones está compuesta de dos partes, una parte alámbrica que es la que comunica al sistema de visión con la computadora que es el sistema central de procesamiento, y la otra capa de comunicación se realiza en una red inalámbrica WLAN (Wireless local area network), que sirve para comunicar al robot móvil con la unidad de procesamiento.

Se eligió una red WLAN de este tipo porque tiene un ancho de banda mayor que los sistemas de transmisión, como un sistema de bluetooth o un sistema de radiocontrol (Zigbee).

5.3.5 Sistema de visión

El sistema de visión es parte de la capa de percepción pasiva ya que es un sensor fijo dentro del espacio de trabajo y no tiene una interacción con el usuario.

En el presente trabajo se usará un sistema de visión artificial externo al robot para que el robot perciba al ambiente independientemente de la tarea que este ejecute, por lo que se optó por un sistema de visión en reacTIVision. Este último es un software que permite conocer la posición (x, y), orientación (θ), velocidad, centroide, identificador (ID), de los fiducials. ReacTIVision es un software de uso libre que manda mensajes TUIO vía UDP puerto 3333 a cualquier aplicación con cliente TUIO. Se destaca que el espacio de trabajo está limitado por el campo de visión de la cámara.

El sistema de visión utiliza una cámara externa que ve el campo de trabajo, la imagen es convertida en blanco y negro para crear mayor contraste en la imagen, lo anterior por un algoritmo de umbrales adaptable. La imagen se divide en regiones negras y blancas de forma alternada que forman una gráfica. En el gráfico generado se buscan secuencias codificadas, estas secuencias codificadas son los fiducials. Y por medio de una base de datos se conoce el identificador y se calcula la información de cada fiducial. Después la información obtenida de cada fiducial es enviada a través del protocolo TUIO, que más tarde será la información que se necesitara en Simulink.

Para detectar algún objeto en el espacio de trabajo es necesario que tenga un fiducial visible para la cámara.

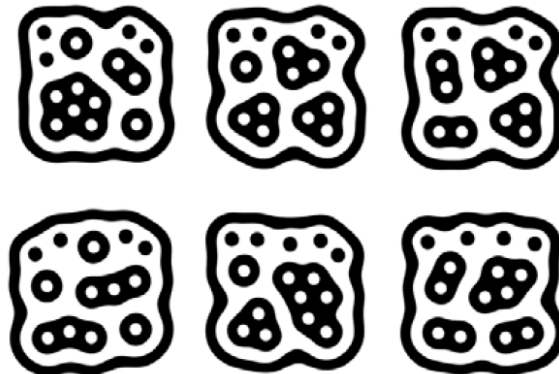


Ilustración 25. Imágenes de distintos fiduciales

Una desventaja de este sistema de visión es el reto de detectar espacialmente (x, y, z) un objeto dentro del espacio de trabajo, ya que ReactIVision solo es capaz de proporcionarnos las coordenadas en el plano (x, y), por lo que la coordenada en el eje Z será obtenida de una base de datos previamente hecha.

Se utilizaron fiduciales porque es un sistema de visión robusto y puede soportar perturbaciones siempre y cuando se vea el fiducial, las deformaciones debido a la cámara pueden ser compensadas, y los cambios de iluminación no afectan al sistema. Las rotaciones y traslaciones que puedan sufrir los diferentes objetos son fácilmente detectables, el fondo puede ser relativamente complejo sin que nos afecte, siempre y cuando la superficie de operación sea plana y también pueden ser utilizados en tiempo real.

Este sistema de visión es parte de la etapa experimental en el desarrollo de espacios inteligentes, pero se plantea la posibilidad de cambiar este tipo de sistema por otra tecnología diseñada específicamente para el robot.

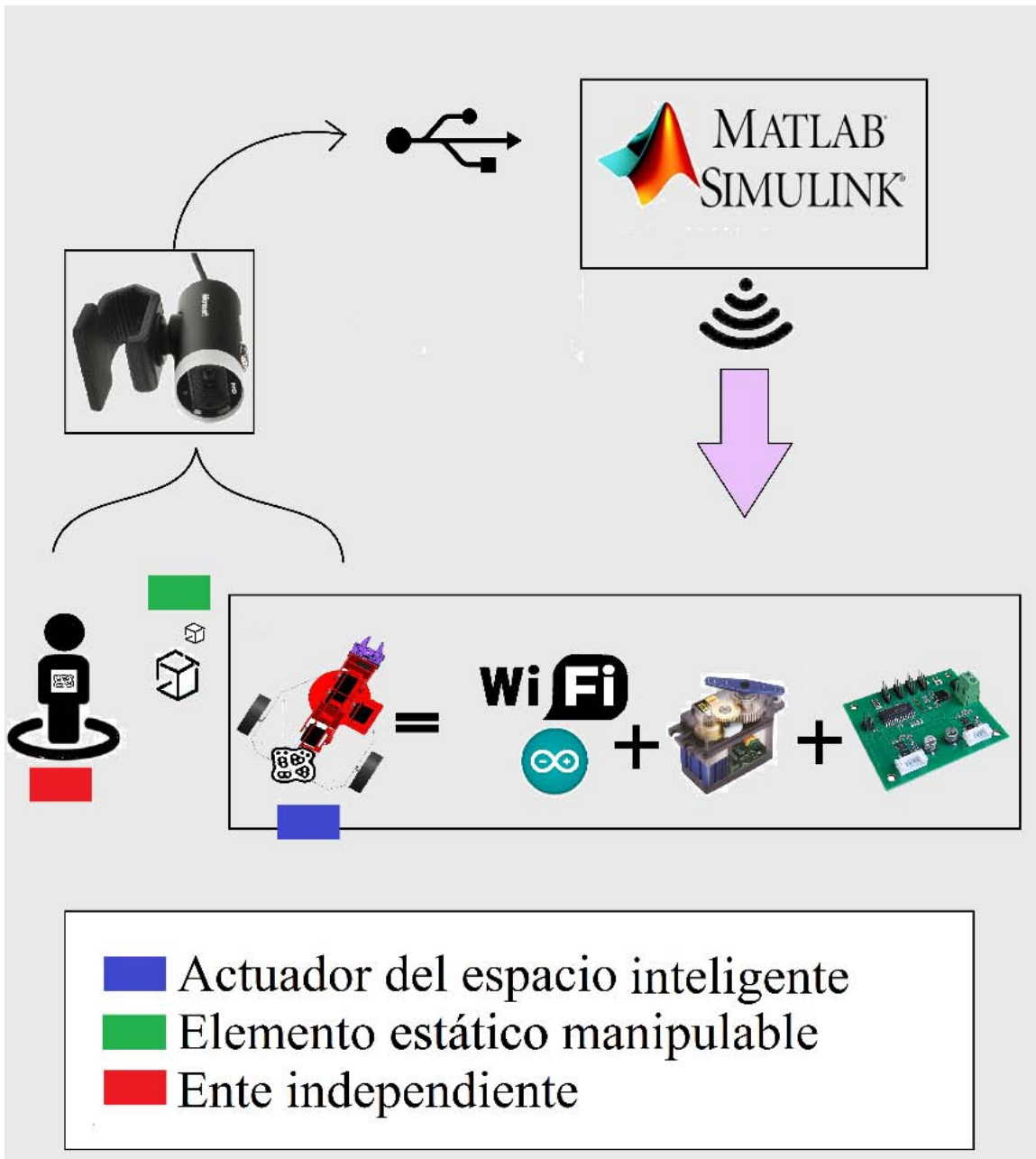


Ilustración 26. Esquema representativo del funcionamiento del espacio inteligente.

Planeación de movimientos dentro del espacio inteligente

6.1. Método de campos potenciales

El método de campos potenciales es una herramienta que permitirá al manipulador móvil transportarse de un punto a otro evadiendo obstáculos, conociendo únicamente su modelo cinemático. Lo anterior es posible tomando los campos potenciales como vectores que representan una distancia repulsiva para los obstáculos y una distancia atractiva para la meta, los vectores se asocian a comandos de velocidad que guiarán a la plataforma móvil a la meta deseada. El método ha sido usado en ^[17] de manera exitosa por el Dr. Víctor Javier González Villela (VJGV).

La plataforma móvil del robot tiene una configuración de ruedas diferencial o unicycle, que consta de un par de ruedas sobre un mismo eje que son controladas de manera independiente, adicionalmente tiene una rueda loca para darle estabilidad a la estructura del robot. Este tipo de estructura tiene restricciones cinemáticas no holonómicas, que impiden a la estructura desplazarse en ciertas direcciones sin antes haber rotado. Por lo anterior no es posible obtener su modelo de posición integrando sus ecuaciones cinemáticas.

El problema de llegar a la meta se reduce a encontrar la dirección y la velocidad en la que la plataforma móvil podrá llegar al punto deseado. Sea el punto inicial $P_i=(x_i,y_i)$ y el punto final $P_f=(x_f,y_f)$. La distancia entre ambos puntos queda definida por:

$$d_g = \sqrt{(x_f - x_i)^2 + (y_f - y_i)^2}$$

Ecuación 39. Distancia de la plataforma a la meta

El ángulo a la meta será determinado por la siguiente formula:

$$\theta_g = \tan^{-1} \left(\frac{y_f - y_i}{x_f - x_i} \right)$$

Ecuación 40. Angulo a la meta de la plataforma

El ángulo de error será determinado por la siguiente formula:

$$\theta_e = \theta_g - \theta_i$$

Ecuación 41. Angulo de error de la plataforma

La velocidad será determinada de acuerdo a las siguientes dos reglas, dependiendo sea el caso:

$$v = \begin{cases} v_{max} & \text{si } |d_g| > k_r \\ \frac{v_{max}}{k_r} d_g & \text{si } |d_g| \leq k_r \end{cases}$$

Ecuación 42. Velocidad de la plataforma. Ver cita 20

La velocidad máxima v_{max} está limitada por la velocidad máxima de la plataforma móvil. K_r es el radio del área de docking.

La velocidad angular máxima de la plataforma móvil será cuando el ángulo de error $[\theta_e]$ sea $\pm 90^\circ$, y la mínima cuando el ángulo de error sea 0° . La forma de determinar la velocidad angular es por la siguiente expresión, donde ω_{max} es la máxima velocidad angular de la plataforma móvil:

$$\omega = \omega_{max} * \sin(\theta_e)$$

Ecuación 43. Velocidad angular máxima de la plataforma

La velocidad del robot $[v]$ y la velocidad angular de la plataforma móvil $[\omega]$ permitirán orientar al robot a la posición final deseada. A una máxima velocidad lineal si el robot se encuentra fuera del área de docking y con delicadeza cuando se encuentre dentro del área de docking.

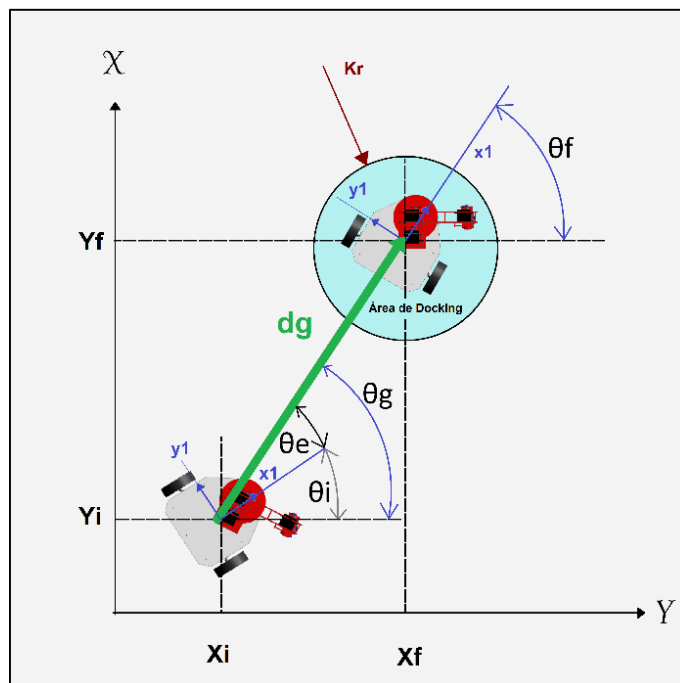


Ilustración 27 Definición del problema de llegar a la meta.

6.2. Evasión de obstáculos por medio de campos potenciales

Como se mencionó anteriormente el método de campos potenciales [U_0] toma en cuenta a los obstáculos como vectores repulsivos de distancia. Y se usa la mínima distancia del obstáculo [O], de esta forma la función queda:

$$U_o = \begin{cases} \frac{1}{2} \eta \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right)^2 & \text{sí } \rho \leq \rho_0 \\ 0 & \text{sí } \rho > \rho_0 \end{cases}$$

Ecuación 44. Campos potenciales Ver cita 20

Donde [η] es una ganancia constante, [ρ_0] representa el límite de la influencia del campo potencial y [ρ] es la mínima distancia del robot al obstáculo. La selección de [η] y [ρ_0] depende del robot móvil. Si los objetos son circulares con radio [ρ_r], entonces la distancia [ρ] es la distancia al centro del objeto [ρ_c] menos [ρ_r]. Y la ecuación queda como:

$$U_o = \begin{cases} \frac{1}{2} \eta \left(\frac{1}{\rho_c - \rho_r} - \frac{1}{\rho_0} \right)^2 & \text{sí } \rho_c \leq \rho_0 + \rho_r \\ 0 & \text{sí } \rho_c > \rho_0 + \rho_r \end{cases}$$

Ecuación 45. Campos potenciales 2. Ver cita 20

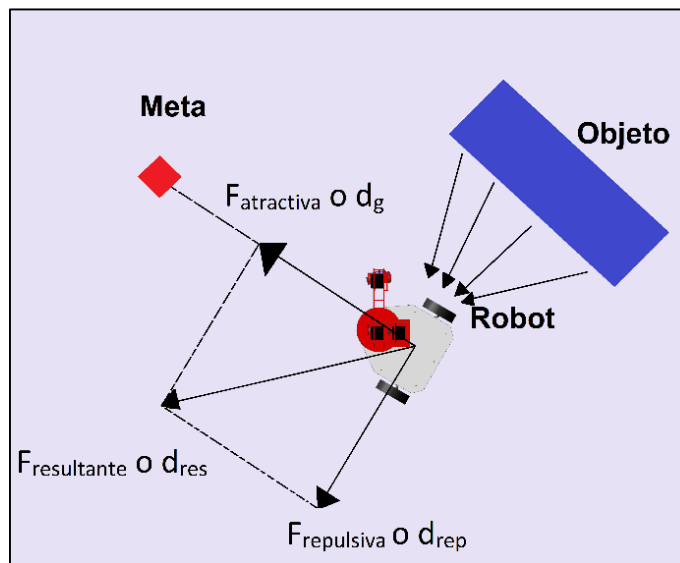


Ilustración 28. Método de campos potenciales con obstáculo

6.3. Coordinación de movimientos del robot híbrido

Por campos potenciales la velocidad del manipulador móvil dependerá del área donde se encuentre, por ejemplo si esta fuera del área de docking el manipulador se acercará a la meta a su máxima velocidad y en el caso del área de docking el manipulador se acerca a la meta reduciendo su velocidad cada vez más.

El manipulador móvil es una combinación de dos tipos de robots, un manipulador antropomórfico y una plataforma móvil. En el presente trabajo se buscará un movimiento más natural que asemeje al comportamiento humano, haciendo una analogía, por ejemplo te encuentras de pie y necesitas tomar tu teléfono móvil que está a un par de metros de distancia, comienzas a caminar y ya que estas acercándote extiendes tu brazo para tomar el celular, lo anterior sin dejar de caminar hasta que alcanzas el móvil. La tarea que has desarrollado inconscientemente es una coordinación de movimiento entre tu desplazamiento y tu brazo.

El manipulador móvil tendrá un movimiento coordinado entre la plataforma móvil y el brazo robótico en el presente trabajo se realizará una coordinación de movimientos que dependa solo de la posición de la plataforma móvil en el espacio, de esta forma el manipulador se moverá a la vez que se esté acercando a la meta. Por tal motivo el movimiento del manipulador se parametrizó, de esta forma no importa la distancia de la plataforma a la meta, el brazo se moverá al tiempo que lo hace la plataforma móvil. Con este tipo de coordinación se busca un movimiento que se asemeje al comportamiento humano.

Sea el punto inicial del manipulador (x_i, y_i) , sea el punto (x_{f1}, y_{f1}) para recoger un objeto, sea el punto (x_{f2}, y_{f2}) para dejar el objeto y sean las coordenadas del manipulador móvil (x_{mm}, y_{mm}) .

La distancia total se determina por:

$$d_{total-1} = \sqrt{(x_{f1} - x_i)^2 + (y_{f1} - y_i)^2}$$

Ecuación 46. Distancia total punto de inicio punto 1

$$d_{total-2} = \sqrt{(x_{f2} - x_{f1})^2 + (y_{f2} - y_{f1})^2}$$

Ecuación 47. Distancia del punto final 1 al punto 2

La distancia del manipulador a la meta se determina por:

$$d_{recoger} = \sqrt{(x_{f1} - x_{mm})^2 + (y_{f1} - y_{mm})^2}$$

Ecuación 48. Distancia de la plataforma al punto 1

$$d_{dejar} = \sqrt{(x_{f2} - x_{mm})^2 + (y_{f2} - y_{mm})^2}$$

Ecuación 49. Distancia de la plataforma al punto 2

Nota: el cálculo d_{dejar} solo se realiza después de haber tomado el objeto a manipular. Y el cálculo de $d_{recojer}$ solo se realiza al ir por el objeto a manipular.

Los parámetros de coordinación se definen a continuación y el máximo valor que pueden alcanzar es de la unidad:

$$mov_1 = \frac{d_{recojer}}{d_{total-1}}$$

Ecuación 50. Parámetro de coordinación 1

$$mov_2 = \frac{d_{dejar}}{d_{total-2}}$$

Ecuación 51. Parámetro de coordinación 2

El parámetro de coordinación para llegar y tomar un objeto manipulable

$$\alpha p = 128 - mov1(128)$$

Ecuación 52. Parámetro de coordinación 1 que toma en cuenta el movimiento del brazo robótico. Parámetro αp , también mencionado como parámetro tomar.

Finalmente el parámetro de coordinación para dejar el objeto manipulable

$$\beta p = 256 - mov2(128)$$

Ecuación 53. Parámetro de coordinación 2 que toma en cuenta el movimiento del brazo robótico. Parámetro βp , también mencionado como parámetro dejar.

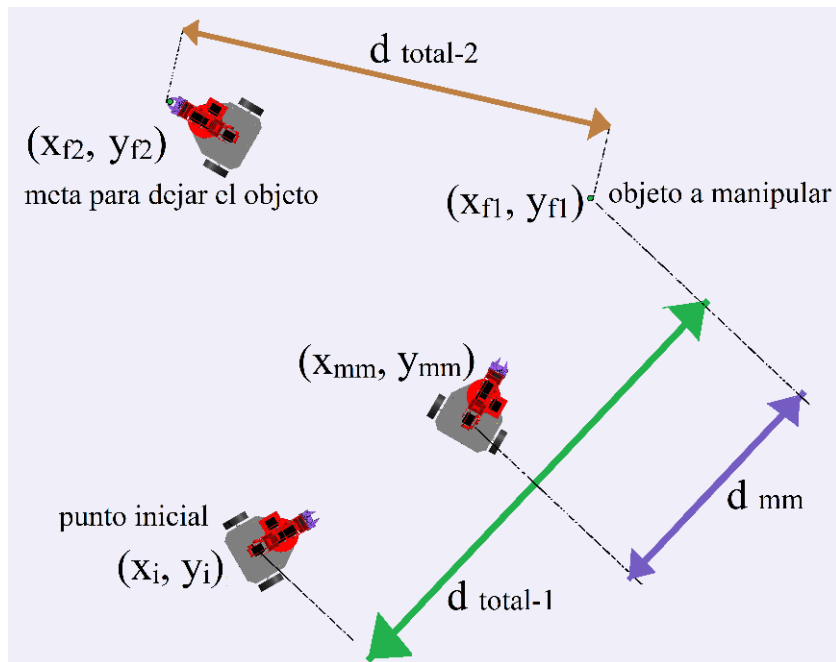


Ilustración 29. Diagrama esquemático donde se muestra la ubicación de los puntos y distancias

Los parámetros mov_1 y mov_2 deben tener valores entre uno y cero, y sus valores deben ser decrecientes en el orden mencionado hasta llegar a cero. Es necesario resaltar que se podrían dar parámetros mov_1 y mov_2 crecientes que pueden afectar la movilidad del brazo en un periodo de tiempo muy corto, lo anterior para el caso del manipulador. Este problema se debe al tipo de plataforma móvil que se tiene. Ya que muchas veces al moverse se aleja del punto al que se desea llegar. Lo anterior solo es una consideración que se debe de tener presente, y que justifica un movimiento del brazo que intenta retraerse en un periodo de tiempo muy corto, sobre todo en casos en donde la plataforma móvil esta de espaldas a su meta.

Por último, la desventaja de este tipo de coordinación son los casos límite donde la distancia tiende a ser cero, en este trabajo no se considera esta situación debido a que tendría que hacerse otro tipo de coordinación donde la plataforma móvil no se mueva y solo lo haga el manipulador. La forma de abordar este tipo de casos con la coordinación de movimientos que se ha definido sería establecer un punto pivote que esté alejado del punto meta para poder seguir trabajando solo en función de la distancia a la meta.

El manipulador móvil escogido tiene un brazo serial de tipo antropomórfico y como efector final un gripper. El brazo serial tiene un rango de movimientos que pueden compensar el ángulo con el que llega la plataforma móvil a la meta. Se utiliza algo parecido al reflejo oculocefálico conocido también como reflejo de ojos de muñeca, un reflejo que tenemos los seres humanos que nos permiten ver un punto fijo al girar la cabeza. En el caso de la

plataforma móvil se estableció un ángulo que coincida con el eje X, puede ser $0[^\circ]$ o $180[^\circ]$, de esta forma el manipulador llegará a la meta con el ángulo deseado, listo para tomar el objeto, este movimiento está parametrizado de tal forma que sea dependiente de la distancia a la que se encuentra.

Matemáticamente:

$$\theta_{base} = \theta_{manipulador} \left(\frac{128 - \alpha p}{128} \right) + \theta_{deseado} \left(\frac{\alpha p}{128} \right)$$

Ecuación 54. Compensación del error debido a la no homonimia de la plataforma

Recordar que el parámetro αp se ha calculado anteriormente, de esta forma se tiene una transición suave de un ángulo inicial a un ángulo deseado que depende de la distancia de la plataforma móvil al objetivo.

La coordinación de movimientos antes descrita implica que el sistema no tiene que hacer el cálculo de todos los parámetros que fueran necesarios para hacer una tarea de esta naturaleza. De esta forma se busca un movimiento más fluido e intuitivo, lo cual implica que el manipulador se mueve de acuerdo a la distancia que hay entre el objetivo de partida y el objetivo final. De esta forma solo nos interesan las distancias. De esta forma se ahorra tiempo de cálculo. Y es un movimiento que el robot realiza de manera intuitiva pareciendo estar mecanizada.

6.4. Docking y predocking otra alternativa a la coordinación de movimientos.

En el trabajo anterior se establecían dos puntos clave para manipular un objeto conocido como docking y predocking. El punto donde el manipulador móvil llegaba se denominaba predocking [pD], y entre el objeto y el manipulador existía una distancia [dD], para el cálculo del predocking era necesario tener la orientación del objeto a manipular, y la ubicación del mismo. Se ejemplifica la tarea en el siguiente diagrama.

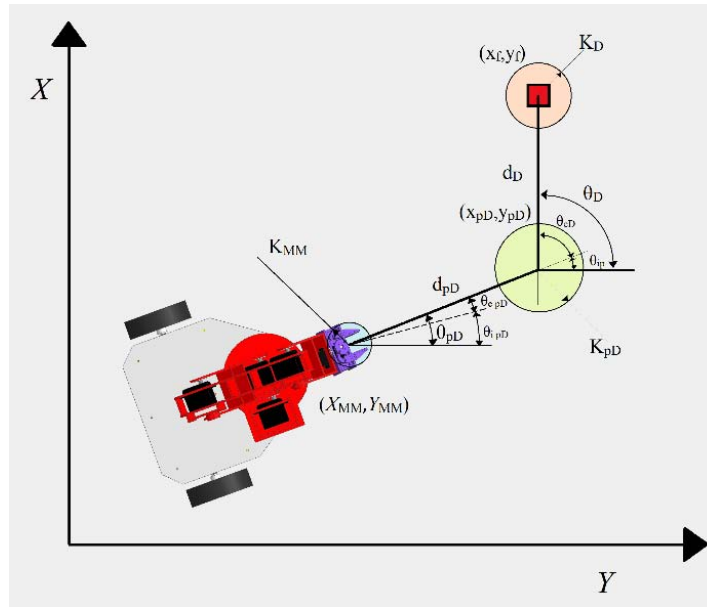


Ilustración 30. Docking y predocking

El utilizar este tipo de trayectoria es muy efectiva pero el problema es que el movimiento del manipulador se es independiente al de la plataforma, primero se mueve la plataforma y después el manipulador.

6.4.1 Base de datos

En el caso de los seres humanos nos es muy sencillo reconocer algo en base a experiencias previas, podemos incluso saber de qué material están hechas ciertas cosas, lo anterior con solo echar un vistazo alrededor y sin gran esfuerzo mental.

En el espacio inteligente se plantea que el robot móvil haga algo similar a lo que nosotros hacemos. Por ejemplo, que sea capaz de saber y conocer que elementos puede manipular y cuales presentan un obstáculo fijo. Se quiere lograr lo anterior con poca cantidad de procesamiento y en tiempo real. Por lo que es necesario cambiar de paradigma y se propone que en vez de que se utilice una gran cantidad de recursos computacionales en el reconocimiento de objetos tridimensionales, se simplifique por medio de un sistema de visión diseñado para robots. Lo anterior y en una primera etapa serán Fiduciales que tendrán asignados en una base de datos las cualidades y características de cada objeto detectado. Esto será una primera iteración que se puede explotar a una mayor escala y posteriormente se pueden remplazar los fiduciales por chips u otro tipo de elementos diseñados para una fácil detección en el caso de los robots.

En el presente trabajo se ejemplifican las diferentes metas a modo de fiduciales reconocibles para el sistema que simulan ser cualquier cosa en el espacio inteligente. Es por esto que se propone una base de datos que contenga las características de cada objeto que se encuentre dentro del espacio inteligente. De esta forma el espacio inteligente será capaz de saber que hay dentro de sí, conocerá cada objeto, y sabrá cualidades de cada objeto, por ejemplo si lo puede manipular o no.

Elementos tentativos dentro de la base de datos:

- Ubicación espacial (x, y, z)
- Orientación e inclinación del objeto.
- Propiedades físicas del objeto como masa, volumen, forma geométrica.
- Materiales que lo componen.
- Datos extras que pueden ser útiles a la hora de manipular como centroide del objeto, fuerza necesaria para sujetar, etc.

Actualmente existen algunos dispositivos electrónicos como la IMU (unidad de medición inercial) que pueden proporcionar información de este tipo, pero no se implementaron porque existen otras formas de conocer este tipo de información.

6.5. Programación en el microcontrolador Arduino

Se decidió la utilización del microcontrolador Arduino R3 debido a la facilidad de programación, es una plataforma de software libre que tiene módulos de fácil adición, en este trabajo se adiciono un módulo wifi-shield que permitió la comunicación mediante una red wifi, que mejora la velocidad de respuesta en comparación con otras tecnologías usadas con anterioridad por alumnos del grupo MRG (Xbee y Bluetooth).

La placa de Arduino de fácil programación mediante un puerto USB, solamente se necesita el software. El software tiene un lenguaje de programación de alto nivel que facilita considerablemente su programación. También encuentras tutoriales que facilitan considerablemente el uso de nuevas tecnologías como el wifi-shield.

Una de las mayores ventajas de Arduino es que posee librerías para muchos dispositivos que pueden montarse en la placa, por ejemplo en este caso se usaron las librerías de wifi, servos y i²C.

Se utilizó una placa de Arduino R3, que tiene un microcontrolador Atmega328, los puertos fueron los necesarios para controlar los 7 servomotores, la tarjeta MD25 así como el wifi-shield. A lo largo del desarrollo del presente trabajo se utilizó considerablemente la comunicación serial para comprobar que las conexiones con los periféricos fuera exitosa.

El microcontrolador Arduino solo se usa para la recepción de datos y el control de los periféricos en el manipulador móvil. En la página de anexos se puede observar el código de programación.

Experimentación

7.1. Objetivos experimentales.

La parte experimental está enfocada a buscar una coexistencia del robot con los seres humanos, dentro del espacio inteligente. El robot solo será un actuador del espacio inteligente que coexista con el ser humano.

La tarea diseñada busca una colaboración por parte del robot en alguna tarea humana, de esta forma la idea de un ambiente inteligente no solo se limita a buscar un confort para los seres humanos que la habitan. También se plantea una colaboración por parte del ente robótico en una tarea como el transporte de material.

7.2. Diseño del experimento.

Para demostrar cooperación la tarea será el transporte de material de un lugar a otro, el manipulador móvil tomará un objeto manipulable que estará alejado del lugar donde se encuentre. Irá por el objeto y lo tomará con la coordinación de movimientos descrita en el capítulo anterior.

7.3. Resultados experimentales

Se tuvieron problemas en la parte mecánica del brazo robótico por lo que fue necesario cambiar la mayoría de los servos por unos con mayor torque, lo anterior debido a que el torque de los antiguos servos no era el necesario para operar a bajas velocidades, y se tenían errores muy grandes en la posición angular.

La emisión y recepción de datos tiene un retraso documentado de 1.7 [s], más lo que tardan en reaccionar los actuadores como los servos. Es un retraso bastante considerable que es necesario reducir al mínimo.

La configuración de la plataforma móvil coordinada con campos potenciales impide controlar la posición angular de la plataforma al llegar a la meta, un error que se tuvo que compensar en la base de la plataforma del robot. Por lo que se sugieren para nuevas etapas experimentales jugar con un controlador que pueda controlar el ángulo con el que llega la plataforma móvil.

La coordinación de movimiento programada permitió que la plataforma móvil y el brazo robótico se movieran en conjunto logrando también que el brazo robótico se mueva de manera suave.

La plataforma móvil se movió de manera exitosa con campos potenciales, y resultó ser una herramienta perfecta para parametrizar el movimiento del brazo robótico.

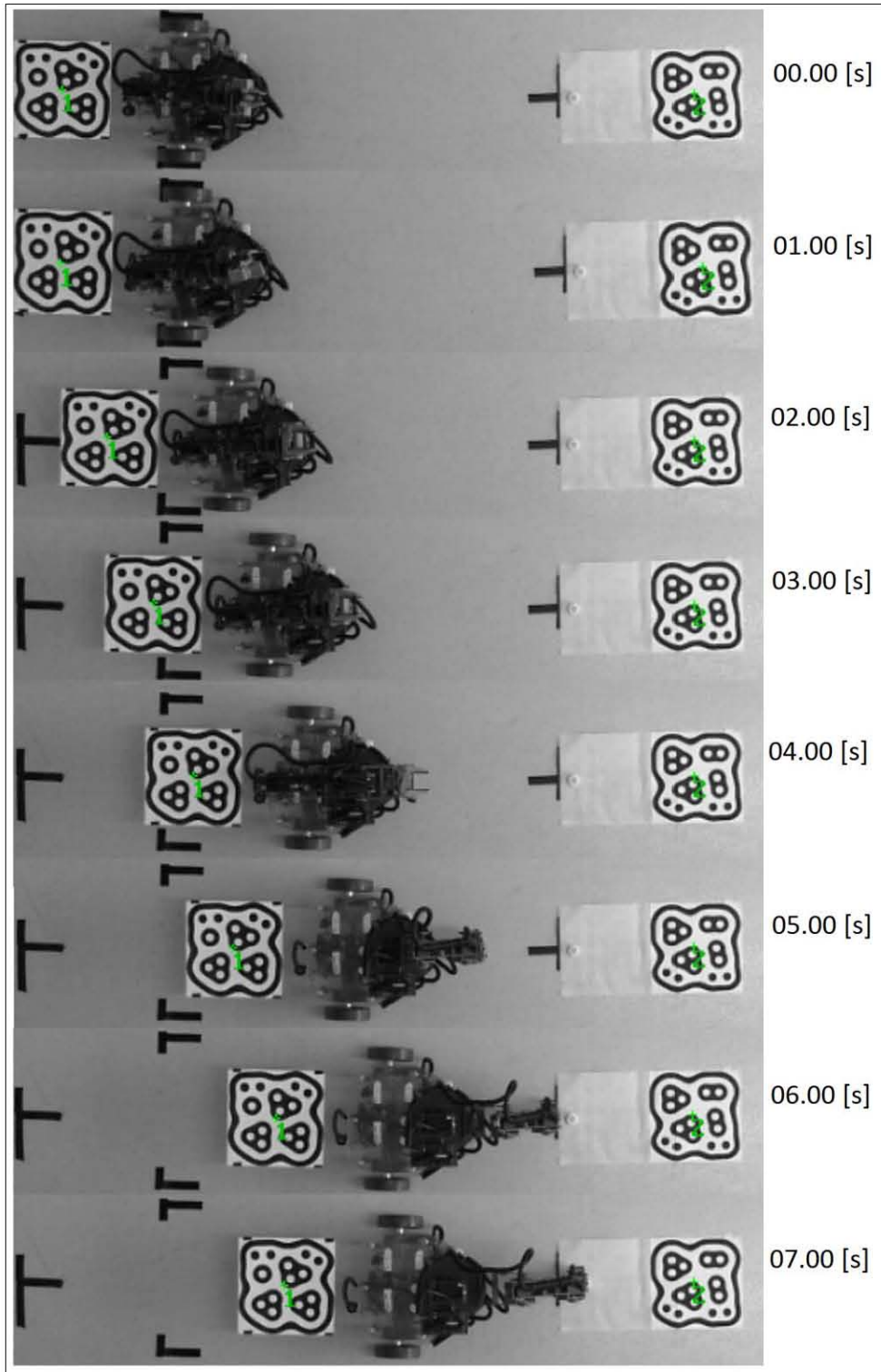


Ilustración 31. Secuencia fotográfica del experimento, el tiempo de ejecución solo es ejemplificativo.

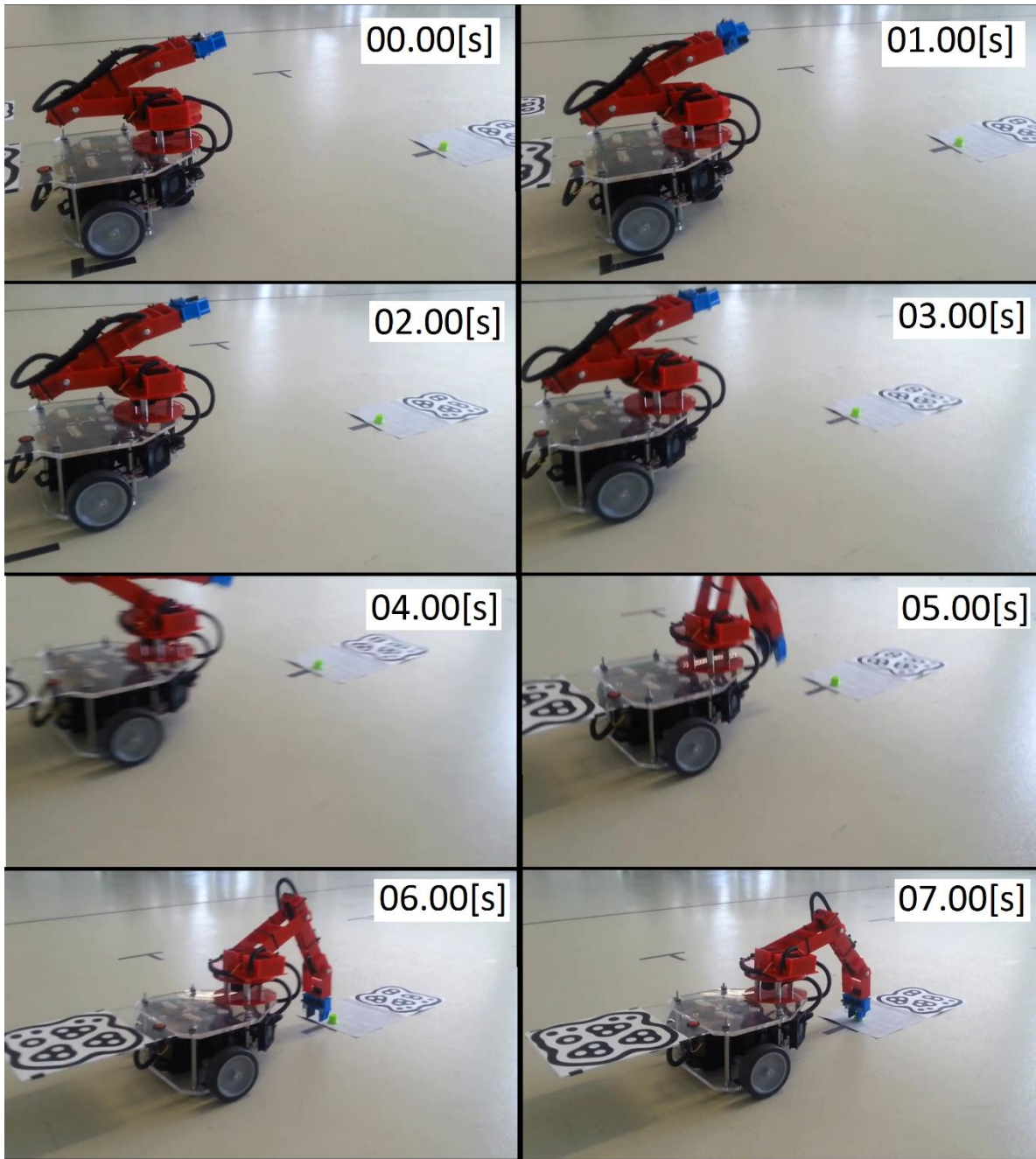


Ilustración 32. Ilustración 30. Secuencia fotográfica del experimento, el tiempo de ejecución solo es ejemplificativo. solo son perspectivas diferentes

7.3.1 Gasto computacional

La realización de estas pruebas fue hecha por separado, y se realizó para medir y corroborar que el gasto computacional es menor en las ecuaciones obtenidas.

Fue necesario hacer esta comparación con el objetivo de medir el gasto computacional y poder así, realizar la comparativa. En ambos casos se utilizó el mismo programa (*Mathematica 9.0*), se eliminó el mayor número de tareas que estaba ejecutando la computadora.

En la primera prueba se realiza la cinemática inversa del manipulador a lo largo de la trayectoria de ida, como se encuentra descrita en el cuarto capítulo. Y en la segunda simulación se mueve el brazo robótico a lo largo de la misma trayectoria, pero con las ecuaciones obtenidas en el mismo capítulo.

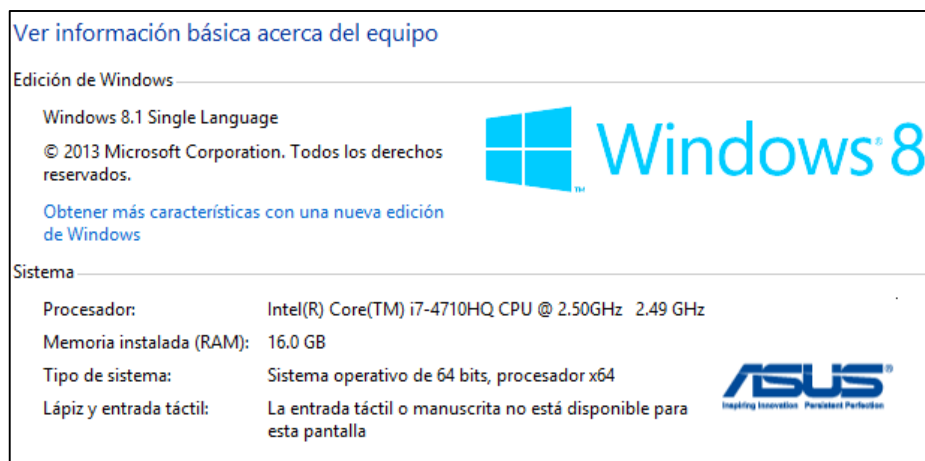


Ilustración 33. Características del equipo de cómputo donde se realizó la prueba, la frecuencia de trabajo tiene un control dinámico por lo que puede cambiar.

Nombre	Estado	7% CPU	19% Memoria	0% Disco	0% Red
Camtasia Recorder (32 bits)		4.2%	215.3 MB	0.3 MB/s	0 Mbps
Aislamiento de gráficos de dispositivo de audio de Windows		2.6%	13.7 MB	0 MB/s	0 Mbps
Administrador de ventanas del escritorio		0.4%	22.0 MB	0 MB/s	0 Mbps
Host de servicio: Servicio local (red restringida) (5)		0.3%	13.4 MB	0.1 MB/s	0 Mbps
System		0.2%	0.3 MB	0.1 MB/s	0 Mbps
Interrupciones del sistema		0.1%	0 MB	0 MB/s	0 Mbps
Host de servicio: Sistema local (red restringida) (11)		0%	124.9 MB	0 MB/s	0 Mbps
Administrador de tareas		0%	17.7 MB	0 MB/s	0 Mbps
Proceso en tiempo de ejecución del cliente-servidor		0%	1.8 MB	0 MB/s	0 Mbps
Monitor de rendimiento y recursos		0%	49.6 MB	0 MB/s	0 Mbps
Explorador de Windows		0%	60.6 MB	0 MB/s	0 Mbps
Wolfram Mathematica 9		0%	195.7 MB	0 MB/s	0 Mbps
WMI Provider Host		0%	1.1 MB	0 MB/s	0 Mbps
XML (32 bits)		0%	2.2 MB	0 MB/s	0 Mbps
Antimalware Service Executable		0%	68.2 MB	0 MB/s	0 Mbps
ROG GameFirst III Service (32 bits)		0%	5.4 MB	0 MB/s	0 Mbps

Ilustración 34. Aplicaciones que estaban consumiendo recursos computacionales al hacer las pruebas.

Primera prueba, se realizó el cálculo de la cinemática inversa para calcular cada una de las 128 posiciones a lo largo del perfil de trayectoria de ida.

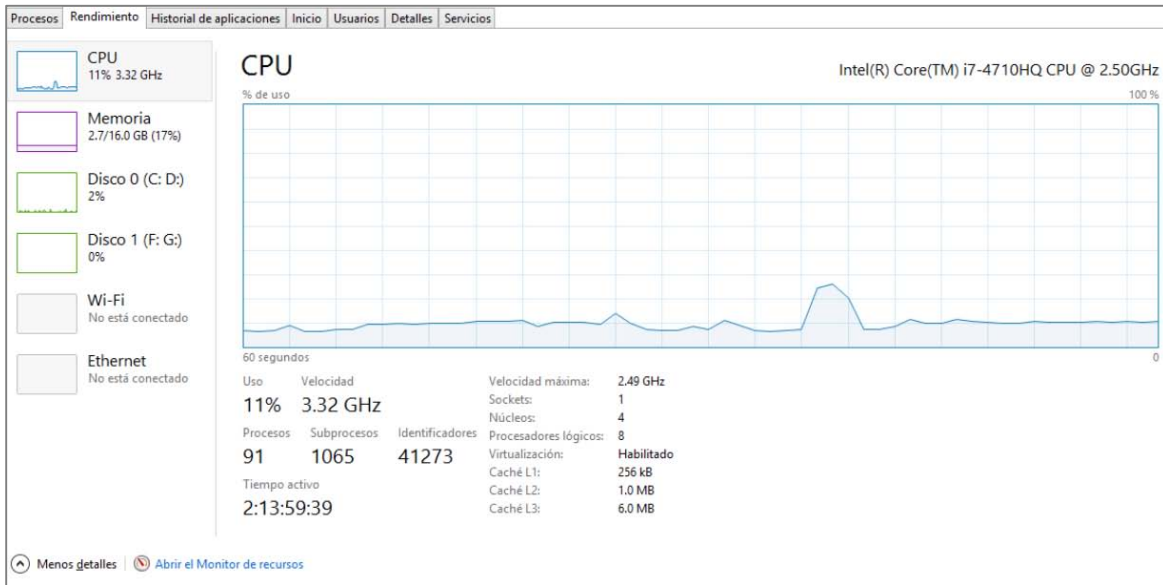


Ilustración 35. Gasto computacional cinemática inversa a lo largo del perfil de trayectoria.

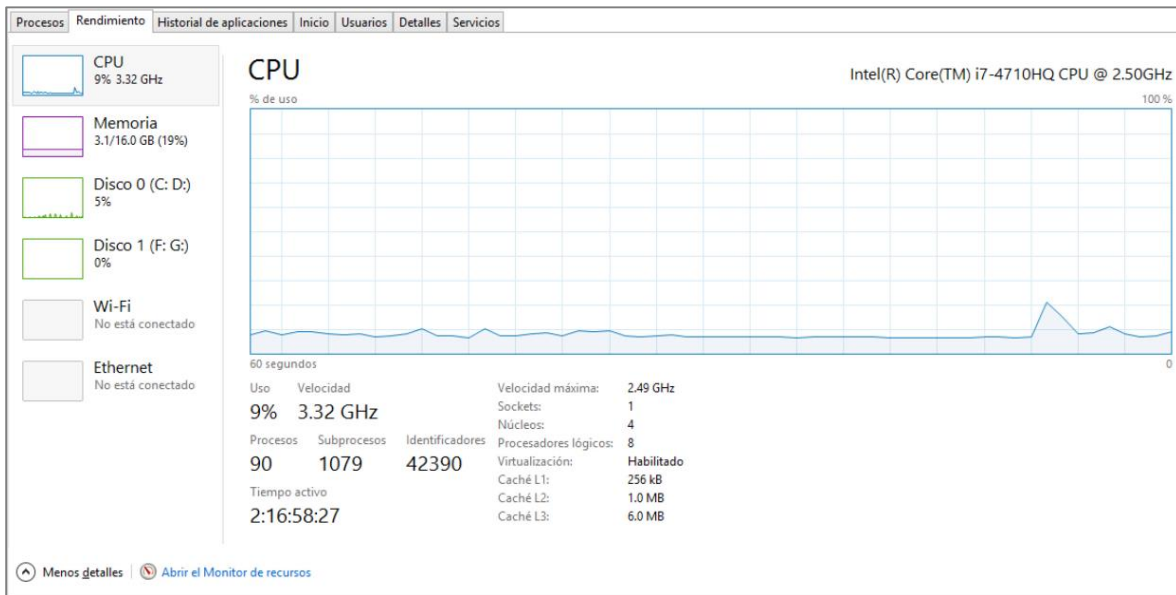


Ilustración 36. Gasto computacional del movimiento del brazo con las ecuaciones de movimiento obtenidas.

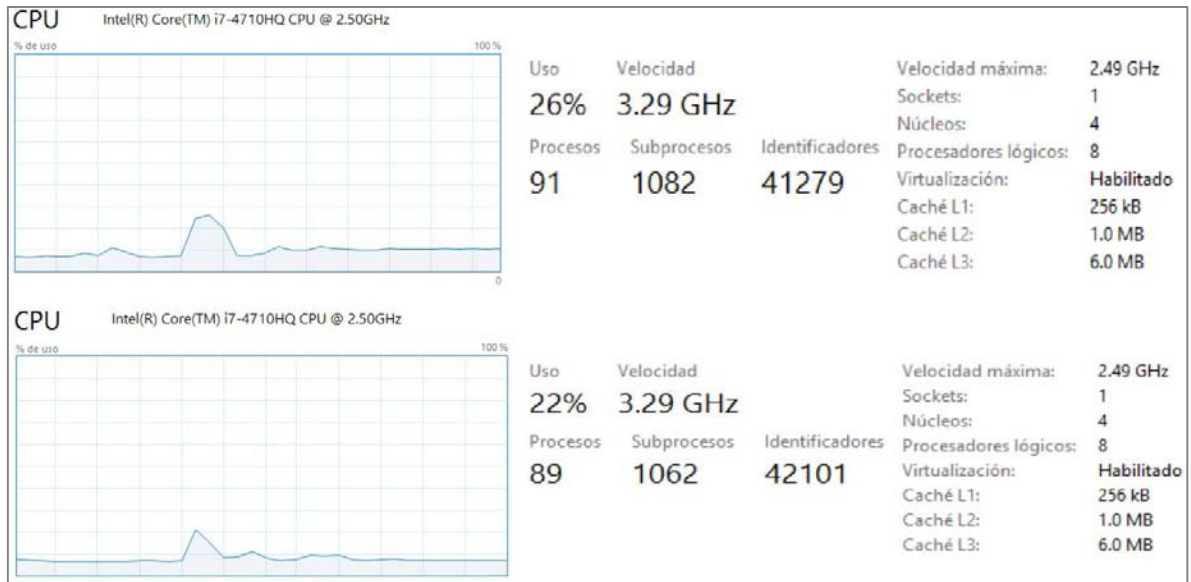


Ilustración 37. Comparación del gasto computacional, los datos mostrados corresponden al momento de mayor uso del CPU, que fue donde se realizó la simulación.

Es necesario señalar que las pruebas realizadas también incluyen una simulación del movimiento del brazo, en ambos casos se procuró que la programación realizada fuera lo más equitativa posible, quitando instrucciones innecesarias que pudieran afectar la medición. Se desactivo el wifi del equipo de cómputo y se cerró el mayor número de aplicaciones.

Observando la comparativa de las gráficas observamos un pico más pequeño debido al gasto de recursos computacionales en la segunda simulación. En amplitud difieren por muy poco apenas un 4% pero el área bajo la curva del primer pico es mayor.

Conclusiones y trabajo a futuro.

8.1. Discusión

En el presente trabajo se utiliza la teoría de campos potenciales para trasladar la plataforma móvil de un lugar a otro. En el caso de la plataforma móvil llega a la meta de manera adecuada pero no se puede controlar el ángulo en el que llega a la meta de la plataforma, lo que implicó una compensación del ángulo por medio del manipulador.

La parte de parametrización del movimiento del manipulador móvil facilita mucho la coordinación de movimientos, y ahorra muchos cálculos computacionales.

La coordinación de movimientos presente en este trabajo ha permitido mover de manera adecuada a la plataforma móvil y al manipulador antropomórfico para que trabajen en conjunto. Hubo problemas en la parte física del manipulador que tuvieron que ser considerados en la parte de coordinación de movimientos.

El sistema de visión es una herramienta que nos permitió el desarrollo del presente trabajo, y se insiste en la idea de que los espacios inteligentes pueden tener otro tipo de formas para detectar lo que se encuentra dentro de ellos.

8.2. Conclusiones

El espacio inteligente que se desarrolló en el presente trabajo es un prototipo funcional a escala que sienta las bases de una coexistencia de un robot con un ser humano en un mismo espacio de trabajo.

En espacios inteligentes puede haber una coexistencia entre robots y humanos ya sea o no en cooperación y es necesario comenzar a tener una base de datos de las características específicas de cada objeto, que permitan al espacio inteligente conocer cada vez más cosas del entorno.

En el presente trabajo se aplicó con éxito una nueva forma de coordinar movimientos entre una plataforma móvil y un brazo robótico. La coordinación planteada simplifica considerablemente los cálculos y se puede hablar de una coordinación entre ambos robots (la plataforma móvil y el manipulador), pero hay que tener en cuenta que hay casos donde este tipo de coordinación no aplica o no es recomendable, por ejemplo cuando la distancia es muy poca o cuando la distancia es muy grande.

En el experimento que se diseñó, se corroboró que la coordinación de movimiento funciona exitosamente, la tarea que se planteó fue tomar un objeto con el propósito de transportarlo sin importar en que parte del espacio de trabajo se encuentre. El espacio inteligente es capaz de ser consiente ¿en qué parte del espacio se encuentra? e ir por él. Y coexiste con otro objeto, sin problema alguno.

Los actuadores influyeron mucho en el experimento, y fue necesario el remplazo de la mayoría de los servos, para realizar adecuadamente el movimiento del brazo con un perfil de trayectoria.

La programación de los algoritmos de intuición artificial se realizó con éxito, y la coordinación de movimientos se reduce a los comandos de velocidad que se generan de campos potenciales. Y se genera un sistema semiautónomo capaz de generar respuestas por sí mismo.

La intuición artificial resulta una herramienta que permite coordinar la totalidad de movimientos del manipulador móvil, pero es necesario definir una distancia de seguridad para evitar obstáculos. Este es un pretexto más para utilizar campos potenciales en este tipo de coordinación intuitiva.

Los algoritmos de intuición permiten al manipulador móvil desarrollar la tarea de transporte de material en un menor tiempo, y eliminan la posición de predocking.

La intuición artificial simula un movimiento natural sobre aprendido que puede ser repetido de forma automática y repetidas veces, es necesario señalar que el algoritmo programado integra un algoritmo de aprendizaje, es decir no tiene la capacidad de aprender. Pero la intuición artificial programada si se dio como consecuencia del aprendizaje y en base a la experiencia adquirida en este caso del programador. Tiene la ventaja de generar respuestas espontaneas, y con bajos recursos computacionales.

8.3. Trabajo a futuro

El manipulador móvil que se usó durante el presente trabajo presenta restricciones mecánicas debido al par en los actuadores que necesitan ser consideradas a la hora de mover el manipulador. Se observó que los servomotores no siempre llegaban a la posición deseada, sobre todo a bajas velocidades, por lo que se propone un controlador de servos con retroalimentación extra a la retroalimentación del controlador interno del servo.

La plataforma móvil utilizada tiene una configuración de ruedas diferencial, que limitan el tipo de movimientos que puede realizar la plataforma, se propone una experimentación con plataformas holonómicas para que el manipulador móvil pueda ser más versátil, de esta forma la plataforma se podrá mover con mayor libertad en el espacio de trabajo.

Para poder trabajar en terrenos más accidentados se propone cambiar la plataforma móvil diferencial por otro tipo de plataforma como orugas o una plataforma en configuración de hexápodo.

Un trabajo que queda pendiente y se considera posibles es la expansión del espacio inteligente a un ambiente inteligente, que implicaría el conjunto de diversos espacios inteligentes.

El movimiento del manipulador está hecho con una memoria de cálculos en base a perfiles geométricos establecidos, propongo un sistema más flexible que admita otros perfiles geométricos. Lo anterior debido a que existen casos límites por ejemplo cuando la distancia es muy corta, en este caso el movimiento del manipulador tendría que desarrollarse sin la necesidad de mover la plataforma móvil.

Un trabajo a futuro es considerar a la dinámica del manipulador y la dinámica de la plataforma en la coordinación de movimientos.

Se considera adecuado agregar sensores que ayuden al sistema de visión sobre todo a la hora de manipular objetos. Ya que es un reto que el manipulador realmente tome el objeto a manipular.

La coordinación de movimientos que posee el manipulador móvil no toma en cuenta la dirección de la distancia, solo toma el valor de la magnitud lo que puede arrojar parámetros (dejar y mover) crecientes. En la siguiente iteración de este trabajo se puede tomar en consideración la orientación de la dirección y del móvil para evitar parámetros crecientes.

Para el caso particular de una plataforma móvil como la que se utilizó en el presente trabajo, la programación con campos potenciales permite llegar a la meta y evadir adecuadamente obstáculos, pero campos potenciales no permite controlar el ángulo de llegada de la plataforma móvil esto también se debe a las restricciones cinemáticas de la configuración elegida. Existen otras formas de controlar el desplazamiento de la plataforma para poder controlar el último ángulo de la plataforma, esto exige hacer cambios entre más de dos controles distintos. Se propone un sistema de control de este tipo para poder controlar mejor la plataforma móvil.

Bibliografía

1. Juan Manuel Vargas Medina. (Septiembre de 2004). Tipos de inteligencia. Consultado el día 19 de marzo de 2014 en: <http://www.ingenieria.unam.mx/~guiaindustrial/entorno/menu6.htm>
2. J. D. Greenwood; “Historia de la psicología. Un enfoque conceptual”; Mc Graw Hill, 1a. ed.; México, 2011.
3. A.M. Sandoval; “Experimentación en tiempo real”; Facultad de ingeniería, UNAM (2011), México.
4. R. Descartes; “Tratado de las pasiones del alma”; In J. Cottingham, R. Stoothhoff, & D. Murdoch (Trans.), The philosophical writings of Descartes (vol. 1). Cambridge University Press (original work published 1649), (1985).
5. J. Negrete, E. González; “De la Filosofía a la inteligencia artificial”; Ed. Limusa (1992), México.
6. M. Manubhai, K. S. Huang, I. Mikic; “Dynamic Context Capture and Distributed Video Arrays for Intelligent Spaces”; IEEE transactions on systems, man, and cybernetics part a: systems and humans, vol. 35, no. 1, January 2005.
7. Stuart J. Russell, Peter Norving; “Artificial Intelligence. A Modern Approach”; Ed. Prentice Hall (2010), United States of America.
8. David G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints”; Computer Science Department, University of British Columbia; Vancouver, B.C., Canada (January 5th, 2004)
9. Kazuyuki Morioka, Joo-Ho Lee and Hideki Hashimoto; “Intelligent Space for human Centered Robotics”; Meiji University, Ritsumeikan University, University of Tokyo (Japan 2008).
10. Pentland, A.; “Smart rooms”; Masachussets Institute of Technology (1996).
11. Coen, M. H.; “Design principles for intelligent environments”. Masachussets Institute of Technology (1996).
12. Shafer, S., Krumm, J., Brumitt, B., Meyers, B., Czerwinski, M., and Robbins; “The new easyliving project at microsoft research.” Consultado en <http://research.microsoft.com/apps/pubs/?id=68393> el día 18 de Agosto de 2014.
13. X. Alemán, P. Haya, G. Montoro. “Hacia un entorno inteligente basado en un interfaz en lenguaje natural”. Departamento de ingeniería informática, Universidad Autónoma de Madrid, España (2004)
14. A. Bobick, J. Davis, S. Intille; “The KidsRoom: An Example Application using a deep perceptual interface”; The MIT Media Laboratory.
15. Definiciones de Robot, consultado el 28 de Mayo en: http://www.linkmesh.com/robots/articulos/definiciones_de_los_robots.php

16. Yi Cao, Ke Lu, Xiujuan Li and Yi Zang; “Accurate Numerical Methods for Computing 2D and 3D Robot Workspace [Journal]” // International Journal of Advanced Robotic Systems : INTECH, August 2011.
17. González V. V. J., Parkin R., López P. M., Dorador G. J. M., Guadarrama L.J. “A wheeled mobile robot with obstacle avoidance capability”, Departamento de Ingeniería Mecatrónica, Facultad de Ingeniería UNAM, CU, México 2004.
18. Díaz Hernández Octavio, “Tesis: Intuición artificial aplicada a la teleoperación”, Departamento de Ingeniería Mecatrónica, Facultad de Ingeniería UNAM, CU, México 2014.
19. S.A. Alshahrani¹, H. Diken², A. A. N. Aljawi², “Optimum trajectory function for minimum energy requirements of a spherical Robot”. 1: Master student, 2: Associate Professor, production and mechanical system Design Department King Abdulaziz University, Jeddah, 21589, Saudi Arabia.
20. Escobedo Castillo P. M. “TESIS: Manipulador móvil: Estudio sobre la coordinación de movimientos de un manipulador serial acoplado”, Facultad de Ingeniería UNAM, CU, México 2012
21. Julio C. Correa, Rafael E. Vásquez, Juan Ramírez-Macías. “Manipuladores Robóticos una mano a la industria (Revista edición 29)” Consultado en: http://www.metalactual.com/revista/29/soldadura_roboticos.pdf el día 08/06/2015.

[Fig. 1.] Robot serial KUKA KR-6, obtenido de “Manipuladores Robóticos (Revista)”, Julio C. Correa, Rafael E. Vásquez, Juan Ramírez-Macías, Elkin A. Tabora. Consultado en: http://www.metalactual.com/revista/29/soldadura_roboticos.pdf el día 08/06/2015.

[Fig2.] coparoman (Mayo 2013); “Sistemas de coordenadas del Robot”, consultado el día 1 de septiembre de 2014 en: http://coparoman.blogspot.mx/2013_05_01_archive.html

[Fig3.] Mg. Samuel Oporto Díaz (Mayo 2013); “Introducción a la Robótica”, consultado el día 2 de septiembre de 2014 en: <http://slideplayer.es/slide/92174/>

[Fig4.]J. M. Selig, “Introductory Robotics”; Department of Electrical and Electronic Engineering, South Bank Polytechnic, Prentice Hall (1992).

[Fig. 5.] coparoman (Mayo 2013); “Sistemas de coordenadas del Robot”, consultado el día 1 de septiembre de 2014 en: http://coparoman.blogspot.mx/2013_05_01_archive.html

[Fig. 6.] ROBOTIC ELECTRIC GUITAR (Posted on March 4, 2013); “SCARA Arm”, consultado el día 2 de septiembre de 2014 en: <http://vindy500.wordpress.com/2013/03/04/scara-arm/>

[Fig. 7.] José Ramón López “Tipos de Robots”, consultado el día 2 de septiembre de 2014 en: <http://cerezo.pntic.mec.es/rlopez33/tecno/tercero/robot/tipos.html>

[Fig. 8.] Engineer JAU “El espacio de trabajo en Robótica, parte 1”, consultado el día 2 de septiembre de 2014 en: <http://jauelingeniero.wordpress.com/2013/07/07/bases-sobre-espacios-de-trabajo-en-robotica-parte-1/>

[Fig. 8.] P. M. Escobedo Castillo “Manipulador Móvil”.

[Fig. 10.] Manipulador móvil tipo humanoide PR2. Consultado el día 19 de septiembre de 2014 en: <https://www.willowgarage.com/pages/pr2/overview>

Anexos

9.1. Tarjeta MD25 y motores EMG30

Es una tarjeta para controlar un par de motores de corriente directa modelo EMG30 por medio del protocolo de comunicación I²C. Puede proporcionar hasta tres amperes por cada motor, dispone de capacitores necesarios para la eliminación de ruidos producido por los motores. Solo requiere ser alimentada por una fuente externa que en el caso de este trabajo es una pila de 12 volts recargable. Dispone de un controlador de velocidad con retroalimentación para verificar que la velocidad deseada de los motores sea la obtenida. Por default la tarjeta tiene una dirección 0x58 en el protocolo de i²C. Cuenta con 17 registros internos para escribir y leer la información deseada.

Register	Name	Read/Write	Description
0	Speed1	R/W	Motor1 speed (mode 0,1) or speed (mode 2,3)
1	Speed2/Turn	R/W	Motor2 speed (mode 0,1) or turn (mode 2,3)
2	Enc1a	Read only	Encoder 1 position, 1st byte (highest), capture count when read
3	Enc1b	Read only	Encoder 1 position, 2nd byte
4	Enc1c	Read only	Encoder 1 position, 3rd byte
5	Enc1d	Read only	Encoder 1 position, 4th (lowest byte)
6	Enc2a	Read only	Encoder 2 position, 1st byte (highest), capture count when read
7	Enc2b	Read only	Encoder 2 position, 2nd byte
8	Enc2c	Read only	Encoder 2 position, 3rd byte
9	Enc2d	Read only	Encoder 2 position, 4th byte (lowest byte)
10	Battery volts	Read only	The supply battery voltage
11	Motor 1 current	Read only	The current through motor 1
12	Motor 2 current	Read only	The current through motor 2
13	Software Revision	Read only	Software Revision Number
14	Acceleration rate	R/W	Optional Acceleration register
15	Mode	R/W	Mode of operation (see below)
16	Command	R/W	Used for reset of encoder counts and module address changes

Tabla 3. Registros internos de la tarjeta MD25

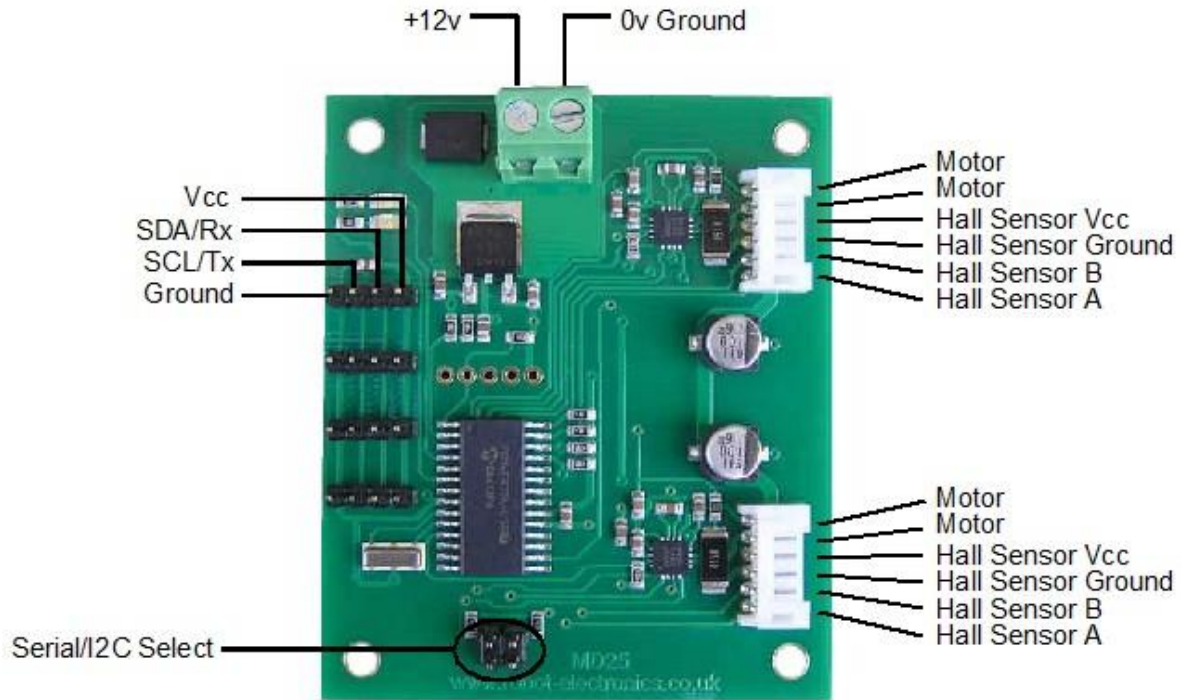


Ilustración 38. Vista superior de la tarjeta Md25

Los motores EMG30. Son motores de corriente directa que trabajan a doce volts, tienen un encoder integrado que nos permite conocer su velocidad con una resolución de un grado, proporcionan un torque de 1.5 kg/cm, cuentan con una caja de engranes con relación de 30:1, la flecha del motor después de la caja de engranes tiene una velocidad máxima de 170 rpm y mínima de 1.5 rpm, consume un máximo de 2.5 A de corriente con rotor bloqueado, y tiene un consumo nominal de 4.2 W. También tienen sensores de efecto Hall, que permiten ver la tensión de la batería, la corriente que consumen, la velocidad y aceleración.



Ilustración 39. Motor Emg30 con caja de engranes acoplado

9.2. Etapa de potencia y alimentación

La alimentación eléctrica del robot manipulador móvil es una batería recargable de doce volts, modelo: BR-1204, que puede suministrar cuatro [Amper hora].

Es necesario alimentar cinco servos de aeromodelismo de los cuales cuatro servos son modelo VS-2, y un servo VS-5, todos los servos funcionan con cinco volts de corriente directa. Cada servo tiene una corriente nominal de 0.2 amperes, por lo que será necesario bajar la tensión de doce volts a cinco. Para esta etapa se diseñó e implemento un circuito de regulación con reguladores (7809, 78R05). Estos reguladores tienen poco error a la salida en comparación con otros similares, así como protección de sobrecalentamiento, protección de sobre cargar ya sea por corriente o voltaje y un pin por si queremos deshabilitarlos. Se diseñaron dos etapas de regulación, la primera es de doce a nueve volts y la segunda etapa es de nueve a cinco volts. Estas dos etapas de regulación disminuyen la posibilidad de sobrecalentamiento en los reguladores, y para garantizar esto fue necesario la implementación de un ventilador, para disipar el calor de los reguladores. Ya que a pesar de que la corriente nominal de cada servo es de tan solo 0.2 [A], los servos consumen hasta 0.8 [A] cuando la flecha está bloqueada.

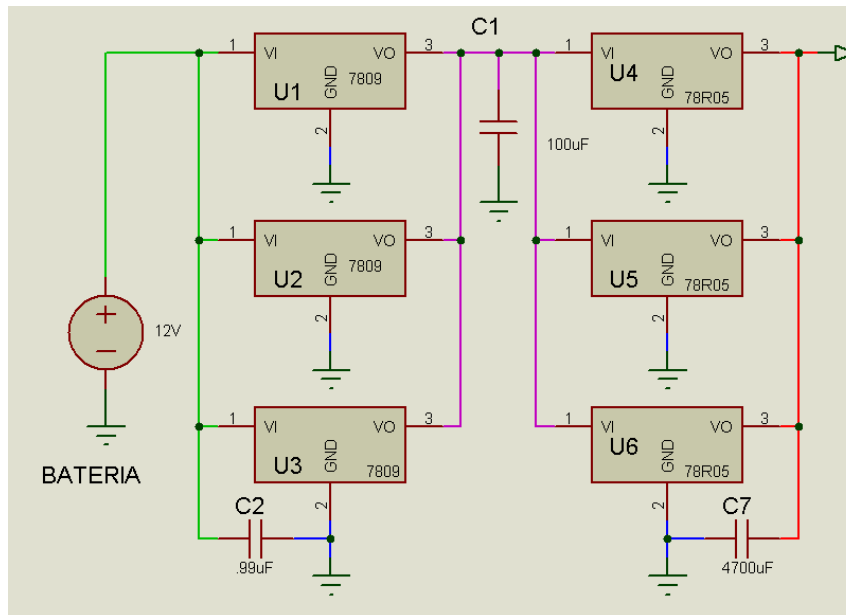


Ilustración 40. Diseño de la etapa de regulación de doce a 5 volts. Fue necesario la puesta en paralelo de los reguladores debido a la corriente necesitada.

Programas

9.3. Programa, cinemática directa del brazo serial en configuración antropomórfica.

Brazo serial en configuración antropomórfica. Autor: Pablo Carlos Ávila.

```

In[7]:= ClearAll;
Manipulate[
  Q[theta_, alpha_] :=
  { Cos[theta] - Sin[theta] * Cos[alpha] Sin[theta] * Sin[alpha]
    Sin[theta] Cos[theta] * Cos[alpha] - Cos[theta] * Sin[alpha]
      0 Sin[alpha] Cos[alpha] };
  a[theta_, b_, d_] := {d * Cos[theta], d * Sin[theta], b};

  Grafica[tabla_, ejex_, ejey_, rojo_, verde_, azul_] :=
  ListPlot[tabla,
    ImageSize -> 500,
    Joined -> True,
    Frame -> True,
    FrameLabel -> {ejex, ejey},
    BaseStyle -> {24, FontFamily -> "Arial"},
    GridLines -> Automatic,
    PlotStyle -> {AbsoluteThickness[6],
      RGBColor[rojo, verde, azul]};

  (*Los parámetros a utilizar son:
  ai-1: ángulo de Z_i a Z_(i+1) medida sobre el eje de X^i
  ai-1: distancia entre eje Z_i a Z_(i+1) medida sobre el eje de X^i
  bi: distancia entre el eje X^(i-1) a X^i medida sobre Z_i
  ei.: ángulo de X^(i-1) a X^i medida sobre el eje de Z_i
  *)

```

i	α_i	b_i	a_i	θ_i
1	$\frac{\pi}{2}$	alturainicial	0	$+\frac{\pi}{2} + \theta_1$
2	0	0	largodelbrazo1	θ_2
3	0	0	largodelbrazo2	$\pi - \theta_3$;*)
4	$\frac{\pi}{2}$	0	0	θ_4
5	0	largo del efector final	0	θ_5
6	0	0	ancho del efector final	$\frac{\pi}{2}$

Ilustración 41. Cinemática Directa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

```

(*parametros espaciales en caso de que el
brazo no este en el sistema de referencia absoluto*)
a0 = {0, 0, 0};

(*parametros depositados de la tabla*)
(*a1=a[θ1,b1,a1]; Q1=Q[θ1,α1];*)

a1 = a[+  $\frac{\text{Pi}}$  + θ1, 21.5, 0]; a2 = a[θ2, 0, 15]; a3 = a[Pi - θ3, 0, 15];
Q1 = Q[+  $\frac{\text{Pi}}$  + θ1,  $\frac{\text{Pi}}$ ]; Q2 = Q[θ2, 0]; Q3 = Q[Pi - θ3, 0];

a4 = a[θ4, 0, 0]; a5 = a[θ5, 12, 0]; a6 = a[ $\frac{\text{Pi}}$ , 0, 3];
Q4 = Q[θ4,  $\frac{\text{Pi}}$ ]; Q5 = Q[θ5, 0]; Q6 = Q[ $\frac{\text{Pi}}$ , 0];

cero = {0, 0, 0};
u1 = a0 + a1;
u2 = a0 + a1 + Q1.a2;
u3 = a0 + a1 + Q1.a2 + Q1.Q2.a3;
u4 = a0 + a1 + Q1.a2 + Q1.Q2.a3 + Q1.Q2.Q3.a4;
u5 = a0 + a1 + Q1.a2 + Q1.Q2.a3 + Q1.Q2.Q3.a4 + Q1.Q2.Q3.Q4.a5;
u6 = a0 + a1 + Q1.a2 + Q1.Q2.a3 + Q1.Q2.Q3.a4 + Q1.Q2.Q3.Q4.a5 + Q1.Q2.Q3.Q4.Q5.a6;

S1p = Line[{cero, u1}];
S2p = Line[{u1, u2}];
S3p = Line[{u2, u3}];
S4p = Line[{u3, u4}];
S5p = Line[{u4, u5}];
S6p = Line[{u5, u6}];

barral = Graphics3D[{AbsoluteThickness[30], RGBColor[1, 0, 0], S1p}];
barra2 = Graphics3D[{AbsoluteThickness[15], RGBColor[0, 1, 0], S2p}];
barra3 = Graphics3D[{AbsoluteThickness[15], RGBColor[0, 0, 1], S3p}];
barra4 = Graphics3D[{AbsoluteThickness[5], RGBColor[0, 0, 1], S4p}];
barra5 = Graphics3D[{AbsoluteThickness[5], RGBColor[.5, 0, 1], S5p}];
barra6 = Graphics3D[{AbsoluteThickness[5], RGBColor[0, 0, 0], S6p}];

Show[barral, barra2, barra3, barra4,
barra5, barra6, Axes → True, AxesLabel → {"X", "Y", "Z"},
ImageSize → 500,
BaseStyle → {12, FontFamily → "Arial"},
PlotRange → {{-50, 50}, {-50, 50}, {0, 70}},
{θ1, 0, Pi, Pi / 180}, {θ2, 10 * Pi / 180, Pi, Pi / 180}, {θ3, 28 * Pi / 180, Pi, Pi / 180}
, {θ4, 0, Pi, Pi / 180}, {θ5, 0, Pi, Pi / 180}]

```

Ilustración 42. Cinemática Directa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

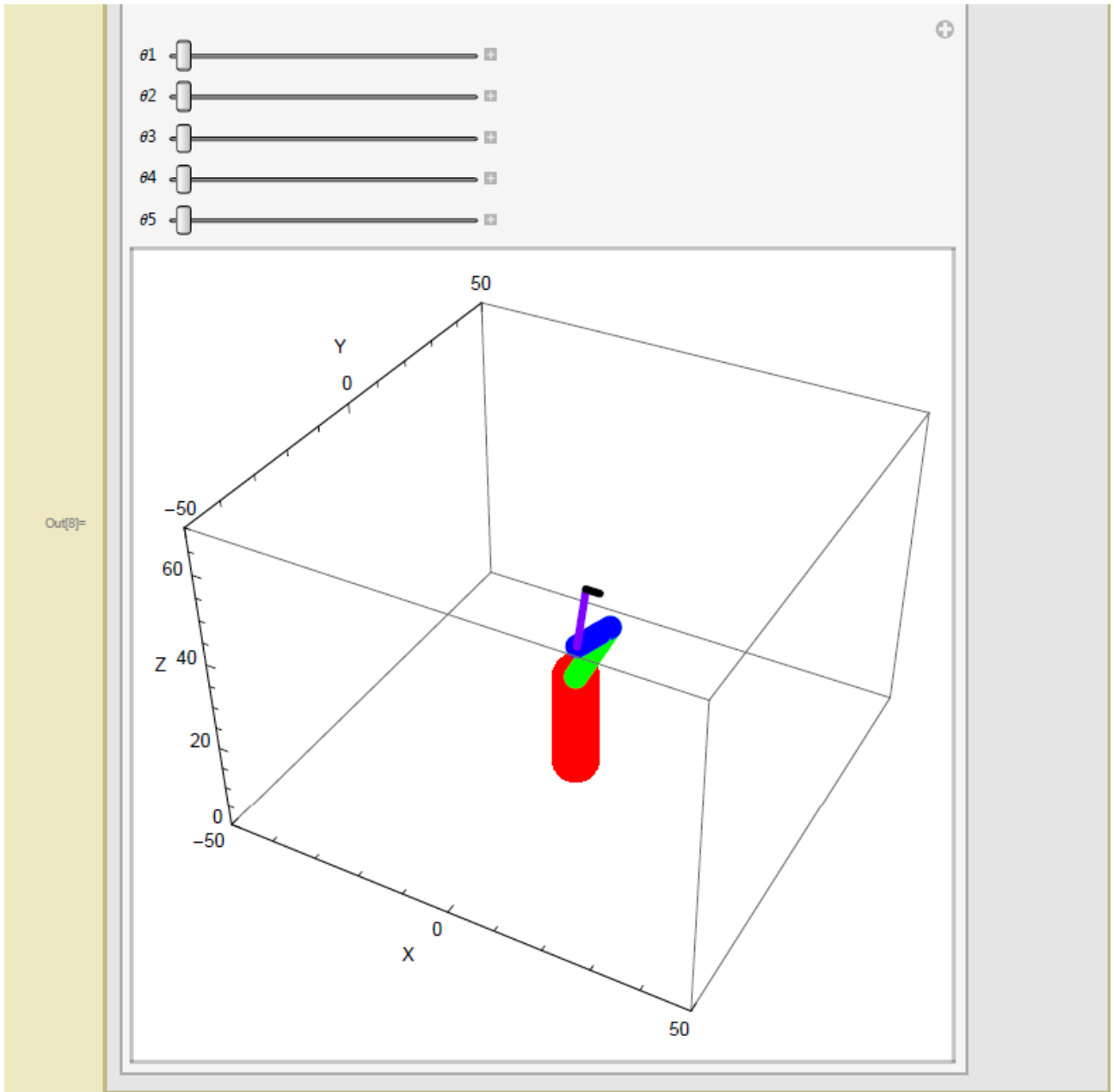


Ilustración 43. Cinemática Directa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

9.4. Programa, cinemática inversa del brazo serial en configuración antropomorfica.

```
Cinemática Inversa. Pablo Carlos Ávila.  
Cinemática inversa  
  
In[3]:= ClearAll;  
Clear[θ1, θ2, θ3, θ4, θ5];  
Q[theta_, alpha_] :=  
  
$$\begin{pmatrix} \cos[\theta] & -\sin[\theta] \cos[\alpha] & \sin[\theta] \sin[\alpha] \\ \sin[\theta] & \cos[\theta] \cos[\alpha] & -\cos[\theta] \sin[\alpha] \\ 0 & \sin[\alpha] & \cos[\alpha] \end{pmatrix};$$
  
a[theta_, b_, d_] := {d * Cos[theta], d * Sin[theta], b};  
  
Grafica [tabla_, ejex_, ejey_, rojo_, verde_, azul_] :=  
  ListPlot[tabla,  
    ImageSize → 500,  
    Joined → True,  
    Frame → True,  
    FrameLabel → {ejex, ejey},  
    BaseStyle → {24, FontFamily → "Arial"},  
    GridLines → Automatic,  
    PlotStyle → {AbsoluteThickness[6],  
      RGBColor[rojo, verde, azul]};  
(*Partes en las que se discretiza la trayectoria*)  
T = 128;  
  
(*Tamaño de las graficas*)
```

Ilustración 44. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

Parámetros de la configuración del Robot

In[12]=

```
(*Los parámetros a utilizar son:
ai-1: ángulo de  $\hat{z}_i$  a  $\hat{z}_{(i+1)}$  medida sobre el eje de  $X^{\wedge}_i$ 
ai-1: distancia entre eje  $\hat{z}_i$  a  $\hat{z}_{(i+1)}$  medida sobre el eje de  $X^{\wedge}_i$ 
bi: distancia entre el eje  $X^{\wedge}_{(i-1)}$  a  $X^{\wedge}_i$  medida sobre  $\hat{z}_i$ 
ei.: ángulo de  $X^{\wedge}_{(i-1)}$  a  $X^{\wedge}_i$  medida sobre el eje de  $\hat{z}_i$ 
```

i	α_i	b_i	a_i	θ_i
1	$\frac{\pi}{2}$	alturainicial	0	$+\frac{\pi}{2}+\theta_1$
2	0	0	largodelbrazo1	θ_2
3	0	0	largodelbrazo2	$\pi-\theta_3$;*)
4	$\frac{\pi}{2}$	0	0	θ_4
5	0	largo del efector final	0	θ_5
6	0	0	ancho del efector final	$\frac{\pi}{2}$

```
(*parametros espaciales en caso de que el
brazo no este en el sistema de referencia absoluto*)
a0 = {0, 0, 0};
```

```
(*parametros depositados de la tabla*)
(*al=a[ $\theta_1$ , $b_1$ , $a_1$ ]; Q1=Q[ $\theta_1$ , $\alpha_1$ ];*)
```

```
a1 = a[ $+\frac{\pi}{2} + \theta_1$ , 21.5, 0]; a2 = a[ $\theta_2$ , 0, 15]; a3 = a[ $\pi - \theta_3$ , 0, 15];
```

```
Q1 = Q[ $+\frac{\pi}{2} + \theta_1$ ,  $\frac{\pi}{2}$ ]; Q2 = Q[ $\theta_2$ , 0]; Q3 = Q[ $\pi - \theta_3$ , 0];
```

```
a4 = a[ $\theta_4$ , 0, 0]; a5 = a[ $\theta_5$ , 12, 0]; a6 = a[ $\frac{\pi}{2}$ , 0, 3];
```

```
Q4 = Q[ $\theta_4$ ,  $\frac{\pi}{2}$ ]; Q5 = Q[ $\theta_5$ , 0]; Q6 = Q[ $\frac{\pi}{2}$ , 0];
```

Planteamiento del vector u

In[18]=

```
cero = {0, 0, 0};
u = a0 + a1 + Q1.a2 + Q1.Q2.a3 + Q1.Q2.Q3.a4 + Q1.Q2.Q3.Q4.a5;
u6 = a0 + a1 + Q1.a2 + Q1.Q2.a3 + Q1.Q2.Q3.a4 + Q1.Q2.Q3.Q4.a5 + Q1.Q2.Q3.Q4.Q5.a6;
MatrixForm[Simplify[u]]
```

Ilustración 45. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

Definición de la matriz Jacobiana y Hesiana

```
In[23]= (*Jakobiano*)
J = {
  {∂θ1 u[[1]], ∂θ2 u[[1]], ∂θ3 u[[1]]},
  {∂θ1 u[[2]], ∂θ2 u[[2]], ∂θ3 u[[2]]},
  {∂θ1 u[[3]], ∂θ2 u[[3]], ∂θ3 u[[3]]}
};
MatrixForm[J];
(*Hesiano gradiente del jacobiano*)
H = {
  {∂θ1,θ1 u[[1]], ∂θ1,θ2 u[[1]], ∂θ1,θ3 u[[1]]},
  {∂θ2,θ1 u[[1]], ∂θ2,θ2 u[[1]], ∂θ2,θ3 u[[1]]},
  {∂θ3,θ1 u[[1]], ∂θ3,θ2 u[[1]], ∂θ3,θ3 u[[1]]}
};
MatrixForm[H];
Fin
```

Definición una trayectoria

```
In[28]= (*Punto inicial de la trayectoria*)
P1 = {9, -7, 25};
(*Punto final de la trayectoria*)
P2 = {30, 0, 2};

tray = ParametricPlot3D[{
  P1[[1]] +  $\frac{i}{T}$  * (P2[[1]] - P1[[1]]),
  P1[[2]] +  $\frac{i}{T}$  * (P2[[2]] - P1[[2]]),
  P1[[3]] +  $\frac{i}{T}$  * (P2[[3]] - P1[[3]])},
{i, 0, T}]
```

Solución de la Posición Inicial del robot

```
In[31]= θ4 = 120 * Degree;
PosHome = FindRoot[{
  u[[1]] == P1[[1]],
  u[[2]] == P1[[2]],
  u[[3]] == P1[[3]]},
{θ1, 50 * Degree},
{θ2, 10 * Degree},
{θ3, 20 * Degree},
MaxIterations → 50]

({θ1, θ2, θ3} /. PosHome) / Degree
```

Ilustración 46. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

Solucion de la trayectoria con un perfil Cubico [Cubi]

Solucion de la trayectoria con un perfil ARMONICO SIMPLE[Cosenoidal]

$$P(t) = P_0 + \left(\frac{3t^2}{t_f^2} - \frac{2t^3}{t_f^3} \right) (P_f - P_0) \quad ;$$

Solucion de la trayectoria con un perfil Cubico [Cubi]

Solucion de la trayectoria con un perfil ARMONICO SIMPLE[Cosenoidal]

```
In[104]:=
P(t) = P_0 + \left( \frac{3t^2}{t_f^2} - \frac{2t^3}{t_f^3} \right) (P_f - P_0) \quad ;

(*T=50;*)
e1i = e1 /. PosHome;
e2i = e2 /. PosHome;
e3i = e3 /. PosHome;

For[i = 0, i <= T, i++,
  PCubix = P1[[1]] + (P2[[1]] - P1[[1]]) * \left( \frac{3 * i^2}{T^2} - \frac{2 * i^3}{T^3} \right);
  PCubiy = P1[[2]] + (P2[[2]] - P1[[2]]) * \left( \frac{3 * i^2}{T^2} - \frac{2 * i^3}{T^3} \right);
  PCubiz = P1[[3]] + (P2[[3]] - P1[[3]]) * \left( \frac{3 * i^2}{T^2} - \frac{2 * i^3}{T^3} \right);
  PosRobotCubi[i] = FindRoot[{
    u[[1]] == PCubix,
    u[[2]] == PCubiy,
    u[[3]] == PCubiz,
    {e1, e1i},
    {e2, e2i},
    {e3, e3i},
    MaxIterations -> 50];
  e1i = e1 /. PosRobotCubi[i];
  e2i = e2 /. PosRobotCubi[i];
  e3i = e3 /. PosRobotCubi[i];
];
PosRobotCubi[0]
PosRobotCubi[10];
(*Graficaci3n de la Posici3n*)
tabla1Cubi = Table[{t, (e1 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}];
tabla2Cubi = Table[{t, (e2 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}];
tabla3Cubi = Table[{t, (e3 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}];

Theta1Cubi = Grafica[tabla1Cubi, "t", "e1", 1, 0, 0];
Theta2Cubi = Grafica[tabla2Cubi, "t", "e2", 0, 1, 0];
Theta3Cubi = Grafica[tabla3Cubi, "t", "e3", 0, 0, 1];
```

Ilustraci3n 47. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

```

Maximo1Cubi = Max[Table[{(e1 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}]];
Maximo2Cubi = Max[Table[{(e2 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}]];
Maximo3Cubi = Max[Table[{(e3 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}]];

Minimo1Cubi = Min[Table[{(e1 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}]];
Minimo2Cubi = Min[Table[{(e2 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}]];
Minimo3Cubi = Min[Table[{(e3 /. PosRobotCubi[t]) / Degree}, {t, 0, T, 1}]];

Show[Theta1Cubi,
  Graphics[Text["e1", {.5 * T, Minimo1Cubi + .5 * (Maximo1Cubi - Minimo1Cubi)}]],
  ImageSize -> TAM
]
Show[Theta2Cubi,
  Graphics[Text["e2", {.5 * T, Minimo2Cubi + .5 * (Maximo2Cubi - Minimo2Cubi)}]],
  ImageSize -> TAM
]
Show[Theta3Cubi,
  Graphics[Text["e3", {.5 * T, Minimo3Cubi + .5 * (Maximo3Cubi - Minimo3Cubi)}]],
  ImageSize -> TAM
]

```

```
{e1 -> 0.909753, e2 -> 0.204981, e3 -> 0.454141}
```

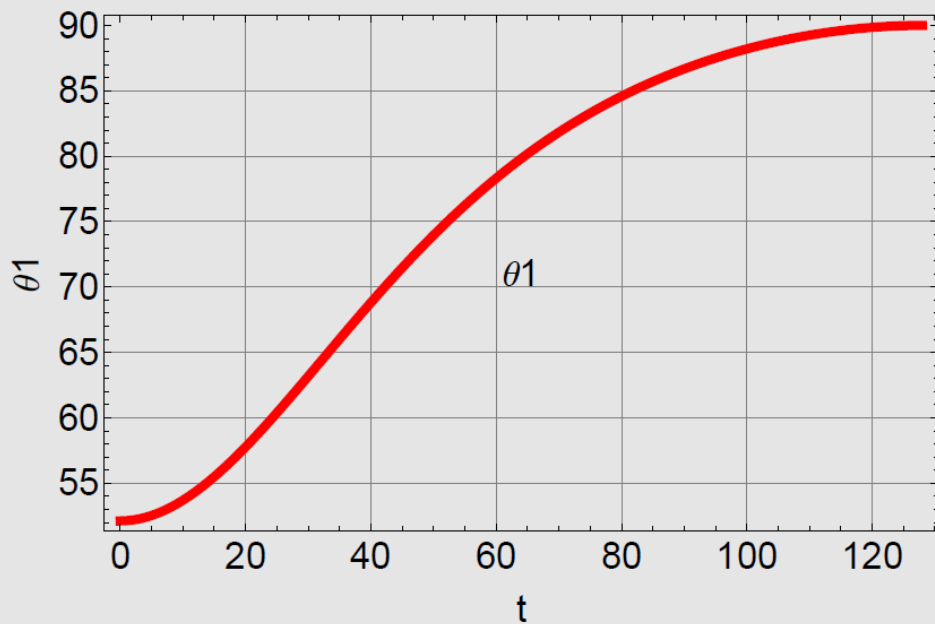


Ilustración 48. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

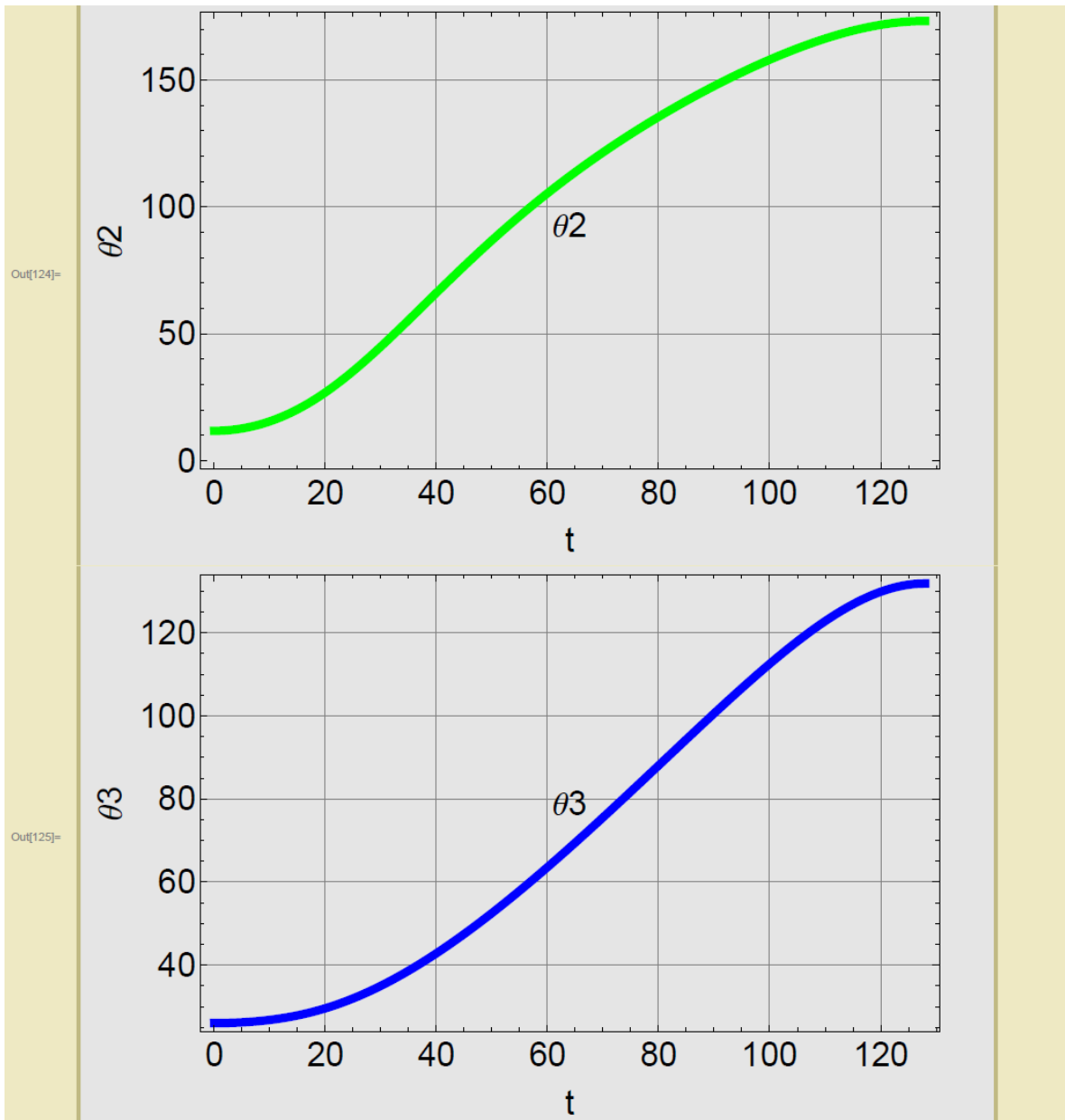


Ilustración 49. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

```

In[152]:= Clear[e1, e2, e3]

(*T=50;*)

For[i = 0, i ≤ T, i++,
  t = i;
  AuxCubi =  $\left(-\frac{12 i}{T^3} + \frac{6}{T^2}\right) * (P2[[1]] - P1[[1]]);$ 
  AuyCubi =  $\left(-\frac{12 i}{T^3} + \frac{6}{T^2}\right) * (P2[[2]] - P1[[2]]);$ 
  AuzCubi =  $\left(-\frac{12 i}{T^3} + \frac{6}{T^2}\right) * (P2[[3]] - P1[[3]]);$ 

  Aucubi = {AuxCubi, AuyCubi, AuzCubi};
  AcelangCubi[i] = IJCubi[i].(Aucubi - MHCubi[i].VelangCubi[i]);

];

(*Grafica de la aceleración*)
AcelangCubi[0];
AcelangCubi[2];
AcelangCubi[3];
atabla1Cubi = Table[{t, AcelangCubi[t][[1]]}, {t, 0, T, 1}];
atabla2Cubi = Table[{t, AcelangCubi[t][[2]]}, {t, 0, T, 1}];
atabla3Cubi = Table[{t, AcelangCubi[t][[3]]}, {t, 0, T, 1}];

alfalCubi = Grafica[atabla1Cubi, "t", "α1", 1, 0, 0];
alfa2Cubi = Grafica[atabla2Cubi, "t", "α2", 0, 1, 0];
alfa3Cubi = Grafica[atabla3Cubi, "t", "α3", 0, 0, 1];

Maximo1CubiACEL = Max[Table[{AcelangCubi[t][[1]]}, {t, 0, T, 1}]];
Maximo2CubiACEL = Max[Table[{AcelangCubi[t][[2]]}, {t, 0, T, 1}]];
Maximo3CubiACEL = Max[Table[{AcelangCubi[t][[3]]}, {t, 0, T, 1}]];

Minimo1CubiACEL = Min[Table[{t, AcelangCubi[t][[1]]}, {t, 0, T, 1}]];
Minimo2CubiACEL = Min[Table[{t, AcelangCubi[t][[2]]}, {t, 0, T, 1}]];
Minimo3CubiACEL = Min[Table[{t, AcelangCubi[t][[3]]}, {t, 0, T, 1}]];

Show[alfalCubi,
Graphics[
Text["α1", {.5 * T, Minimo1CubiACEL + .5 * (Maximo1CubiACEL - Minimo1CubiACEL)}],
ImageSize → TAM]

Show[alfa2Cubi,
Graphics[
Text["α2", {.5 * T, Minimo2CubiACEL + .5 * (Maximo2CubiACEL - Minimo2CubiACEL)}],
ImageSize → TAM]

Show[alfa3Cubi,
Graphics[
Text["α3", {.5 * T, Minimo3CubiACEL + .5 * (Maximo3CubiACEL - Minimo3CubiACEL)}],
ImageSize → TAM]

```

Ilustración 50. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

```

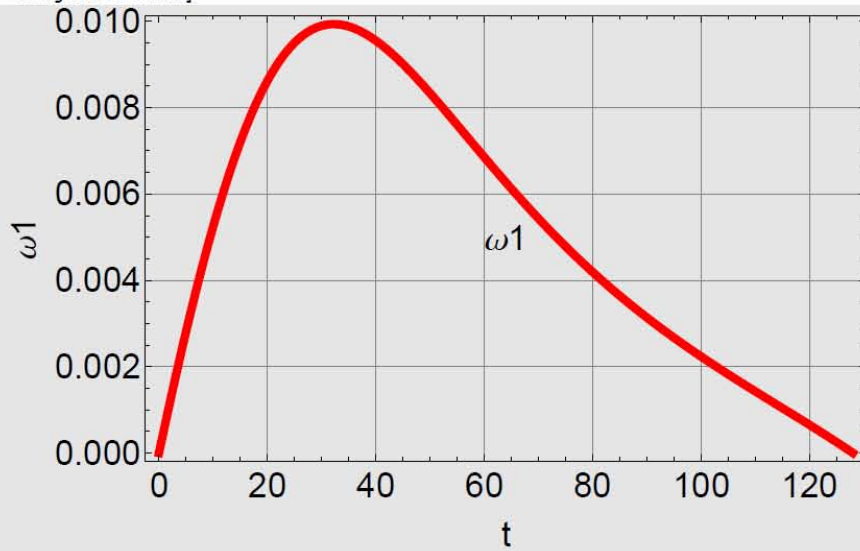
Minimo1CubiVEL = Min[Table[{VelangCubi[t][[1]]}, {t, 0, T, 1}]];
Minimo2CubiVEL = Min[Table[{VelangCubi[t][[2]]}, {t, 0, T, 1}]];
Minimo3CubiVEL = Min[Table[{VelangCubi[t][[3]]}, {t, 0, T, 1}]];
Show[omegalCubi ,
Graphics[
Text[" $\omega_1$ ", {.5 * T, Minimo1CubiVEL + .5 * (Maximo1CubiVEL - Minimo1CubiVEL)}]],
ImageSize -> TAM]

Show[omega2Cubi,
Graphics[
Text[" $\omega_2$ ", {.5 * T, Minimo2CubiVEL + .5 * (Maximo2CubiVEL - Minimo2CubiVEL)}]],
ImageSize -> TAM]

Show[omega3Cubi ,
Graphics[
Text[" $\omega_3$ ", {.5 * T, Minimo3CubiVEL + .5 * (Maximo3CubiVEL - Minimo3CubiVEL)}]],
ImageSize -> TAM]

```

Out[149]=



Out[150]=

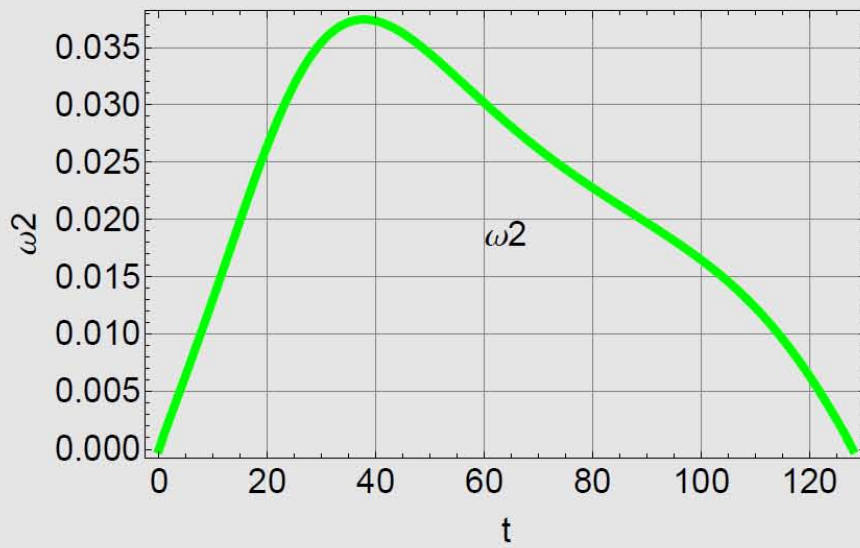
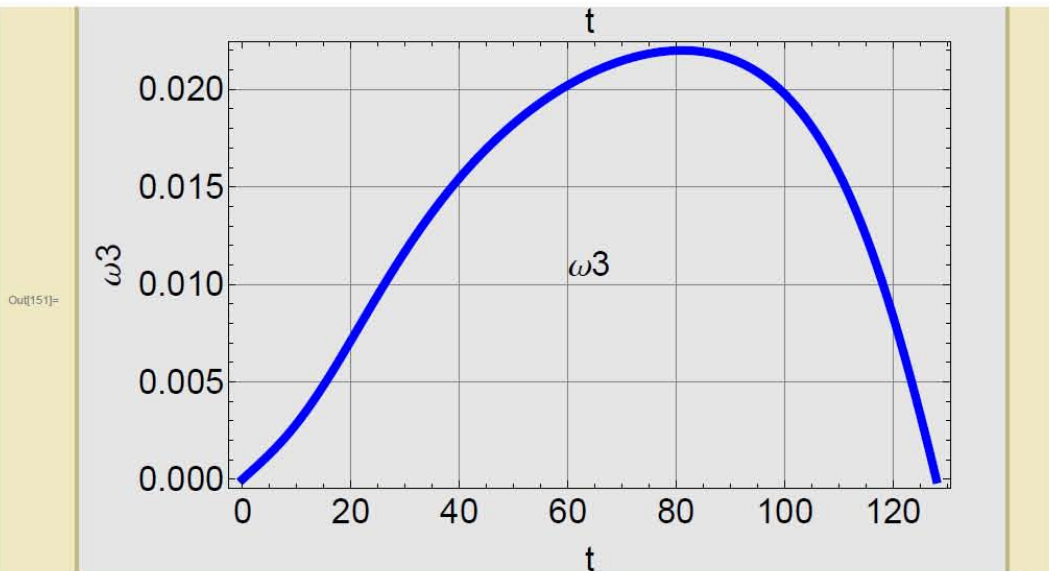


Ilustración 51. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.



Calculo de la Aceleración y gráfica

```
In[152]:=
Clear[ø1, ø2, ø3]

(*T=50;*)

For[i = 0, i ≤ T, i++,
  t = i;
  AuxCubi = (- 12 i / T^3 + 6 / T^2) * (P2[[1]] - P1[[1]]);
  AuyCubi = (- 12 i / T^3 + 6 / T^2) * (P2[[2]] - P1[[2]]);
  AuzCubi = (- 12 i / T^3 + 6 / T^2) * (P2[[3]] - P1[[3]]);

  Aucubi = {AuxCubi, AuyCubi, AuzCubi};
  AcelangCubi[i] = IJCubi[i] . (Aucubi - MHCubi[i] . VelangCubi[i]);

];

(*Grafica de la aceleración*)
AcelangCubi[0];
AcelangCubi[2];
AcelangCubi[3];
atabla1Cubi = Table[{t, AcelangCubi[t][[1]]}, {t, 0, T, 1}];
atabla2Cubi = Table[{t, AcelangCubi[t][[2]]}, {t, 0, T, 1}];
atabla3Cubi = Table[{t, AcelangCubi[t][[3]]}, {t, 0, T, 1}];

alfa1Cubi = Grafica[atabla1Cubi, "t", "α1", 1, 0, 0];
alfa2Cubi = Grafica[atabla2Cubi, "t", "α2", 0, 1, 0];
alfa3Cubi = Grafica[atabla3Cubi, "t", "α3", 0, 0, 1];
```

Ilustración 52. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

```

Maximo1CubiACEL = Max[Table[{AcelangCubi [t][[1]]}, {t, 0, T, 1}]];
Maximo2CubiACEL = Max[Table[{AcelangCubi [t][[2]]}, {t, 0, T, 1}]];
Maximo3CubiACEL = Max[Table[{AcelangCubi [t][[3]]}, {t, 0, T, 1}]];

Minimo1CubiACEL = Min[Table[{t, AcelangCubi [t][[1]]}, {t, 0, T, 1}]];
Minimo2CubiACEL = Min[Table[{t, AcelangCubi [t][[2]]}, {t, 0, T, 1}]];
Minimo3CubiACEL = Min[Table[{t, AcelangCubi [t][[3]]}, {t, 0, T, 1}]];

Show[alfaCubi ,
Graphics[
Text[" $\alpha_1$ ", {.5 * T, Minimo1CubiACEL + .5 * (Maximo1CubiACEL - Minimo1CubiACEL)}],
ImageSize -> TAM]

Show[alfa2Cubi ,
Graphics[
Text[" $\alpha_3$ ", {.5 * T, Minimo2CubiACEL + .5 * (Maximo2CubiACEL - Minimo2CubiACEL)}],
ImageSize -> TAM]

Show[alfa3Cubi ,
Graphics[
Text[" $\alpha_3$ ", {.5 * T, Minimo3CubiACEL + .5 * (Maximo3CubiACEL - Minimo3CubiACEL)}],
ImageSize -> TAM]

```

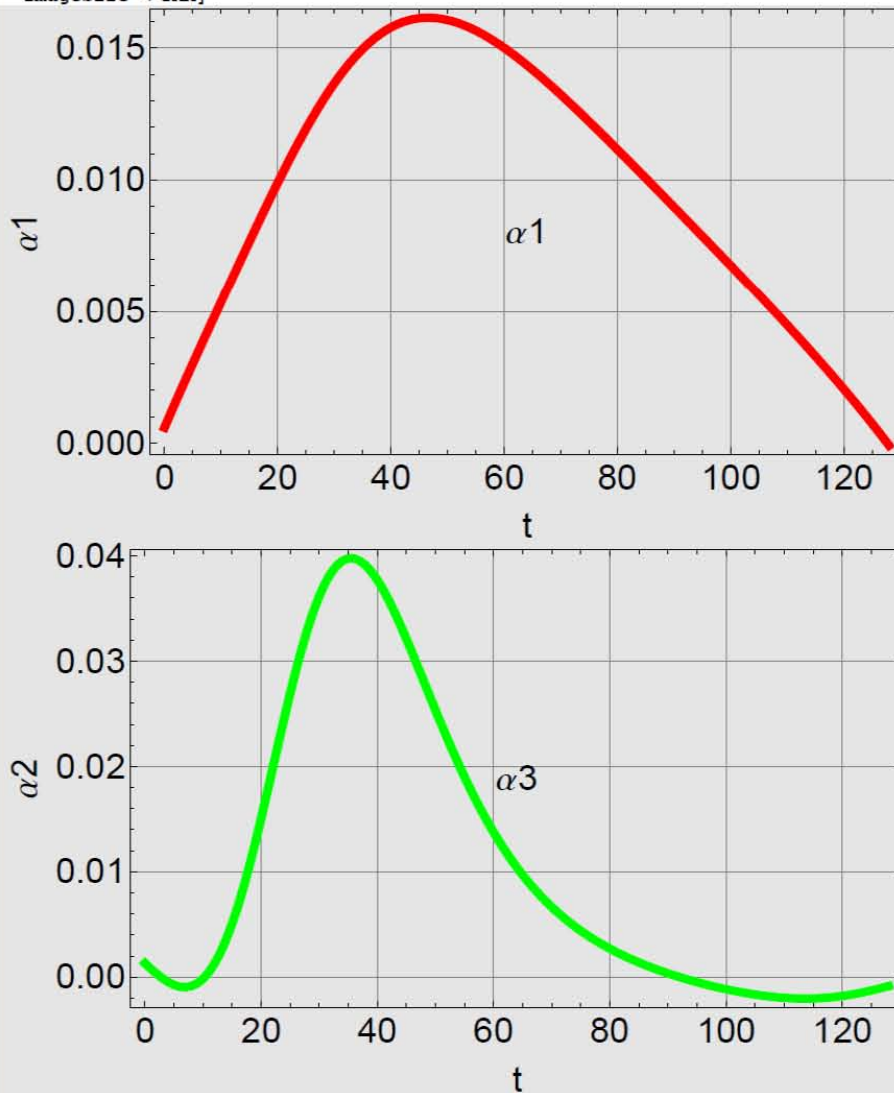
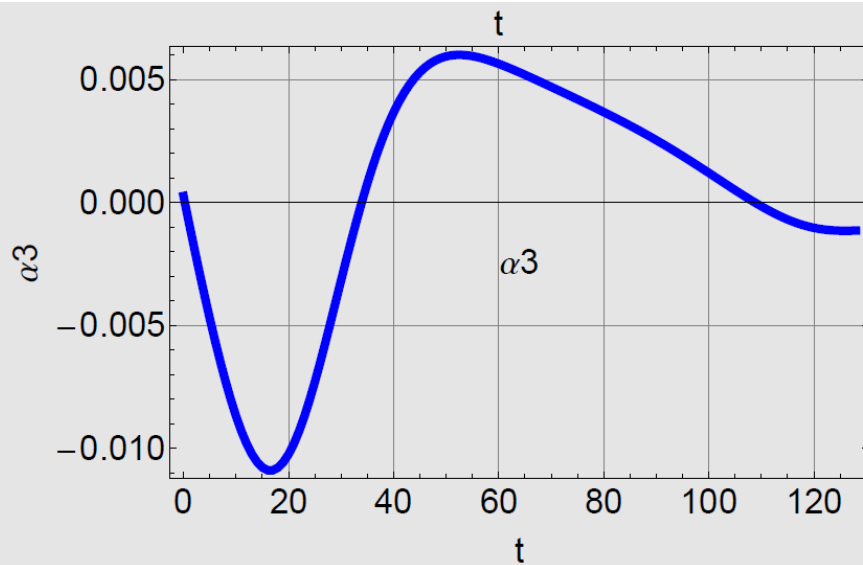


Ilustración 53. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.



Simulación

In[172]:=

```

Clear[ø1, ø2, ø3];
ø4 = 120 * Degree;
Manipulate[
  cero = {0, 0, 0};
  cubLinea1 = Line[{cero, a0}];
  cubLinea2 = Line[{a0, a0 + a1}] /. PosRobotCubi[i];
  cubLinea3 = Line[{a0 + a1, a0 + a1 + Q1.a2}] /. PosRobotCubi[i];
  cubLinea4 = Line[{a0 + a1 + Q1.a2, a0 + a1 + Q1.a2 + Q1.Q2.a3}] /. PosRobotCubi[i];
  cubLinea5 = Line[{a0 + a1 + Q1.a2 + Q1.Q2.a3,
    a0 + a1 + Q1.a2 + Q1.Q2.a3 + Q1.Q2.Q3.a4 + Q1.Q2.Q3.Q4.a5}] /. PosRobotCubi[i];

  cubbarral = Graphics3D[{AbsoluteThickness[0], RGBColor[1, 0, 0], cubLinea1}];
  cubbarra2 = Graphics3D[{AbsoluteThickness[30], RGBColor[1, 0, 0], cubLinea2}];
  cubbarra3 = Graphics3D[{AbsoluteThickness[15], RGBColor[0, 1, 0], cubLinea3}];
  cubbarra4 = Graphics3D[{AbsoluteThickness[15], RGBColor[0, 0, 1], cubLinea4}];
  cubbarra5 = Graphics3D[{AbsoluteThickness[10], RGBColor[.5, 0, 1], cubLinea5}];

  Show[cubbarral, cubbarra2, cubbarra3, cubbarra4,
    cubbarra5, tray, Axes -> True, AxesLabel -> {"X", "Y", "Z"},
    ImageSize -> Simulacion, BaseStyle -> {12, FontFamily -> "Comics"},
    PlotRange -> {{-50, 50}, {-50, 50}, {0, 50}},

```

Ilustración 54. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

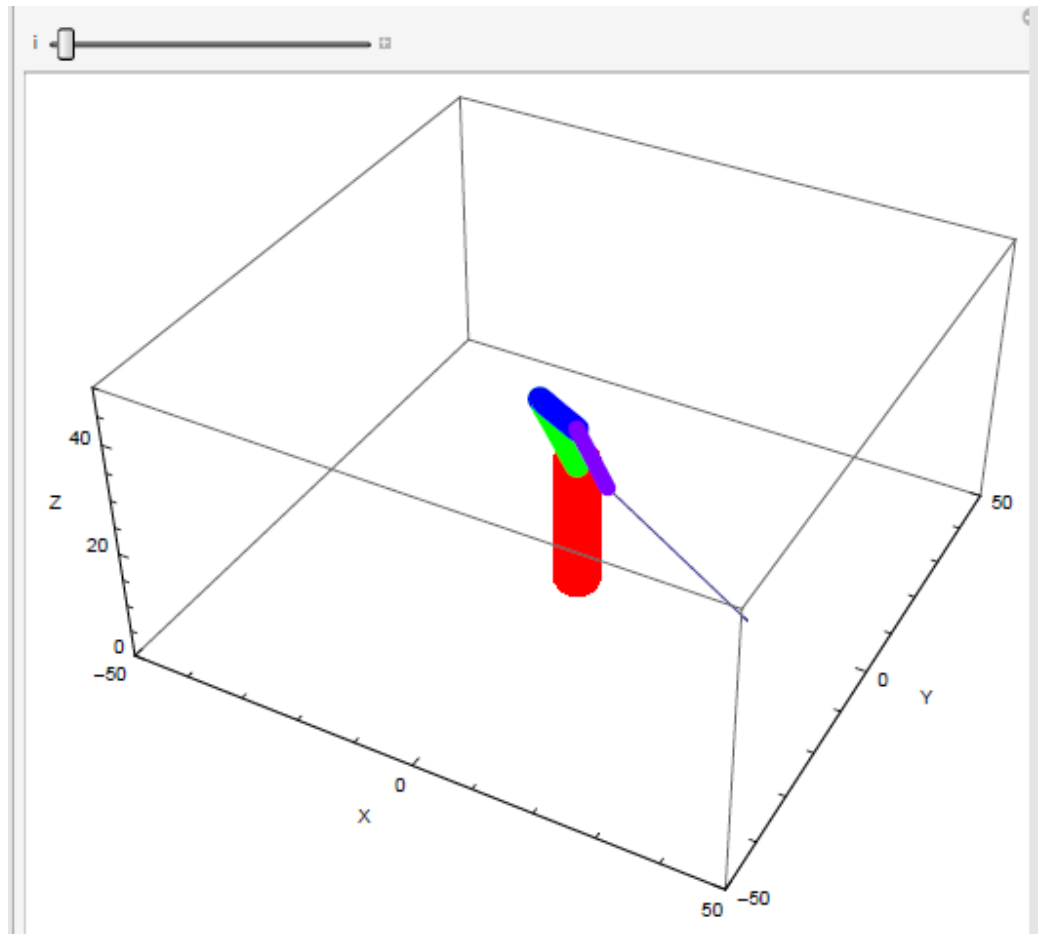


Ilustración 55. Cinemática inversa, del manipulador móvil. Obtenida con Wolfram Mathematica 9.

9.5. Diseño de la plataforma para fiducial

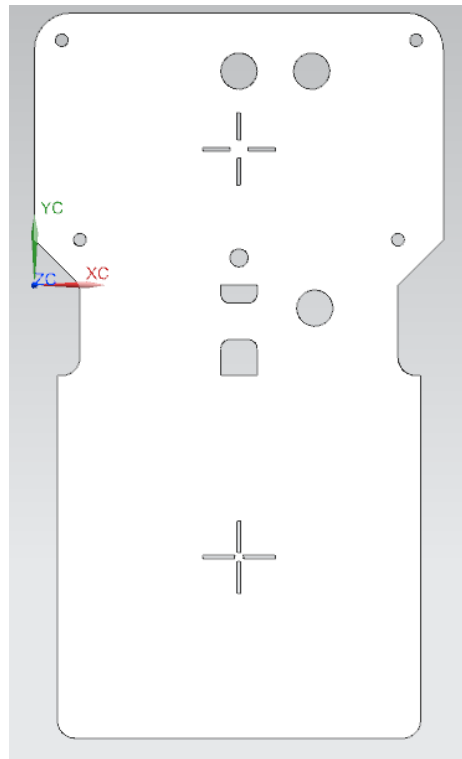


Ilustración 56. Diseño de la plataforma para el fiducial

9.6. Código de programación del microcontrolador Arduino R3

```
/*  
-----COMUNICACIÓN IC2-----  
*UNA RESISTENCIA DE 10K A 5V -ARDUINO PIN A4 SIGNAL DATA  
*UNA RESISTENCIA DE 10K A 5V -ARDUINO PIN A5 SIGNAL CLOCK  
&& pines a usar A4 y A5  
  
-----Servos-----  
//Pin--(rango de movimiento en grados) eslabón---color del cable  
//3--(0--180) base-----verde  
//5--(0--180) hombro1----verde blanco  
//6--(28-180) hombro2----café  
//9--(0--180) muñeca----anaranjado blanco  
//10--(0--180) Gmuñeca---azul  
//11--(10-150) ACmuñeca-café blanco  
  
&& pines a usar PWM 3 5 6 9 10 11
```

```

////////////////////////////////////
//Diagrama esquemático de los pines utilizados

+13- wifi
+12- wifi
+11--servo,
+10--servo,
+9---servo,
+8-wifi

+7-wifi
+6--servo,
+5--servo,
+A0-      +4- wifi
+A1-      +3--servo,
+A2-      +2- wifi
+A3-
+A4-IC2 SDA azul blanco  +1--COMUNICACION SERIAL
+A5-IC2 SCL naranja blanco +0--COMUNICACION SERIAL

*/

// Librerías utilizadas
#include <Wire.h>
#include <WiFi.h>
#include <WiFiUdp.h>
//librería servo
#include <Servo.h>

//configuración de parámetros para la md25
#define i2cAddress 0x58 //dirección de la tarjeta
#define speed1 0x00 //dirección del registro de velocidad 1
#define speed2 0x01 //dirección del registro de velocidad 2
#define VOLTREAD 0x0A

char ssid[] = "Nombre de la re"; //nombre de la red wifi
char pass[] = "clave de la red"; //clave de la red wifi (WPA)
int status = WL_IDLE_STATUS;

int keyIndex = 0; // largo de la clave de tu wifi (instrucción solo para WEP)

byte a,b,c,d,e,f,g,h; //se reciben los bytes que enviará MatLab
unsigned int localPort =666; // puerto local que enviara

```

```

char packetBuffer[255];

WiFiUDP Udp;

//definición de los servos a utilizar
Servo Base;
Servo Hombro1;
Servo Hombro2;
Servo Muneca;
Servo Gmuneca;
Servo ACmuneca;

void setup()
{
  Wire.begin();          //inicia la comunicación y espera un puerto abierto:
  Serial.begin(9600);    //velocidad de comunicación serial
  while (!Serial)
  {
    ; // Esperando la conexión.
  }
  // chequea la presencia de wifi shield:
  if (WiFi.status() == WL_NO_SHIELD)
  {
    Serial.println("sin WiFi shield ");
    // no continua hasta que se conecte:
    while(true);
  }

  // Esperando la conexión a la red Wifi :
  while ( status != WL_CONNECTED)
  {
    Serial.print("esperando conexión a SSID: ");
    Serial.println(ssid);
    // Connect to WPA/WPA2 network. Change this line if using open or WEP network:
    status = WiFi.begin(ssid, pass);

    // espera 10 segundos para la conexión:
    delay(10000);
    // start the server:
    // you're connected now, so print out the status:
    printWifiStatus();
    Udp.begin(localPort);
  }

  //Asignación de los pines a usar para cada servo
  Base.attach(2);        //(0--180) base-----verde

```

```

Hombro1.attach(3);      //(0--180) hombro1----verde blanco
Hombro2.attach(5);      //(28-180) hombro2----café
Muneca.attach(6);       //(0--180) muñeca-----anaranjado blanco
Gmuneca.attach(8);      //(0--180) Gmuñeca----azul
ACmuneca.attach(9);     //(10-150) ACmuñeca---café blanco

//posición de reposo del brazo
Base.write(90);
Hombro1.write(10);
Hombro2.write(28);
Muneca.write(150);
Gmuneca.write(180);
ACmuneca.write(10);
delay(500);             //Tiempo de espera para que se posicionen los servos
}

void loop()
{

// if there's data available, read a packet
int packetSize = Udp.parsePacket();
if(packetSize)
{
int len = Udp.read(packetBuffer,255);
if (len >0){ packetBuffer[len]=0;
//velocidades
a=packetBuffer[0];      // Velocidad1
b=packetBuffer[1];      // velocidad2
//servos
c=packetBuffer[2];      // Base
d=packetBuffer[3];      // Hombro1
e=packetBuffer[4];      // Hombro2
f=packetBuffer[5];      // Muneca
g=packetBuffer[6];      // Gmuneca
h=packetBuffer[7];      // ACmuneca

Serial.print(a);
Serial.print(" ");
Serial.print(b);
Serial.print(" ");
Serial.print(c);
Serial.print(" ");
Serial.print(d);
Serial.print(" ");
Serial.print(e);
Serial.print(" ");
Serial.print(f);

```

```

Serial.print(" ");
Serial.print(g);
Serial.print(" ");
Serial.print(h);

Serial.print(" MD25, Voltaje de la batería:");
int Voltaje=volts();
Serial.print("Batería v = ");
Serial.print(Voltaje/10, DEC);          // imprime el valor del voltaje que alimenta a la md25
Serial.print(".");
Serial.println(Voltaje%10, DEC);

Serial.println();

//transmisión de datos para la tarjeta md25
digitalWrite(13,HIGH);
Wire.beginTransmission(i2cAddress);    // manda velocidad 1
Wire.write(speed1);
Wire.write(a);
Wire.endTransmission();

Wire.beginTransmission(i2cAddress);    // manda velocidad 2
Wire.write(speed2);
Wire.write(b);
Wire.endTransmission();
digitalWrite(13,LOW);

// manipulador(int Base,int Hombro1,int Hombro2,int Muneca,int Gmuneca,int ACmuneca)
manipulador( c, d, e, f, g, h);

}}
else {
  WiFi.disconnect();
  WiFi.begin(ssid, pass);
}
}

void printWifiStatus()
{
  // print the SSID of the network you're attached to:
  Serial.print("SSID: ");
  Serial.println(WiFi.SSID());

  // print your WiFi shield's IP address:
  IPAddress ip = WiFi.localIP();

```

```

Serial.print("IP Address: ");
Serial.println(ip);

// print the received signal strength:
long rssi = WiFi.RSSI();
Serial.print("signal strength (RSSI):");
Serial.print(rssi);
Serial.println(" dBm");
}
//método para controlar los servomotores del manipulador
boolean manipulador(int gBase,int gHombro1,int gHombro2,int gMuneca,int gGmuneca,int gACmuneca)
{
//Base_____ (0--180)
if(gBase>179)
{
gBase=179;
}
if(gBase<0)
{
gBase=0;
}
//hombro1_____ (0--180)
if(gHombro1>179)
{
gHombro1=179;
}
if(gHombro1<0)
{
gHombro1=0;
}
//hombro2_____ (28-180)
if(gHombro2>179)
{
gHombro2=179;
}
if(gHombro2<28)
{
gHombro2=28;
}
//moñeca_____ (0--180)
if(gMuneca>179)
{
gMuneca=179;
}
if(gMuneca<0)
{
gMuneca=0;
}
}

```



```

//gmoñeca_____ (0--180)
if(gGmuneca>179)
{
  gGmuneca=179;
}
if(gGmuneca<0)
{
  gGmuneca=0;
}
//Acmuñeca_____ (10-150)
if(gACmuneca>150)
{
  gACmuneca=150;
}
if(gACmuneca<10)
{
  gACmuneca=10;
}
Base.write(gBase);
Hombro1.write(gHombro1);
Hombro2.write(gHombro2);
Muneca.write(gMuneca);
Gmuneca.write(gGmuneca);
ACmuneca.write(gACmuneca);
}
// método para leer el voltaje de la tarjeta md25
int volts()
{
  Wire.beginTransmission(i2cAddress);
  Wire.write(VOLTREAD);
  Wire.endTransmission();

  Wire.requestFrom(i2cAddress, 1);
  while(Wire.available() < 1);
  int Volts = Wire.read();

  return Volts;
}

```

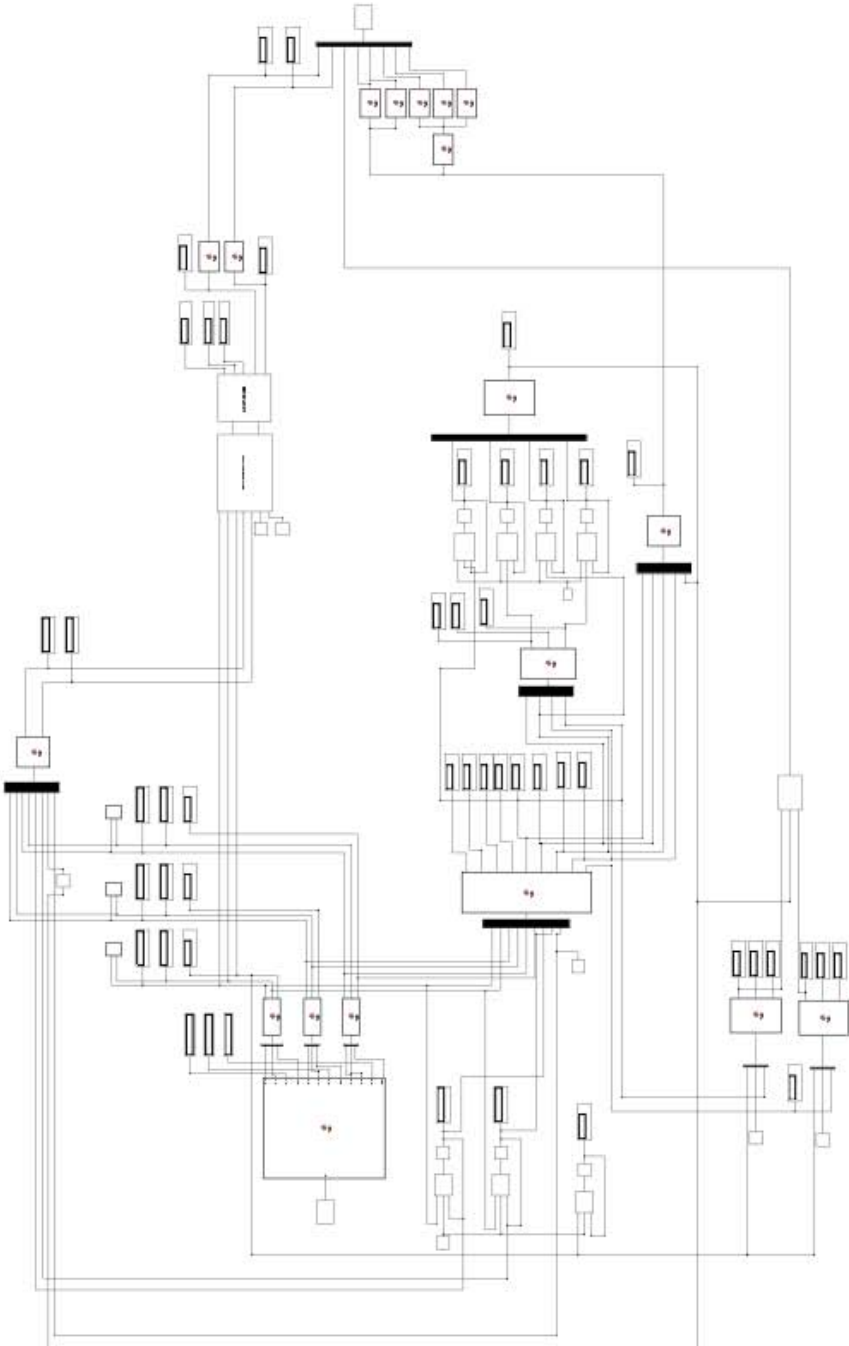


Ilustración 57. Programa en Matlab, visión general

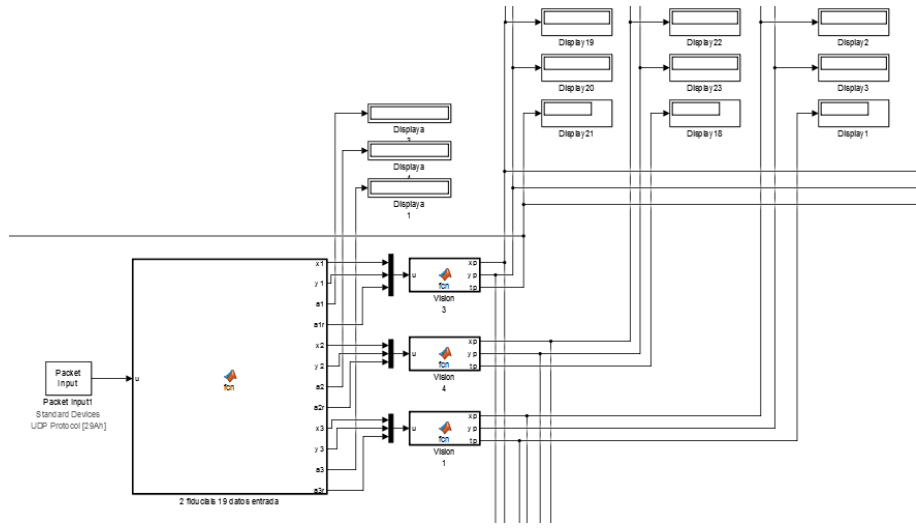


Ilustración 58. Procesamiento de los datos de visión.

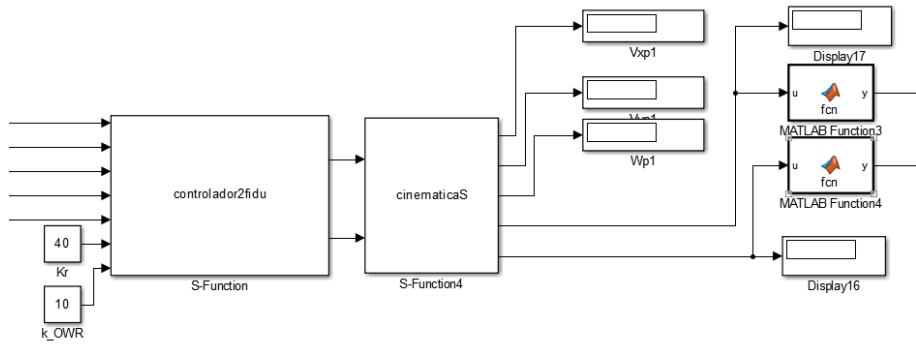


Ilustración 59. Controlador y Cinemática de la plataforma móvil

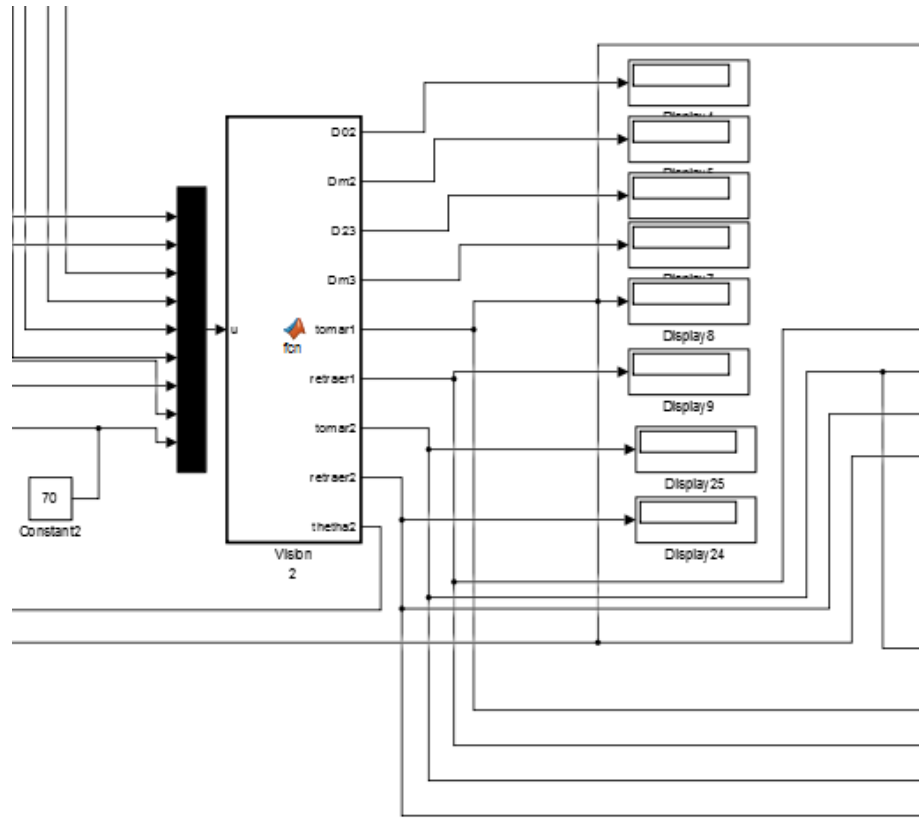


Ilustración 60. Bloque de coordinación de movimientos

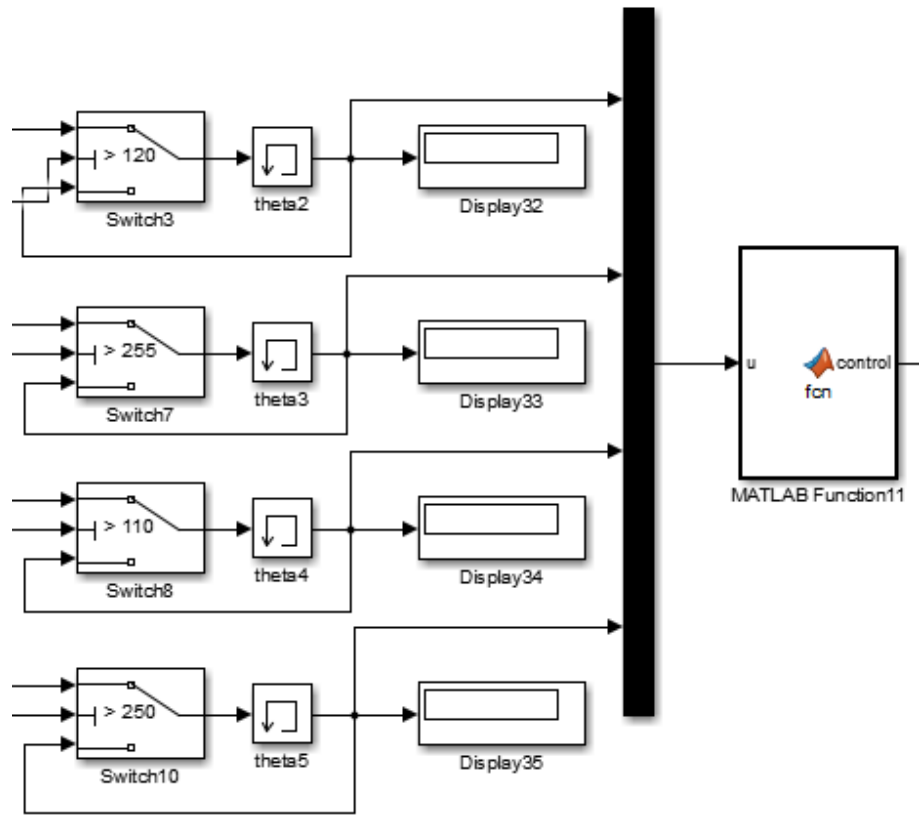


Ilustración 61. Máquina de estados

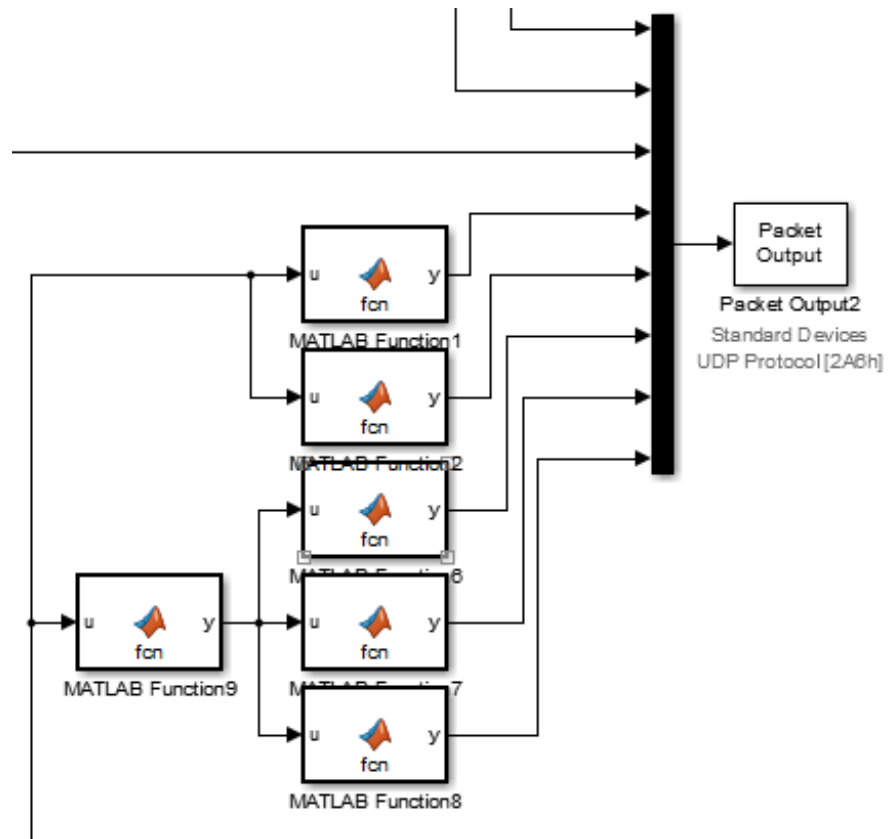


Ilustración 62. Envío de datos generados en Matlab al microcontrolador y cálculo de los ángulos para el manipulador

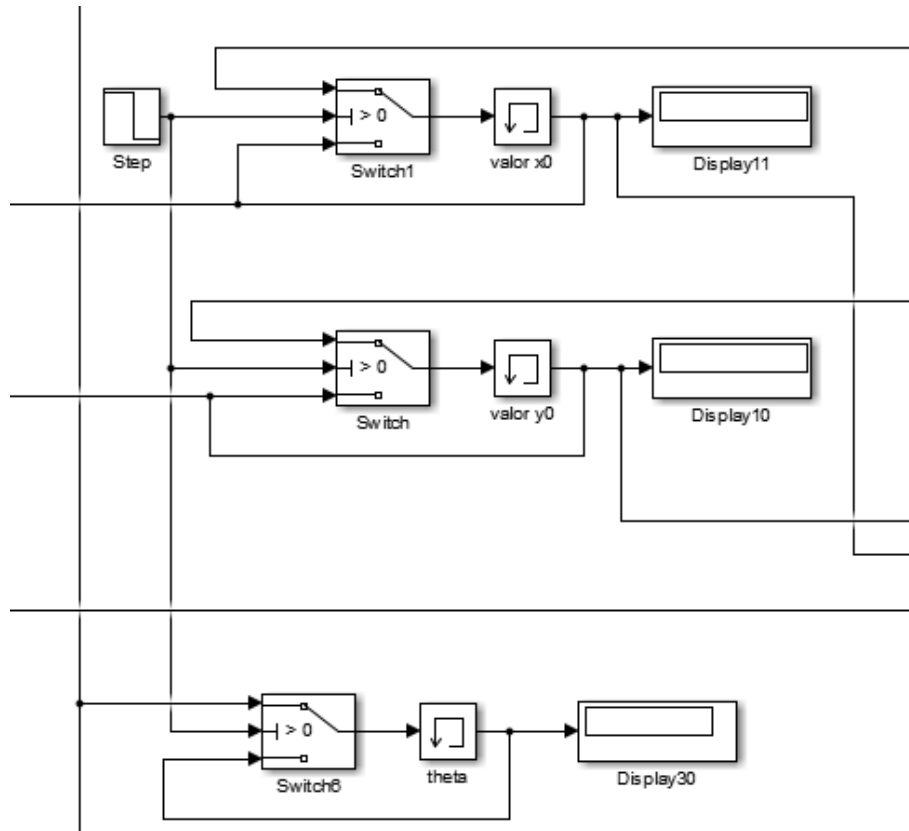


Ilustración 63. Generación de variables internas del programa

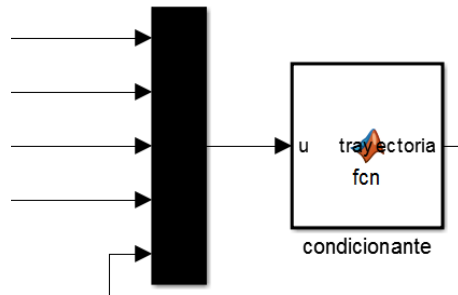


Ilustración 64. Selección de las distintas trayectorias a realizar por el manipulador.